ગુજરાત રાજ્યના શિક્ષણવિભાગના પત્ર-ક્રમાંક
મશબ / 1213 / 989 / છ, તા.29-1-2014 થી—મંજૂર

કમ્પ્યૂટર-અધ્યયન

ધોરણ 10

પ્રતિજ્ઞાપત્ર

ભારત મારો દેશ છે.
બધાં ભારતીયો મારાં ભાઈબહેન છે.
હું મારા દેશને ચાહું છું અને તેના સમૃદ્ધ અને
વૈવિધ્યપૂર્ણ વારસાનો મને ગર્વ છે.
હું સદાય તેને લાયક બનવા પ્રયત્ન કરીશ.
હું મારાં માતાપિતા, શિક્ષકો અને વડીલો પ્રત્યે આદર રાખીશ
અને દરેક જણ સાથે સભ્યતાથી વર્તીશ.
હું મારા દેશ અને દેશબાંધવોને મારી નિષ્ઠા અર્પું છું.
તેમનાં કલ્યાણ અને સમૃદ્ધિમાં જ મારું સુખ રહ્યું છે.

રાજ્ય સરકારની વિનામૂલ્યે યોજના હેઠળનું પુસ્તક

તમસોમા જ્યોતિર્ગમય
ગુજરાત રાજ્ય શાળા પાઠ્યપુસ્તક મંડળ
'વિદ્યાયન', સેક્ટર 10-એ, ગાંધીનગર-382010

© ગુજરાત રાજ્ય શાળા પાઠ્યપુસ્તક મંડળ, ગાંધીનગર
આ પાઠ્યપુસ્તકના સર્વ હક ગુજરાત રાજ્ય શાળા પાઠ્યપુસ્તક મંડળને હસ્તક છે.
આ પાઠ્યપુસ્તકનો કોઈ પણ ભાગ કોઈ પણ રૂપમાં ગુજરાત રાજ્ય શાળા
પાઠ્યપુસ્તક મંડળના નિયામકની લેખિત પરવાનગી વગર પ્રકાશિત કરી શકાશે નહિ.

વિષય-સલાહકાર
પ્રો. આર. પી. સોની

લેખન-સંપાદન
ડૉ. હર્ષલ આરોલકર (કન્વીનર)
ડૉ. સંજય એમ. શાહ
ડૉ. પ્રીતિ એસ. સજ્જા
ડૉ. કુંતલ પી. પટેલ

અનુવાદ
શ્રી રોહિત દોશી
શ્રી ગિરીશ એસ. બ્રહ્મભટ્ટ
શ્રી સાકેત એ. દવે

સમીક્ષા
શ્રી બિમલ કે. રાવલ
શ્રી રજનીકાન્ત એ. પંડ્યા
શ્રી પંકજ આર. શુક્લ
શ્રી રમણીક એલ. ગીલાતર
શ્રી રાજશ્રી એન. પડિયા
શ્રી સેજલ ડી. ત્રિવેદી

ભાષાશુદ્ધિ
શ્રી અશોકકુમાર એમ. દવે

સંયોજન
શ્રી આશિષ એચ. બોરીસાગર
(વિષય-સંયોજક : ગણિત)

નિર્માણ-આયોજન
શ્રી હરેશ એસ. લીમ્બાચીયા
(નાયબ નિયામક : શૈક્ષણિક)

મુદ્રણ-આયોજન
શ્રી હરેશ એસ. લીમ્બાચીયા
(નાયબ નિયામક : ઉત્પાદન)

પ્રસ્તાવના

માધ્યમિક અને ઉચ્ચતર માધ્યમિક શિક્ષણ બોર્ડે ઑપન સોર્સ ઑપરેટિંગ સિસ્ટમ અને તેને સુસંગત વિવિધ મુદ્દાઓ માટેના કમ્પ્યૂટર-અધ્યયનને લગતા ઓપન સોર્સ સૉફ્ટવેર આધારિત નવો અભ્યાસક્રમ તૈયાર કર્યો છે. આ અભ્યાસક્રમ ગુજરાત સરકાર દ્વારા મંજૂર કરવામાં આવ્યો છે.

ગુજરાત સરકાર દ્વારા મંજૂર થયેલા **ધોરણ 10**ના **કમ્પ્યૂટર-અધ્યયન** વિષયના નવા અભ્યાસક્રમ અનુસાર તૈયાર કરવામાં આવેલું આ પાઠ્યપુસ્તક વિદ્યાર્થીઓ સમક્ષ મૂકતાં મંડળ આનંદ અનુભવે છે.

આ વિષયનું અંગ્રેજી માધ્યમનું પાઠ્યપુસ્તક પ્રસિદ્ધ કરતાં પહેલાં એની હસ્તપ્રતની આ સ્તરે શિક્ષણકાર્ય કરતા શિક્ષકો અને તજ્જ્ઞો દ્વારા સર્વાંગી સમીક્ષા કરાવવામાં આવી છે અને તેમનાં સૂચનો અનુસાર હસ્તપ્રતમાં યોગ્ય સુધારા-વધારા કર્યા પછી આ પાઠ્યપુસ્તક પ્રસિદ્ધ કરવામાં આવ્યું છે. અંગ્રેજી માધ્યમના પાઠ્યપુસ્તકનો આ ગુજરાતી અનુવાદ છે.

પ્રસ્તુત પાઠ્યપુસ્તકને વિષયવસ્તુલક્ષી, રસપ્રદ અને ક્ષતિરહિત બનાવવા માટે મંડળે પૂરતી કાળજી લીધી છે, તેમ છતાં શિક્ષણમાં રસ ધરાવનાર વ્યક્તિઓ પાસેથી પુસ્તકની ગુણવત્તા વધારે તેવાં સૂચનો આવકાર્ય છે.

ડૉ. ભરત પંડિત	**ડૉ. નીતિન પેથાણી**
નિયામક	કાર્યવાહક પ્રમુખ
તા.3-3-2015	ગાંધીનગર

પ્રથમ આવૃત્તિ : 2014; પુનઃમુદ્રણ : 2014

પ્રકાશક : ગુજરાત રાજ્ય શાળા પાઠ્યપુસ્તક મંડળ, 'વિદ્યાયન', સેક્ટર 10-એ, ગાંધીનગર વતી ડૉ. ભરત પંડિત, નિયામક

મુદ્રક :

મૂળભૂત ફરજો

ભારતના દરેક નાગરિકની ફરજ નીચે મુજબ રહેશે :*

(ક) સંવિધાનને વફાદાર રહેવાની અને તેના આદર્શો અને સંસ્થાઓનો, રાષ્ટ્રધ્વજનો અને રાષ્ટ્રગીતનો આદર કરવાની;

(ખ) આઝાદી માટેની આપણી રાષ્ટ્રીય લડતને પ્રેરણા આપનારા ઉમદા આદર્શોને હૃદયમાં પ્રતિષ્ઠિત કરવાની અને અનુસરવાની;

(ગ) ભારતનાં સાર્વભૌમત્વ, એકતા અને અખંડિતતાનું સમર્થન કરવાની અને તેમનું રક્ષણ કરવાની;

(ઘ) દેશનું રક્ષણ કરવાની અને રાષ્ટ્રીય સેવા બજાવવાની હાકલ થતાં, તેમ કરવાની;

(ચ) ધાર્મિક, ભાષાકીય, પ્રાદેશિક અથવા સાંપ્રદાયિક ભેદોથી પર રહીને, ભારતના તમામ લોકોમાં સુમેળ અને સમાન બંધુત્વની ભાવનાની વૃદ્ધિ કરવાની, સ્ત્રીઓના ગૌરવને અપમાનિત કરે, તેવા વ્યવહારો ત્યજી દેવાની;

(છ) આપણી સમન્વિત સંસ્કૃતિના સમૃદ્ધ વારસાનું મૂલ્ય સમજી તે જાળવી રાખવાની;

(જ) જંગલો, તળાવો, નદીઓ અને વન્ય પશુપક્ષીઓ સહિત કુદરતી પર્યાવરણનું જતન કરવાની અને તેની સુધારણા કરવાની અને જીવો પ્રત્યે અનુકંપા રાખવાની;

(ઝ) વૈજ્ઞાનિક માનસ, માનવતાવાદ અને જિજ્ઞાસા તથા સુધારણાની ભાવના કેળવવાની;

(ટ) જાહેર મિલકતનું રક્ષણ કરવાની અને હિંસાનો ત્યાગ કરવાની;

(ઠ) રાષ્ટ્ર પુરુષાર્થ અને સિદ્ધિનાં વધુ ને વધુ ઉન્નત સોપાનો ભણી સતત પ્રગતિ કરતું રહે એ માટે, વૈયક્તિક અને સામૂહિક પ્રવૃત્તિનાં તમામ ક્ષેત્રે શ્રેષ્ઠતા હાંસલ કરવાનો પ્રયત્ન કરવાની;

(ડ) માતા-પિતાએ અથવા વાલીએ ૬ વર્ષથી ૧૪ વર્ષ સુધીની વયના પોતાના બાળક અથવા પાલ્યને શિક્ષણની તકો પૂરી પાડવાની.

*ભારતનું સંવિધાન : કલમ 51-क

અનુક્રમણિકા

1. HTML નો પરિચય 1

2. HTMLમાં Head અને Body વિભાગ 18

3. HTMLમાં છબીઓનું વ્યવસ્થાપન 37

4. HTMLમાં યાદી અને કોષ્ટકનો ઉપયોગ 55

5. કેલ્કનો પરિચય 81

6. કેલ્કમાં ડેટાનું ઓર્ડરિંગ અને ફોર્મેટિંગ 97

7. કેલ્કમાં વિધેય 127

8. કેલ્કમાં આલેખની રચના 153

9. સમસ્યા અને સમસ્યાનું નિરાકરણ 175

10. સી ભાષાનો પરિચય 192

11. સી ભાષામાં ડેટા પ્રકાર, પ્રક્રિયકો અને પદાવલિઓ 216

12. નિવેશ(Input) / નિર્ગમ (Output) પ્રક્રિયાઓનો ઉપયોગ 239

13. નિર્ણય માળખાં 255

14. લૂપ નિયંત્રણ માળખાં 270

15. એરે 284

16. વિધેય 296

- પરિશિષ્ટ - I 309
- પરિશિષ્ટ - II 310
- પરિશિષ્ટ - III 311
- પરિશિષ્ટ - IV 312

આ પાઠ્યપુસ્તક વિશે...

પ્રિય શિક્ષકમિત્રો,

કમ્પ્યૂટર-સાક્ષરતાનો દ્રુતગતિએ ફેલાવો કરવાના ધ્યેય સાથે ગુજરાત સરકારે ICT@School કાર્યક્રમ હેઠળ 6000થી વધુ શાળાઓને અદ્યતન કમ્પ્યૂટર-સામગ્રી પૂરી પાડી છે. નવી નીતિની પહેલ તરીકે તમામ શાળાઓને ઉબન્ટુ (લિનક્સનું એક સ્વરૂપ) ઓપરેટિંગ સિસ્ટમ અને અન્ય ઑપન સોર્સ સૉફ્ટવેર પૅકેજ આપવામાં આવ્યાં છે, જેથી શાળાઓ લાઇસન્સ જેવી કોઈ પણ સમસ્યાઓની ચિંતા કર્યા વિના સૉફ્ટવેરનો ઉપયોગ કરી શકે કે તેને બદલી શકે. અગાઉનાં પાઠ્યપુસ્તકો મોટા ભાગે માલિકીહક ધરાવતાં સોફ્ટવેર પર આધારિત હોઈ નવા અભ્યાસક્રમ આધારિત પાઠ્યપુસ્તકો ફરી તૈયાર કરવાની આવશ્યકતા જણાઈ. આ ઉપરાંત ધોરણ 8ને પ્રાથમિક વિભાગમાં સ્થાનાંતરિત કરવામાં આવ્યું હોવાથી પણ આ અનિવાર્ય હતું. આથી, કમ્પ્યૂટર-અધ્યયનના વિવિધ વિષયો માટે ઑપન સોર્સ ઓપરેટિંગ સિસ્ટમ અને તેને સુસંગત ઑપન સોર્સ સૉફ્ટવેર ટૂલ્સ આધારિત નવું વિષયવસ્તુ ધોરણ 9થી12 માટે ક્રમાનુસાર પૂરું પાડવામાં આવ્યું છે.

ધોરણ 10નું આ પાઠ્યપુસ્તક 'કમ્પ્યૂટર-અધ્યયન' વિષયની શ્રેણીમાં બીજું પુસ્તક છે. ધોરણ-9માં વિદ્યાર્થીઓ ઑપન સોર્સ ઓપરેટિંગ સિસ્ટમ ઉબન્ટુ, ઑપન ઑફિસ શબ્દ પ્રક્રિયક રાઇટર, પ્રસ્તુતિ માટેનું ટૂલ ઈમ્પ્રેસ, ઈન્ટરનેટના ઉપયોગને લગતી માહિતીની શોધ, ઈ-મેઈલ, ફાઈલ ડાઉનલોડિંગ વગેરેના અભિગમો તથા સલામતીનાં પાસાં વિશે અભ્યાસ કરી ચૂક્યા છે. દસમા ધોરણના આ પાઠ્યપુસ્તકમાં તેમને શરૂઆતમાં વેબપેજ અને તેમાં માહિતીની રજૂઆત માટે HTML નામની મૂળભૂત માર્કઅપ ભાષાનો પરિચય કરાવવામાં આવ્યો છે. ઑપન ઑફિસમાં કેલ્સી નામના સ્પ્રેડશીટ ઘટકની પણ વિસ્તૃત ચર્ચા કરવામાં આવી છે. ત્યાર પછી તેઓ સમસ્યાના ઉકેલની પદ્ધતિઓ અને સી પ્રોગ્રામિંગ ભાષા વિશે અભ્યાસ કરશે. સી ભાષાની સરળ સમજૂતી માટે અનેક ઉદાહરણો તેનાં વિવરણ સહિત આપવામાં આવ્યાં છે. અમારું માનવું છે કે, પ્રોગ્રામિંગ ભાષાનો અભ્યાસ વિદ્યાર્થીની તર્કક્ષમતાનો વિકાસ કરવામાં અવશ્ય મદદરૂપ સાબિત થશે.

અમને આશા છે કે, આ પ્રયાસ ધોરણ 10ના વિદ્યાર્થીઓને ઉપયોગી નીવડશે અને આપ પણ ઑપન સોર્સ ઉબન્ટુ ઓપરેટિંગ સિસ્ટમ અને ટૂલ્સના ઉપયોગ દ્વારા પ્રાયોગિક કાર્ય અને શિક્ષણનો આનંદ મેળવશો.

પ્રિય વિદ્યાર્થીઓ,

કમ્પ્યૂટરને લગતી વિવિધ વિભાવનાઓ, ઓપરેટિંગ સિસ્ટમ ઉબન્ટુ તથા અન્ય પ્રચલિત ઓપરેટિંગ સિસ્ટમ, શબ્દ-પ્રક્રિયક અને પ્રસ્તુતિ માટેના ઑપન ઑફિસનાં ઘટકો, ઈન્ટરનેટ તથા તેના ઉપયોગથી તમે સુપરિચિત છો ત્યારે હવે પછીનો સમય એનાથી આગળ વધીને HTML, સ્પ્રેડશીટ કેલ્સી અને સી પ્રોગ્રામિંગ ભાષા જેવા અદ્યતન મુદ્દાઓનો અભ્યાસ કરવાનો છે.

ધોરણ 10ના આ પાઠ્યપુસ્તકનાં પ્રકરણ 1 થી 4માં માર્કઅપ ભાષા HTMLના અગત્યના મુદ્દાઓનો સમાવેશ કરવામાં આવ્યો છે. આ પ્રકરણોમાં HTMLના વિકાસ, HTML દસ્તાવેજનું માળખું અને દસ્તાવેજ બનાવવા માટે જરૂરી એવાં ટેગનાં ઉદાહરણ સહિત સંપૂર્ણ સમજૂતી આપવામાં આવી છે. સંબંધિત ઉદાહરણ સાથે Head અને Body ટેગની સમજૂતી તેમના ગુણધર્મો સાથે ફરી આપવામાં આવી છે. અંતમાં HTML કોડમાં ચિત્રો, યાદીઓ અને કોષ્ટકનો કેવી રીતે ઉપયોગ થઈ શકે તેની રજૂઆત અને સમજૂતી વિવિધ ઉદાહરણો દ્વારા આપવામાં આવી છે. પ્રકરણ 5 થી 8 માં ઑપનઑફિસના સ્પ્રેડશીટ માટેના ઘટક Calcનો સમાવેશ કરવામાં આવ્યો છે. કેલ્સીના અભ્યાસમાં વિગતોની સુધારણા (editing) અને સંરચના (formating), કેલ્સી વિધેયો અને આલેખ બનાવવાનો સમાવેશ કરવામાં આવ્યો છે. સમસ્યાના ઉકેલ અને પ્રોગ્રામિંગ અંગેનું પ્રાથમિક જ્ઞાન અત્યંત જરૂરી હોવાને કારણે સી પ્રોગ્રામિંગ ભાષાના ડેટાપ્રકારો, પ્રક્રિયકો, પદાવલિઓ, ઇનપુટ/આઉટપુટ પ્રક્રિયાઓ, નિર્ણયમાળખાં, ઍરે, લૂપનિયંત્રણ માળખાં અને સી વિધેયોનો સમાવેશ પ્રકરણ 9 થી 16 માં કરવામાં આવ્યો છે. આપેલ મુદ્દાઓનું વિવરણ કરતાં અનેક ઉદાહરણો યોગ્ય જગ્યાએ સમજૂતી સહિત રજૂ કરવામાં આવ્યાં છે.

વિદ્યાર્થીમિત્રો, અપેક્ષિત છે કે જો તમે આ પાઠ્યપુસ્તક અને તેમાં આપેલા પ્રાયોગિક સ્વાધ્યાયનો ધ્યાનપૂર્વક અભ્યાસ કરશો તો કમ્પ્યૂટર વિનિયોગોને સમજવા માટે જરૂરી એવા પ્રારંભિક છતાં મહત્ત્વના મુદ્દાઓ જેવા કે HTML, Calc અને C પ્રોગ્રામિંગ સાથેના કાર્યમાં પર્યાપ્ત વિશ્વાસ કેળવી શકશો.

HTMLનો પરિચય

ઈન્ટરનેટ અને HTMLની કાર્યપ્રણાલી

અનેકવિધ વિનિયોગમાં હવે કમ્પ્યૂટર બહોળા પ્રમાણમાં ઉપયોગમાં લેવાય છે. જ્યારે બે કે તેથી વધુ કમ્પ્યૂટરો એકબીજાં સાથે જોડાય છે ત્યારે તે સંશાધનોને વહેંચી શકે છે. આવાં એકબીજાં સાથે જોડાયેલ કમ્પ્યૂટરોના સમૂહને કમ્પ્યૂટર નેટવર્ક કહે છે. ઈન્ટરનેટ એ આવાં અનેક કમ્પ્યૂટર નેટવર્કનો સમૂહ છે અને તેથી જ તેને અનેક નેટવર્કનું નેટવર્ક કહે છે. ઈન્ટરનેટના પ્લૅટફૉર્મ પર એક વહેંચાયેલી માહિતી પદ્ધતિ અસ્તિત્વમાં છે, જેને વર્લ્ડ વાઇડ વેબ, WWW અથવા ટૂંકમાં વેબ કહે છે. વેબ અંગેનો પહેલો વિચાર 1991માં ટિમ બર્નર્સ-લી (આકૃતિ 1.1)ને આવ્યો હતો, જ્યારે તેઓ સ્વિટ્ઝરલૅન્ડ ખાતે CERN(યુરોપિયન પરમાણુ સંશોધન સંસ્થા, http://cern.web.cern.ch/CERN/)માં કામ કરતા હતા. વેબ એ ઈન્ટરનેટ પ્લૅટફૉર્મ પરની વિવિધ માધ્યમની (મલ્ટીમીડિયા) માહિતીનો ખજાનો છે. વેબપેઇજ સ્વરૂપે ઉપલબ્ધ વેબ-માહિતીને બ્રાઉઝર(વેબમાહિતીને મેળવી નિહાળવા માટેનો ખાસ વિનિયોગ)ના ઉપયોગ દ્વારા જોઈ શકાય છે. આ વેબપેઇજઓમાં વિભિન્ન સ્થાનો તરફ દર્શાવતાં જોડાણો (links) હોય છે. આ જોડાણોને હાઇપરલિન્ક (hyperlink) તરીકે ઓળખવામાં આવે છે. આવી હાઇપરલિન્ક પર ક્લિક કરીને કોઈ પણ વ્યક્તિ પોતાને ઇચ્છિત સ્થાન પર પહોંચાડી શકે છે. આ ક્રિયાને હાઇપરલિન્ક સાથે દોરવાવું કહેવાય છે. શાબ્દિક લખાણ, ચિત્ર, આકૃતિ વગેરે કોઈ પણ માહિતીને આવી હાઇપરલિન્ક સાથે સાંકળી શકાય છે. લખાણ અને હાઇપરલિન્કને વેબ પર સીધેસીધા રજૂ કરી શકાય નહીં. વેબ બ્રાઉઝર દ્વારા એક વેબપેઇજ કેવી રીતે દર્શાવાય તે વર્ણવવા માટે HTMLના ટૂંકા નામે ઓળખાતી હાઇપરટેક્સ્ટ માર્કઅપ લૅંગ્વેજની જરૂર પડે. આમ, HTML એ વેબપેઇજનું વર્ણન કરતી એક ભાષા છે. HTML એ વેબપેઇજના શીર્ષક, મથાળું, કોષ્ટક, છબી વગેરે જેવા લખાણને નિશાની આપવા માટેની દસ્તાવેજી ભાષા છે. તે યંત્ર સ્વતંત્ર છે અને તેથી HTML સંજ્ઞાના ઉપયોગ વડે લખાયેલ દરેક લખાણને તમામ ઇન્ટરનેટ બ્રાઉઝર સ્વીકારે છે.

આકૃતિ 1.1: ટિમ બર્નર્સ-લી

HTML એ એક પ્રકારની માર્કઅપ (સંજ્ઞાંકિત) ભાષા છે. માર્કઅપ ભાષા એ કેટલીક ચોક્કસ સંજ્ઞાઓ (ટૅગ)નો સમૂહ છે, જે વેબના લખાણને કેવી રીતે રજૂ કરવું તે બાબતે વધારાની માહિતી (વેબ માહિતી ઉપરાંત) ધરાવે છે. HTML ફાઈલ એ એવી ટેક્સ્ટ ફાઈલ છે કે જેમાં વેબ-માહિતીની સાથે સાથે ટૅગ સ્વરૂપે વધારાની ફૉર્મેટિંગ માર્કઅપ માહિતી ધરાવે છે. HTML એ અતિપ્રચલિત માર્કઅપ ભાષા છે અને તે ટૅગનો ચોક્કસ સમૂહ ધરાવે છે. HTML એ SGML (Standardized General Markup Language) પરથી ઊતરી આવી છે, જે 1986માં આંતરરાષ્ટ્રીય માનક સંસ્થા (ISO) દ્વારા મશીન વાંચી શકે તેવા દસ્તાવેજોના આદાન-પ્રદાનની સુવિધા માટે તૈયાર કરવામાં આવી હતી.

આમ, HTML કોડ એ બ્રાઉઝર દ્વારા વેબપેઇજ પર દર્શાવવાની માહિતી અને આ માહિતીની રજૂઆતને મદદરૂપ ટૅગનું સંયોજન છે. આવા આધારસ્તંભરૂપી કોડ વગર વેબપેઇજ પર માહિતી દર્શાવવાનું અશક્ય છે. આથી વેબપેઇજ તૈયાર કરવા, તેનું અર્થઘટન કરવા અને રજૂઆત કરવા HTML કોડ લખવા ફરજિયાત બને છે.

એક સાદો HTML દસ્તાવેજ

તો ચાલો, HTMLનો ઉપયોગ કરી એક સાદું વેબપેઇજ તૈયાર કરીએ, જે RAINBOW વિશે ચર્ચા કરે. વેબપેઇજ પર રજૂ કરવાની માહિતી કોષ્ટક 1.1માં દર્શાવેલ છે.

RAINBOW

Rainbow consists of seven colours. These colours are Violet, Indigo, Blue, Green, Yellow, Orange and Red. They are also acronymed as VIBGYOR.

Rainbow is caused by reflection of light in water droplets in the Earth's atmosphere, resulting in a spectrum of light appearing in the sky. It takes the form of a multi coloured arc.

કોષ્ટક 1.1 : HTML વડે વેબપેઈજ પર રજૂ કરવાનું લખાણ

કોષ્ટક 1.1માં દર્શાવેલ લખાણને દર્શાવવા માટેના HTML કોડ કોડ-લિસ્ટિંગ 1.1માં આપેલ છે.

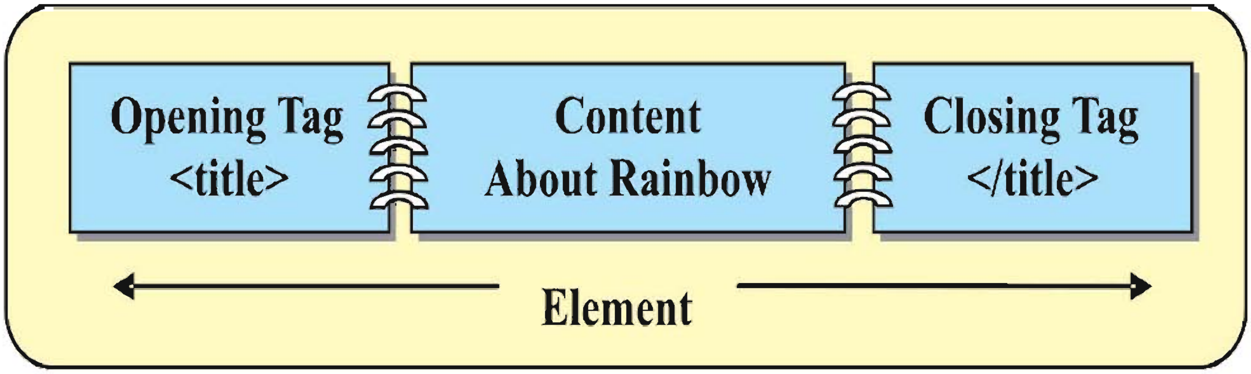
```
<html>
    <head>
        <title> About Rainbow
        </title>
    </head>
    <body>
        <h1> RAINBOW </h1>
       <p>   Rainbow consists of seven colours. These colours
        are Violet, Indigo, Blue, Green, Yellow, Orange and
        Red. They are also acronymed as VIBGYOR.
        </p>
        <p>  Rainbow is caused by reflection of light in water
             droplets in the Earth's atmosphere, resulting in a
             spectrum of light appearing in the sky. It takes the
             form of a multi coloured arc.
        </p>
    </body>
</html>
```

કોડ-લિસ્ટિંગ 1.1

અહીં એ ખાસ અવલોકન કરો કે કોડ-લિસ્ટિંગ 1.1માં rainbowના લખાણ ઉપરાંત કેટલાક ખૂણિયા કૌંસ અને તે દરેકમાં કોઈ ચોક્કસ અક્ષર કે શબ્દો પણ દર્શાવાયા છે. આ ખૂણિયા કૌંસ અને તેની અંદરના શબ્દોને ટૅગ કહે છે.

ટૅગ એ બે બાજુ ખૂણિયા કૌંસથી આવરેલ અક્ષરો, શબ્દો અને આંકડાઓનો બનેલો હોય છે. શરૂ થતા ટૅગ અને બંધ થતા ટૅગની વચ્ચે જરૂરી લખાણને એકસાથે એક એલિમેન્ટ (Element) કહે છે. HTML એલિમેન્ટ ખાલી હોય

અથવા માહિતી દર્શાવવા માટે ગોઠવણી અને પ્રકાશન સંબંધિત વધારાની સૂચનાઓ પણ હોઈ શકે. આકૃતિ 1.2 ઉદાહરણ સાથે ટૅગનું બંધારણ અને એલિમેન્ટ દર્શાવે છે.



આકૃતિ 1.2 : HTML એલિમેન્ટ

આકૃતિ 1.2 એવું દર્શાવે છે કે, **<title> About Rainbow </title>** એ HTML કોડનો એલિમેન્ટ છે. અહીં એ નોંધ લો કે બંધ થતા ટૅગમાં શબ્દની આગળ સ્લેશ **</title>** મૂકવામાં આવ્યો છે. આ ટૅગ એલિમેન્ટનો અંત દર્શાવે છે. ટૅગની જોડ અને તેની વચ્ચેનું લખાણ ભેગું મળીને, HTML એલિમેન્ટ બનાવે છે. અહીં Title ટૅગ HTML કોડ વડે બનતા વેબપેઇજનું શીર્ષક નક્કી કરે છે.

આવા અન્ય એક HTML એલિમેન્ટનું ઉદાહરણ નીચે મુજબ છે :

<h1> RAINBOW </h1>

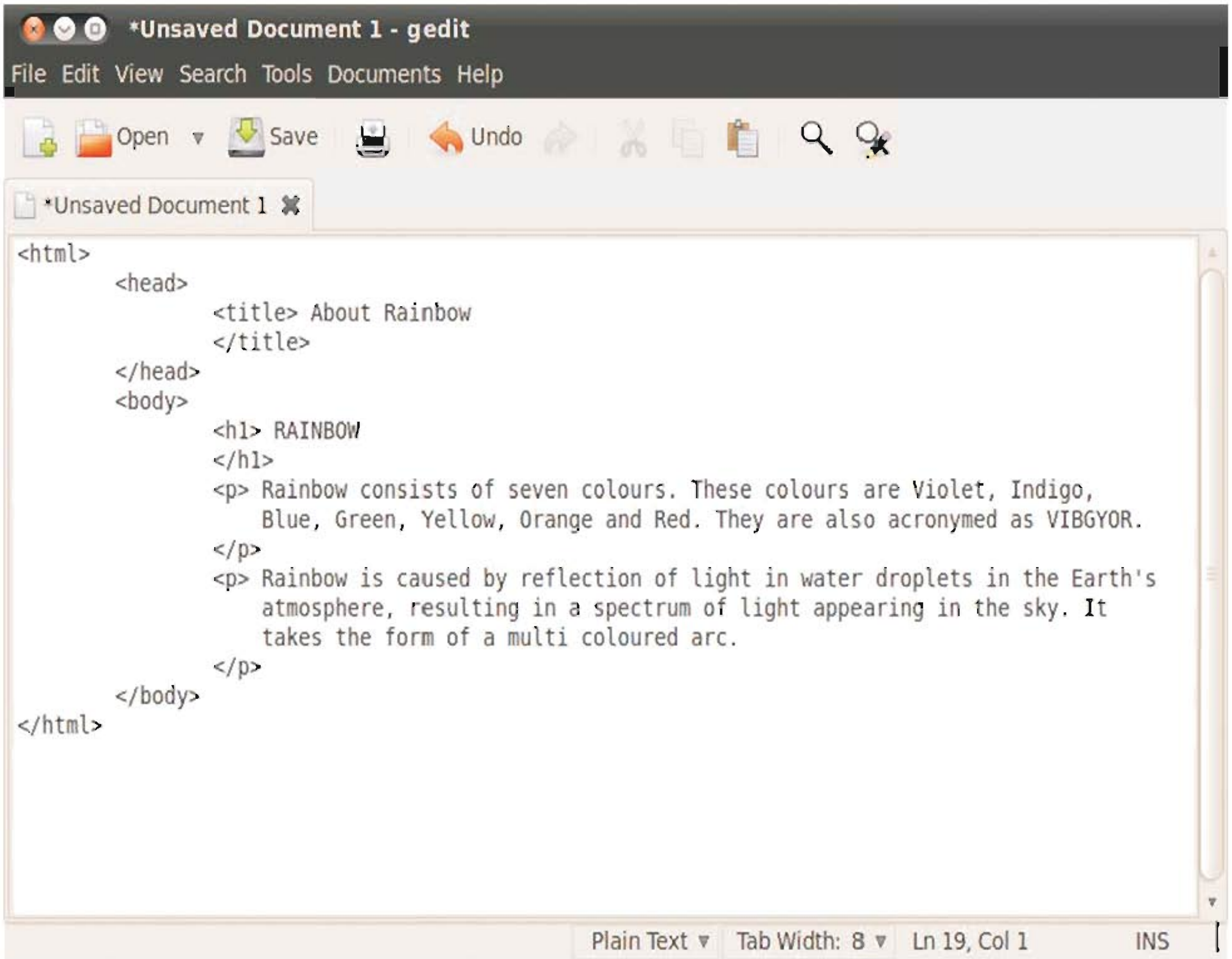
અહીં h1 ટૅગ મથાળા માટે છે.

ઉપર દર્શાવ્યા મુજબ <h1> અને </h1> ની વચ્ચેનું લખાણ મથાળા તરીકે ઓળખવામાં આવશે અને મથાળા તરીકે રજૂ કરવામાં આવશે. એ જ રીતે, <p> અને </p> ટૅગ વચ્ચે ઉપલબ્ધ લખાણ ફકરા તરીકે ગણવામાં આવશે અને જુદા ફકરા તરીકે રજૂ કરવામાં આવશે. આખો દસ્તાવેજ <html> અને </html> ટૅગમાં આવરીને દર્શાવાશે.

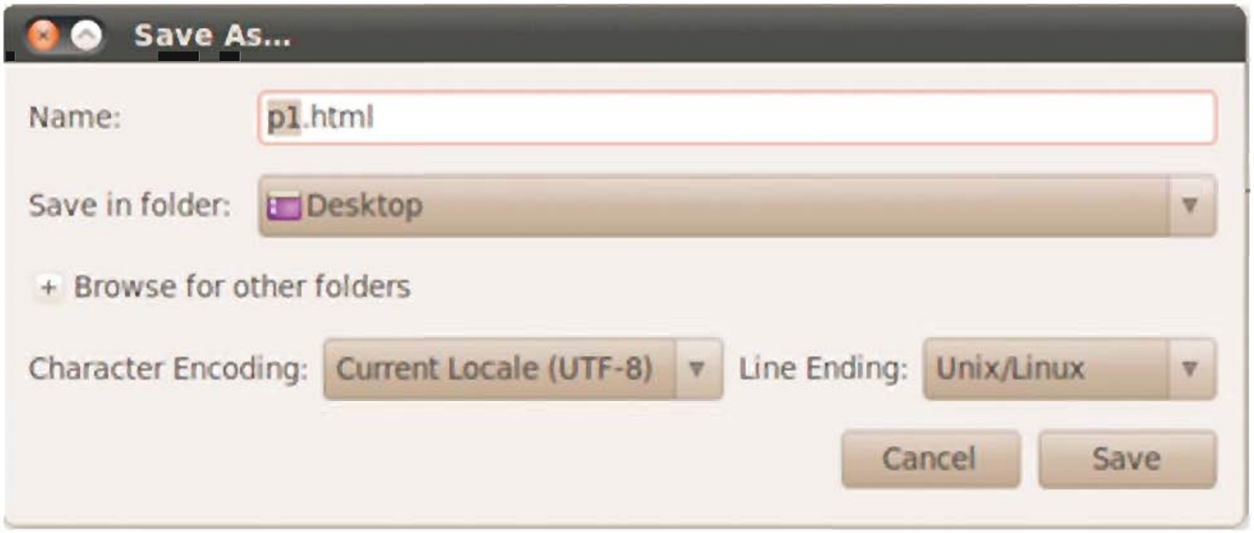
બ્રાઉઝર દ્વારા આ પાનું કેવું દર્શાવાશે તે નિહાળવા નીચે આપેલાં પગલાં અનુસરો :

પગલું 1 : Applications → Accessories → gedit પસંદ કરી gedit નામનો એડિટર પ્રોગ્રામ ખોલો. gedit એ GNOME (GNU નામે ઓળખાતા પ્રોજેક્ટનો ભાગ, MIT દ્વારા બનાવાયેલ મુક્ત સૉફ્ટવેર), ડેસ્કટોપ એન્વાયર્નમેન્ટ, Mac OS X અને માઇક્રોસૉફ્ટ વિન્ડોઝ માટેનું સામાન્ય હેતુ માટેનું ટેક્સ્ટ એડિટર છે. એના બદલે, તમે સ્ક્રીનને મથાળે ઉપલબ્ધ હરોળમાં gedit એડિટરના શૉર્ટકટનો ઉપયોગ પણ કરી શકો.

પગલું 2 : ખાલી gedit વિન્ડોમાં કોડ-લિસ્ટિંગ 1.1માં આપેલ HTML કોડ-લિસ્ટિંગ ટાઇપ કરો. આ કોડ ટાઇપ કરી દીધા પછીનો gedit એડિટરનો દેખાવ આકૃતિ 1.3માં દર્શાવેલ છે. એડિટરના મથાળાની લાઇન પર આપેલ Save વિકલ્પ પસંદ કરીને "p1.html" તરીકે સંગ્રહ કરો. અહીં એ બાબતની નોંધ લો કે, HTML ફાઈલ htm અથવા html અનુલંબન સાથે સાચવી શકાય છે. આકૃતિ 1.4 કોડને સાચવવા માટેની રીત દર્શાવે છે.



આકૃતિ 1.3 : gedit એડિટરમાં લખેલ HTML કોડ



આકૃતિ 1.4 : Save As... ડાયલોગ બૉક્સ

પગલું 3 : Mozilla Firefox કે તમારા કમ્પ્યૂટરમાં પ્રસ્થાપિત કરેલ અન્ય કોઈ પણ બ્રાઉઝરને ખોલો. File → Open File પસંદ કરો. તમને Open File ડાયલોગ બૉક્સ દર્શાવાશે. તમારે જે ફાઈલને ખોલવી હોય તે પસંદ કરી Open બટન પર ક્લિક કરો. એના વિકલ્પરૂપે તમે ફાઈલ પર ડબલ ક્લિક પણ કરી શકો. ફાઈલને જો Mozilla Firefoxમાં ખોલવામાં આવે તો તેનો દેખાવ અને અસર આકૃતિ 1.5 દર્શાવે છે.

આકૃતિ 1.5 : HTML કોડનો ઉપયોગ કરી બ્રાઉઝર દ્વારા દર્શાવાતું વેબપેઈજ

અત્રે એ ખાસ નોંધવું જોઈએ કે, HTML કોડમાં ઉપયોગમાં લેવાતા ટૅગ કેસ-સેન્સિટિવ નથી. એટલે કે, આપણે ટૅગને મોટા (અપર કેસ) કે નાના (લોઅર કેસ) અક્ષરોમાં અથવા બન્નેનું મિશ્રણ કરીને પણ લખી શકીએ. વધુમાં, એકથી વધુ એલિમેન્ટ એક જ લીટીમાં પણ લખી શકાય છે. જોકે, બને ત્યાં સુધી દરેક એલિમેન્ટને નવી લીટી પર લખવા જોઈએ અને વાંચવામાં સરળતા રહે તે માટે યોગ્ય ઇન્ડેન્ટ પાડીને લખવું જોઈએ.

HTML દસ્તાવેજનું માળખું

HTML દસ્તાવેજ મુખ્યત્વે બે ભાગમાં વહેંચાયેલું હોય છે : પ્રથમ મથાળા વિભાગ (Head section) હોય છે અને બીજો બૉડી વિભાગ (Body section). તેને ક્યારેક હેડ એલિમેન્ટ (Head Element) અને બૉડી એલિમેન્ટ (Body Element) તરીકે પણ ઓળખવામાં આવે છે. મથાળા વિભાગમાં પાના વિશેની માહિતી જેવી કે, શીર્ષક (title) અને પાનાનું વર્ણન (description) ધરાવે છે. આ બધી જ માહિતી <head> અને </head> ટૅગની વચ્ચે આવરીને દર્શાવવી પડે છે.

બૉડી વિભાગ <body> અને </body> ટૅગની વચ્ચે આવરીને દર્શાવાય છે. આ એ વિગત છે જે બ્રાઉઝર મારફત જોઈ શકાય છે. મથાળા વિભાગ અને બૉડી વિભાગ એ બન્ને <html> અને </html> ટૅગની વચ્ચે આવરીને દર્શાવાય છે.

HTML શીર્ષક

વેબપેઇજનું શીર્ષક TITLE એલિમેન્ટ દ્વારા રજૂ કરાય છે, જે દસ્તાવેજના મથાળા વિભાગમાં મૂકવામાં આવે છે. અત્રે એ ખાસ નોંધનીય છે કે કોઈ પણ દસ્તાવેજને માત્ર એક જ title એલિમેન્ટ હોય છે. સામાન્ય રીતે દસ્તાવેજના લખાણને ઓળખવા માટે તે ઉપયોગી છે. વળી, titleનું લખાણ દસ્તાવેજના લખાણનો ભાગ નથી. આ કારણોસર આ લખાણ સાદું લખાણ હોવું જોઈએ અને તેમાં હાઈપરલિન્ક જેવા વિશેષ કમાન્ડ ધરાવતા હોવા ન જોઈએ. આ Title એ લખાણને દર્શાવતી વિન્ડોના શીર્ષક તરીકે દેખાશે. Bookmark list અથવા બ્રાઉઝરના ઇતિહાસ(history)માં પણ આ title નું સ્થાન રહે છે. અને તેથી title બને તેટલાં ટૂંકાં હોવાં જોઈએ તેવી ભલામણ કરવામાં આવે છે. કોડ-લિસ્ટિંગ 1.1ના કોડ-લિસ્ટિંગ મુજબ આપણા ઉદાહરણના HTML કોડ માટે "About Rainbow" એ શીર્ષક છે. આકૃતિ 1.5માં દર્શાવ્યા મુજબ તે દર્શાવાતી વિન્ડોના ઉપરના ભાગે દર્શાવાય છે.

HTML હેડિંગ સ્ટાઈલ

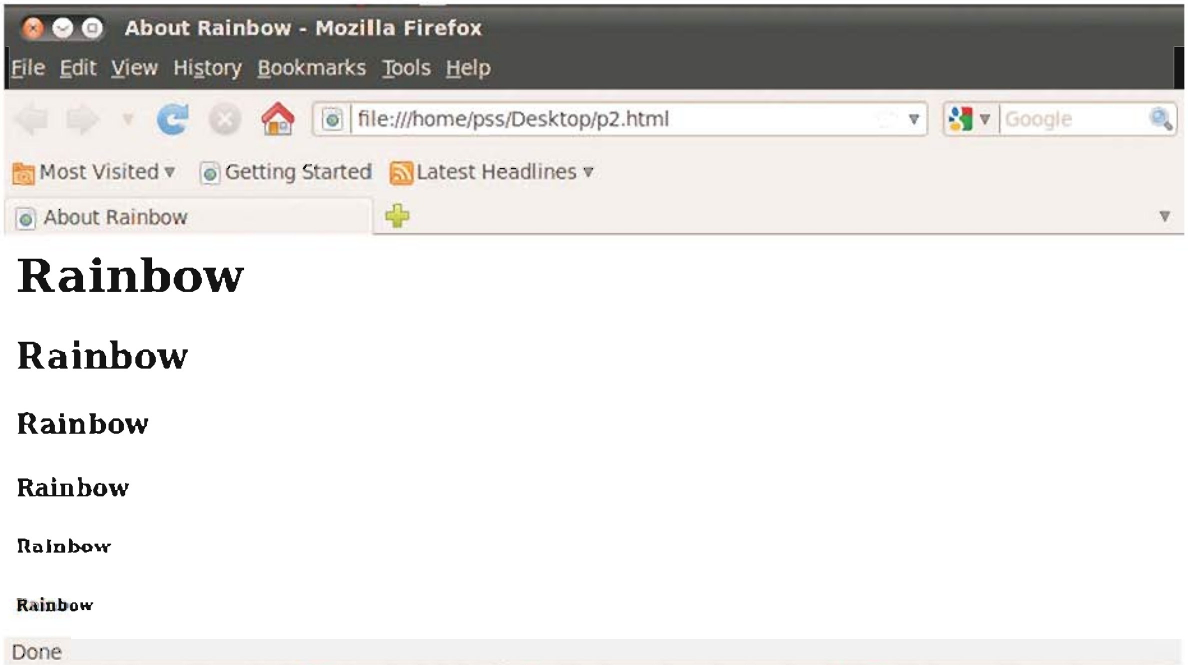
HTML દસ્તાવેજ સામાન્ય રીતે મથાળા સાથે શરૂ થાય છે. કોડ-લિસ્ટિંગ 1.1માં દર્શાવેલ ઉદાહરણ મુજબ મથાળા માટે પ્રથમ પ્રકારની સ્ટાઈલ (h1) ટૅગ ઉપયોગમાં લેવાઈ છે. મથાળાની પ્રથમ પ્રકારની સ્ટાઈલ (h1) કેવી દેખાય છે તે જોવા માટે આકૃતિ 1.5માં દર્શાવેલ મથાળા RAINBOWને જુઓ. HTMLમાં આવી વધુ પાંચ પ્રકારની મથાળાની સ્ટાઈલ

ઉપલબ્ધ છે. કુલ છ પૂર્વપ્રસ્થાપિત સ્ટાઇલ અનુસાર મથાળું દર્શાવી શકાય છે, જે અનુક્રમે h1, h2, h3, h4, h5 અને h6 નામથી ઓળખાય છે. મથાળાના આ છ સ્તર આકૃતિ 1.6માં આપેલ HTML કોડમાં વર્ણવેલ છે.

આકૃતિ 1.6 : HTMLમાં જુદા જુદા સ્તરના મથાળા માટેનો HTML કોડ

gedit એડિટરનો ઉપયોગ કરીને આ કોડ ટાઇપ કરવામાં આવ્યા છે. આ કોડનો "p2.html" નામથી સંગ્રહ કરો. જ્યારે આ કોડ આપણે બ્રાઉઝર દ્વારા નિહાળીશું ત્યારે તે આકૃતિ 1.7માં દર્શાવ્યા મુજબનું દૃશ્ય રજૂ કરશે.

મોટા ભાગનાં બ્રાઉઝર <h1>, <h2> અને <h3>નાં લખાણને દસ્તાવેજના લખાણના સામાન્ય કદ કરતાં મોટા કદમાં દર્શાવે છે. <h4> એલિમેન્ટના લખાણનું કદ લખાણના સામાન્ય કદ જેટલું જ દર્શાવાશે. જોકે, કોઈ વ્યક્તિ આ મથાળાના કદને હંમેશાં પોતાની રીતે પુનઃનિર્ધારિત કરી શકે છે.



આકૃતિ 1.7 : HTMLમાં મથાળાના જુદા જુદા સ્તરનું પરિણામ

લખાણની સુઘડતા વધારતા ટૅગ

h1 થી h6 જેવી મથાળાની વિવિધ સ્ટાઇલ ઉપરાંત HTMLમાં બીજી ઘણી એવી ટૅગ લખાણના ફૉર્મેટિંગ માટે ઉપલબ્ધ છે. આ ટૅગને HTML કોડના બૉડી વિભાગમાં ઉપયોગમાં લઈ શકાય છે. આવા કેટલાક મૂળભૂત ટૅગ નીચે મુજબ છે :

ફકરા માટેના <p> અને </p> ટૅગ

<p> ટૅગ લખાણને ફકરા સ્વરૂપે ગોઠવે છે. લખાણના દરેક ફકરાને <p> અને </p> ટૅગની વચ્ચે મૂકવો જોઈએ. નીચે આવાં ત્રણ યોગ્ય ઉદાહરણ આપેલ છે :

```
<p> This is first paragraph. </p>

<p> The second paragraph is here. This is about multiple
colours of rainbows.
Though it is a temporary event it leaves a great impact
on human mind. </p>
<p> Best of Luck! </p>
```

ઉપર દર્શાવેલ ઉદાહરણો પૈકી બીજું ઉદાહરણ ધ્યાને લો. તમે એ જોઈ શકો છો કે, બીજા ફકરામાં અનેક ખાલી જગ્યા અને કૅરેજ રિટર્ન (જ્યાં Enter કી દબાવી છે અને લખાણ નવી લીટી પર આવી ગયું છે.) જણાય છે. અહીં એ નોંધનીય છે કે, આવી ખાલી જગ્યા(white space)ને એક જ ખાલી જગ્યા તરીકે ગણવામાં આવશે. જ્યારે HTML કોડ બ્રાઉઝરમાં દર્શાવાશે ત્યારે લખાણ આપમેળે સ્ક્રીનની સંપૂર્ણ પહોળાઈ લઈ લેશે. જો તમે બ્રાઉઝરની વિન્ડોનું કદ બદલશો તો બ્રાઉઝર લખાણને નવી લાઇનમાં લઈ જશે. આવી ખાલી જગ્યા (white space)ની વ્યવસ્થા HTML કોડ તૈયાર કરનારને લખેલ કોડ સારી રીતે વાંચી શકાય તે માટે કોડને ઇન્ડેન્ટ પાડીને દર્શાવવાની તેમજ કોડની વચ્ચે જરૂરી વધારાની ખાલી જગ્યા ઉમેરવાની છૂટ આપે છે.

નવી લીટી પરથી શરૂઆત માટેઃ
 અથવા

અન્ય ટૅગ કરતાં ઊલટું,
 એલિમેન્ટને શરૂઆતનો અને અંતનો ટૅગ હોતો નથી.
 એ breakનું ટૂંકું નામ છે. આવા ટૅગ ખાલી (Empty) ટૅગ તરીકે પણ ઓળખાય છે. XHTML જેવી અદ્યતન આવૃત્તિ
 ટૅગનો ઉપયોગ કરે છે. અહીં એ નોંધ લો કે, 'br' અને '/'ની વચ્ચે જગ્યા હોવી જરૂરી છે.

 તેના પછી આવતા લખાણને નીચેની લીટી પર લઈ જાય છે. જો તમારે એકથી વધુ ખાલી લીટીઓ છોડવી હોય તો જરૂર મુજબ એકથી વધુ
 ટૅગ મૂકો.
નો ઉપયોગ દર્શાવતું ઉદાહરણ નીચે મુજબ છે :

```
First example <br> <br>

Second example
```

ફકરામાં
નો ઉપયોગ કરવાથી લખાણમાં ફરજિયાત બ્રેક લાવે છે, અને રજૂઆત વખતે તેમજ બ્રાઉઝર વિન્ડોનું કદ બદલતી વખતે લખાણને વાળીને (wrap કરીને) નવી લાઇન પર લઈ જવામાં તકલીફ પડે છે.

પ્રિ-ફૉર્મેટેડ લખાણ

ઘણી વાર આપણે આપણા લખાણને બ્રાઉઝર બદલે નહીં તેવી રીતે એકથી વધુ ખાલી જગ્યા (white space) સાથે એકથી વધુ લીટીઓમાં દર્શાવાય એવું ઇચ્છીએ છીએ. આ માટે આપણે આપણા આવા લખાણને <pre> અને </pre> નામના બે પ્રિ-ફૉર્મેટેડ ટૅગમાં આવરીને દર્શાવી શકીએ. <pre> સાથે શરૂ થતાં અને </pre> સાથે પૂરા થતા કોઈ પણ લખાણની ગોઠવણી જેમની તેમ જળવાઈ રહે છે. ઉદાહરણ નીચે મુજબ છે :

<pre>

This is first line.

This is second line.

This is third line.

</pre>

બ્રાઉઝરમાં દર્શાવાતું લખાણ આપેલ તમામ ઇન્ડેન્ટ (indentation) સાથે દેખાય છે. નીચેનાં ઉદાહરણનો પણ પ્રયોગ કરી જુઓ :

**Bold: and **

નક્કી કરેલ લખાણને ઘાટા અક્ષરોમાં છાપવા માટે આ ટેગનો ઉપયોગ કરાય છે. આનું ઉદાહરણ નીચે મુજબ રજૂ કરી શકાય :

<p> This is the first paragraph. </p>

અહીં "first" શબ્દને ઘાટા અક્ષરોમાં દર્શાવવામાં આવશે.

Underline: <u> અને </u>

લખાણમાંના અમુક અગત્યના શબ્દ નીચે લીટી દોરેલી દર્શાવાય તે માટે આ ટેગ ઉપયોગમાં લઈ શકાય છે, જેનું ઉદાહરણ નીચે મુજબ આપી શકાય :

<p> This is the <u> first </u> paragraph. </p>

અહીં, first શબ્દની નીચે લીટી સાથે દર્શાવાશે.

Italics: <i> and </i>

લખાણમાંના અમુક શબ્દોને ત્રાંસા (ઇટાલિક્સ ઢબે) છાપવા માટે આ ટેગનો ઉપયોગ કરવામાં આવે છે, જેનું ઉદાહરણ નીચે મુજબ આપી શકાય :

<p> This is the <i> first </i> paragraph. </p>

અહીં first શબ્દ ત્રાંસા અક્ષરો સાથે છપાશે.

છેકી નાંખેલા શબ્દો: <s> અને </s>

<s> અથવા <strike> એલિમેન્ટના લખાણના શબ્દો લખ્યા પછી છેકી નાંખ્યા હોય તે રીતે શબ્દો પર પાતળી લીટી સાથે છપાય છે. અહીં s એ strike શબ્દનું ટૂંકું નામ છે. છેકી નાંખેલા શબ્દોને રજૂ કરતું ઉદાહરણ નીચે મુજબ છે :

<p> This is the <s> cancelled </s> paragraph. </p>

અહીં cancelled લખાણ છેકી નાંખેલા શબ્દો જેવું છપાશે.

ટાઇપરાઇટર ફૉન્ટ: <tt> અને </tt>

<tt> એલિમેન્ટનું લખાણ ટાઇપરાઇટર જેવા અક્ષરોમાં લખાય છે, જેને 'મૉનો-સ્પેસ્ડ ફૉન્ટ' (ટેલિ-ટાઇપ મશીન જેવા) તરીકે પણ ઓળખાય છે, જેનું ઉદાહરણ નીચે મુજબ છે :

<p> This is the <tt> first </tt> paragraph. </p>

અહીં first શબ્દ મૉનો-સ્પેસ્ડ ફૉન્ટમાં દર્શાવાશે.

આવા કેટલાક અન્ય એલિમેન્ટ પણ છે, જે કોષ્ટક 1.2માં દર્શાવેલ છે.

એલિમેન્ટ	વર્ણન
અને	આ એલિમેન્ટનું લખાણ તેની આસપાસના લખાણ કરતાં એક ફોન્ટ સાઇઝ જેટલું નાનું દર્શાવાય છે.
<big> અને </big>	આ એલિમેન્ટનું લખાણ તેની આસપાસના લખાણ કરતાં એક ફોન્ટ સાઇઝ જેટલું મોટું દર્શાવાય છે.
^{અને}	આ એલિમેન્ટનું લખાણ તેની આગળના લખાણ કરતાં નાનું અને તેના કરતાં થોડું ઊંચે (સુપરસ્ક્રિપ્ટ અક્ષર તરીકે) છપાય છે.
_{અને}	<sup> કરતાં ઊલટું, આ એલિમેન્ટનું લખાણ તેની આગળના લખાણ કરતાં નાનું અને તેના કરતાં થોડું નીચે (સબસ્ક્રિપ્ટ અક્ષર તરીકે) છપાય છે.
<acronym> અને </acronym>	આ એલિમેન્ટનું લખાણ ટૂંકા નામ (acronym) તરીકે દર્શાવાય છે.
<dfn> અને </dfn>	તે એક ચોક્કસ પદને વ્યાખ્યાયિત કરે છે.
<q> અને </q>	તે અવતરણ ચિહ્ન (quote) દર્શાવે છે.

કોષ્ટક 1.2 : કેટલાંક અન્ય ફોર્મેટિંગ ટૅગ

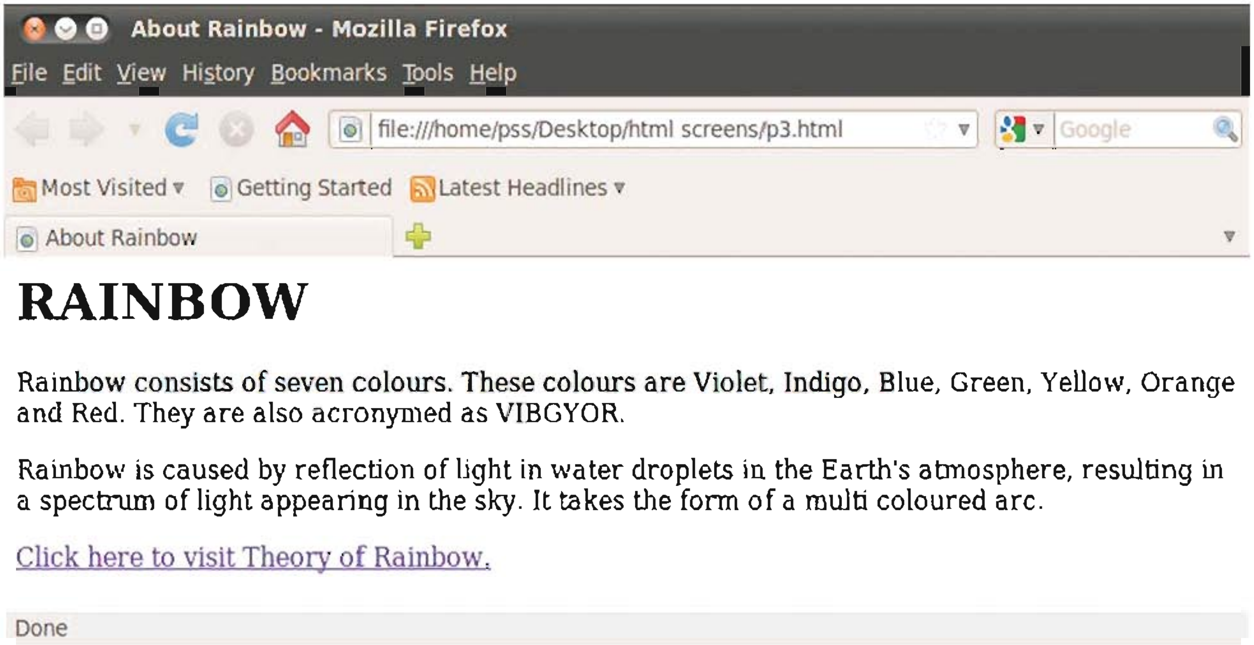
Anchor Tag

જ્યારે કોઈ લખાણ HTML દસ્તાવેજની અંદર જ દર્શાવાય છે, ત્યારે લખાણ અને તેના સ્વરૂપને નિયંત્રિત કરતા અક્ષરો (ટૅગ) ઉપરાંત અન્ય લખાણને સંબોધતી વધારાની માહિતી પણ જરૂરી બને છે. ઘણી વખત વિશેષ ખુલાસો પણ જરૂરી હોય છે. આવા કેટલાક અક્ષરો કે શબ્દોનો સમૂહ કે જે અલગ રંગથી (સામાન્ય રીતે ભૂરા રંગથી) અને નીચે લીટી દોરીને દર્શાવાય છે તેને હાઇપરલિન્ક તરીકે ઓળખાય છે. હાઇપરલિન્ક <a> નામના ટૅગ દ્વારા બનાવવામાં આવે છે, જેમાં 'a'નો અર્થ anchor છે. તો આવો, "p1.html"ને કોડ-લિસ્ટિંગ 1.2માં બતાવ્યા પ્રમાણે સુધારીએ.

```
<html>
    <head>
        <title> About Rainbow
        </title>
    </head>
    <body>
        <h1> RAINBOW </h1>
       <p>  Rainbow consists of seven colours. These colours are Violet,
            Indigo, Blue, Green, Yellow, Orange and Red. They are also
            acronymed as VIBGYOR.
        </p>
        <p> Rainbow is caused by reflection of light in water droplets in
            the Earth's atmosphere, resulting in a spectrum of light appearing
            in the sky. It takes the form of a multi coloured arc.
        </p>
        <p>
         <a href= "p4.html" > Click here to visit Theory of Rainbow. </a>
    </p>
</body>
</html>
```

કોડ-લિસ્ટિંગ 1.2 : હાઇપરલિન્કનો ઉપયોગ દર્શાવતો HTML કોડ

ફાઈલને "p3.html" તરીકે સાચવો. જ્યારે આ કોડને બ્રાઉઝર દ્વારા નિહાળવામાં આવશે ત્યારે તે આકૃતિ 1.8 મુજબ દર્શાવાશે.



આકૃતિ 1.8 : p3.htmlનું પરિણામ

જ્યારે ઉપયોગકર્તા હાઈપરલિન્ક પર ક્લિક કરે ત્યારે 'Theory of Rainbow' વર્ણવતી નિર્ધારિત ફાઈલ ખુલવી જોઈએ. તો ચાલો, હાઈપરલિન્ક દ્વારા નિર્દિષ્ટ ફાઈલ માટે કોડલિસ્ટિંગ 1.3માં દર્શાવ્યા મુજબ HTML કોડ બનાવીએ.

```
<html>
    <head>
        <title> Theory of Rainbow
        </title>
    </head>
    <body>
        <h1> How Rainbow Developed </h1>
        <p>  Rainbow is caused by reflection of light in water droplets
             in the Earth's atmosphere, resulting in a spectrum of light
             appearing in the sky. It takes the form of multi-coloured arc.
        </p>
    </body>
</html>
```

કોડ-લિસ્ટિંગ 1.3 : હાઈપરલિન્ક સાથે જોડાયેલ માહિતી માટેનો HTML કોડ

કોડ-લિસ્ટિંગ 1.3માં દર્શાવેલ HTML કોડનો "p4.html" તરીકે સંગ્રહ કરો. જ્યારે તમે આકૃતિ 1.8માં દર્શાવ્યા મુજબના સાંકળેલ (anchor) લખાણ (Click here to visit Theory of Rainbow) પર ક્લિક કરશો તો તરત જ p4.html ફાઈલનું લખાણ બ્રાઉઝરમાં આવી જશે. (જુઓ આકૃતિ 1.9.)

આકૃતિ 1.9 : p4.htmlનું લખાણ

હાઈપરલિન્ક એ દસ્તાવેજમાં હાઈપર ટેક્સ્ટ (hyper text) બનાવીને તેને અન્ય દસ્તાવેજ સાથે સાંકળીને બે HTML દસ્તાવેજને જોડવાની રીત છે. એક વેબસાઈટ, કે જે ઘણાં બધાં વેબપેઈજનો સમૂહ હોય છે, તે આવા હાઈપરલિન્ક સંચાલન મારફત સંચાલિત થાય છે. અહીં એ નોંધનીય છે કે, આપણે બનાવેલાં વેબપેઈજ, તેના HTML કોડમાં આપણે જે રીતે દર્શાવેલ છે તે પ્રમાણે લખાણને રજૂ કરે છે. આવાં વેબપેઈજ દર્શાવેલ માહિતીને નિયત (static કે fixed) સ્વરૂપે જ રજૂ કરતા હોઈ તેને ‘‘સ્ટેટિક વેબપેજ’’ તરીકે ઓળખવામાં આવે છે.

પૂર્ણ અને સંબંધિત સરનામું

કોડ-લિસ્ટિંગ 1.2માં દર્શાવેલ Click here to visit Theory of Rainbow. ને જુઓ.

http://www.somedomain.com/p4.html એવું પૂર્ણ સરનામું આપવાને બદલે આપણે માત્ર ફાઈલનું નામ "p4.html" જ આપ્યું છે. આ રીતે માત્ર ફાઈલનું નામ જ દર્શાવવાનું ત્યારે જ શક્ય છે કે જ્યારે આપણી p3.html ફાઈલ કે જેમાંથી અન્યને બોલાવવામાં આવે છે તે. (આવી ફાઈલને ‘કોલિંગ ફાઈલ’ કે ‘પેરન્ટ ફાઈલ’ કહે છે.) અને "p4.html" જેને બોલાવવામાં (call કરવામાં) આવે છે, (આવી ફાઈલને ‘રીફર્ડ ફાઈલ’ કે ‘કોલ ફાઈલ’ કહે છે.) તે બન્ને એક જ ડિરેક્ટરીમાં ઉપલબ્ધ હોઈ બોલાવવામાં આવતી (called) ફાઈલનું સ્થાન એ બોલાવનાર ફાઈલને સંબંધિત હોય છે અને એ રીતે એને સંબંધિત સરનામા (Relative Address) તરીકે ઓળખાય છે. જ્યારે સંપૂર્ણ સરનામાને પૂર્ણ સરનામા (absolute address) તરીકે ઓળખવામાં આવે છે. જો બોલાવવામાં આવતી ફાઈલ એક ડિરેક્ટરી ઉપર આવેલી હોય તો આપણે ફાઈલ નામની આગળ ../ લગાવી શકીએ.

નોંધ :

જો ફાઈલના નામની આગળ કોઈ સરનામું આપવામાં ન આવ્યું હોય તો બ્રાઉઝર એવું માની લેશે કે બોલાવવામાં આવેલી ફાઈલ બોલાવનાર (પેરન્ટ) ફાઈલની ડિરેક્ટરીમાં જ છે.

ટૅગના ગુણધર્મો

ટૅગની સાથે વધુ માહિતીને દર્શાવવા ટૅગની સાથે વધારાના ગુણધર્મો જોડવામાં આવે છે. બીજા શબ્દોમાં કહીએ તો વિવિધ ગુણધર્મો (એલિમેન્ટ) બાબતે વધુ વિગતો આપે છે. કોઈ પણ એલિમેન્ટના ગુણધર્મો તેના શરૂઆતના (ઓપનિંગ) ટૅગની સાથે દર્શાવાય છે. ગુણધર્મ બે ભાગના બનેલા હોય છે, જેનો પહેલો ભાગ ગુણધર્મનું નામ હોય છે અને બીજો ભાગ તેના માટે જરૂરી કિંમતનો હોય છે.

ગુણધર્મનું નામ ક્યો ગુણધર્મ ગોઠવવો તે દર્શાવે છે. કોડ-લિસ્ટિંગ 1.2માં દર્શાવેલ <a> ટૅગમાં href એ ગુણધર્મનું નામ છે. કિંમત એ ગુણધર્મ માટે ગોઠવવાની કિંમત છે. hrefના કિસ્સામાં કિંમત p4.html છે (જે જોડાણનું સરનામું છે). આ કિંમત બે અવતરણ ચિહ્નમાં આવરીને દર્શાવવી પડે. નામ અને કિંમતની વચ્ચે બરાબર (=) નું ચિહ્ન મૂકવું પડે છે. ગુણધર્મ સાથેની ટૅગ કેવી રીતે દર્શાવવી તેનાં ઉદાહરણ હવે પછીના વિભાગમાં જોઈએ.

Align ગુણધર્મ

align ગુણધર્મ એવું દર્શાવે છે કે, મથાળું પાનાની ડાબી બાજુ, જમણી બાજુ કે મધ્યમાં કયા સ્થાને દેખાવું જોઈએ. સામાન્ય રીતે લખાણ પાનાની ડાબી બાજુ જ ગોઠવાતું હોય છે. Align ગુણધર્મ સાથે નીચેની ત્રણ પૈકીની કોઈ એક કિંમત ગોઠવી શકાય.

Left : લખાણ પાનાની ડાબી બાજુ ગોઠવાય છે.

Right : લખાણ પાનાની જમણી બાજુ ગોઠવાય છે.

Center : લખાણ પાનાની મધ્યમાં ગોઠવાય છે.

align ગુણધર્મના ઉપયોગો દર્શાવતાં કેટલાંક ઉદાહરણો નીચે મુજબ છે :

```
<p align="right"> This content will be displayed in right aligned form </p>
```

```
<p align="center"> This content will be displayed in center position of the page </p>
```

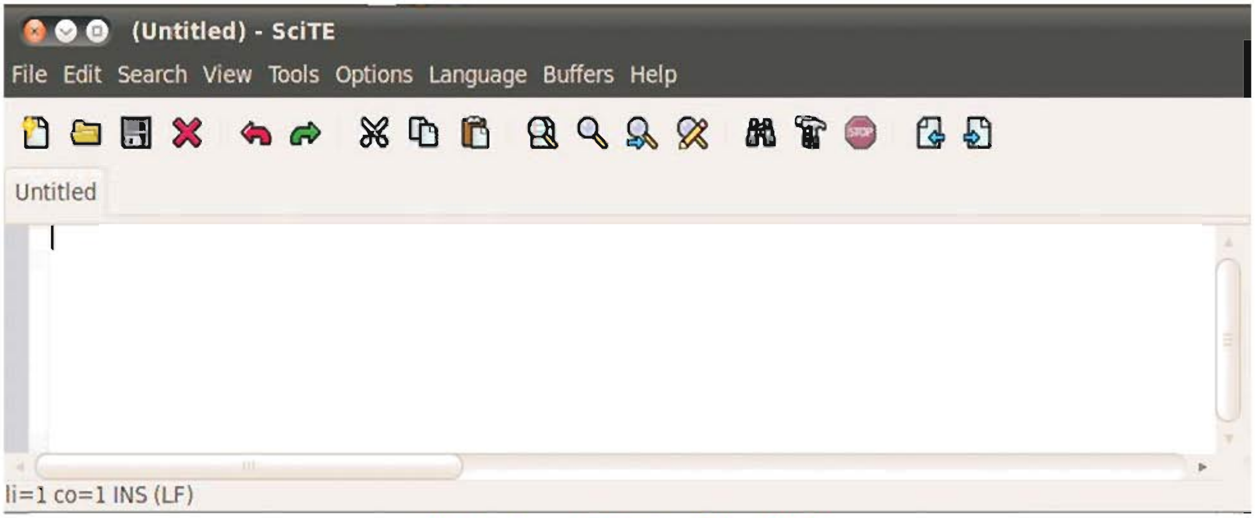
એવા પણ કેટલાક ગુણધર્મો છે કે જે લગભગ દરેક ટૅગ સાથે આવી શકે છે. આવા ગુણધર્મને સાર્વત્રિક ગુણધર્મ કહેવાય. align એ આવો જ એક ગુણધર્મ છે. સાર્વત્રિક ગુણધર્મ હોવાને લીધે align ગુણધર્મને નીચે દર્શાવ્યા મુજબ heading સાથે પણ વાપરી શકાય.

```
<h1 align="center"> Centered Heading </h1>
```

જ્યારે ફકરાના લખાણને સરખું ગોઠવવામાં (align) આવે છે, ત્યારે સરખી ગોઠવણ કરવા માટે લખાણમાં શબ્દોની વચ્ચે ક્યાંક ક્યાંક વધારાની ખાલી જગ્યા ઉમેરવામાં આવે છે. આ રીતે આપમેળે ઉમેરાતી ખાલી જગ્યાને "સૉફ્ટ સ્પેસ" (soft space) કહે છે. જો ઉપયોગકર્તા પોતે ખાલી જગ્યા ઉમેરે તો તેને "હાર્ડ સ્પેસ" (hard space) કહેવાય. જો HTML કોડમાં <pre> અને </pre> ટૅગની વચ્ચે લખાયેલ લખાણ સિવાયના લખાણમાં આવી હાર્ડ સ્પેસ હશે તો તે આપમેળે કાઢી નાંખવામાં આવશે.

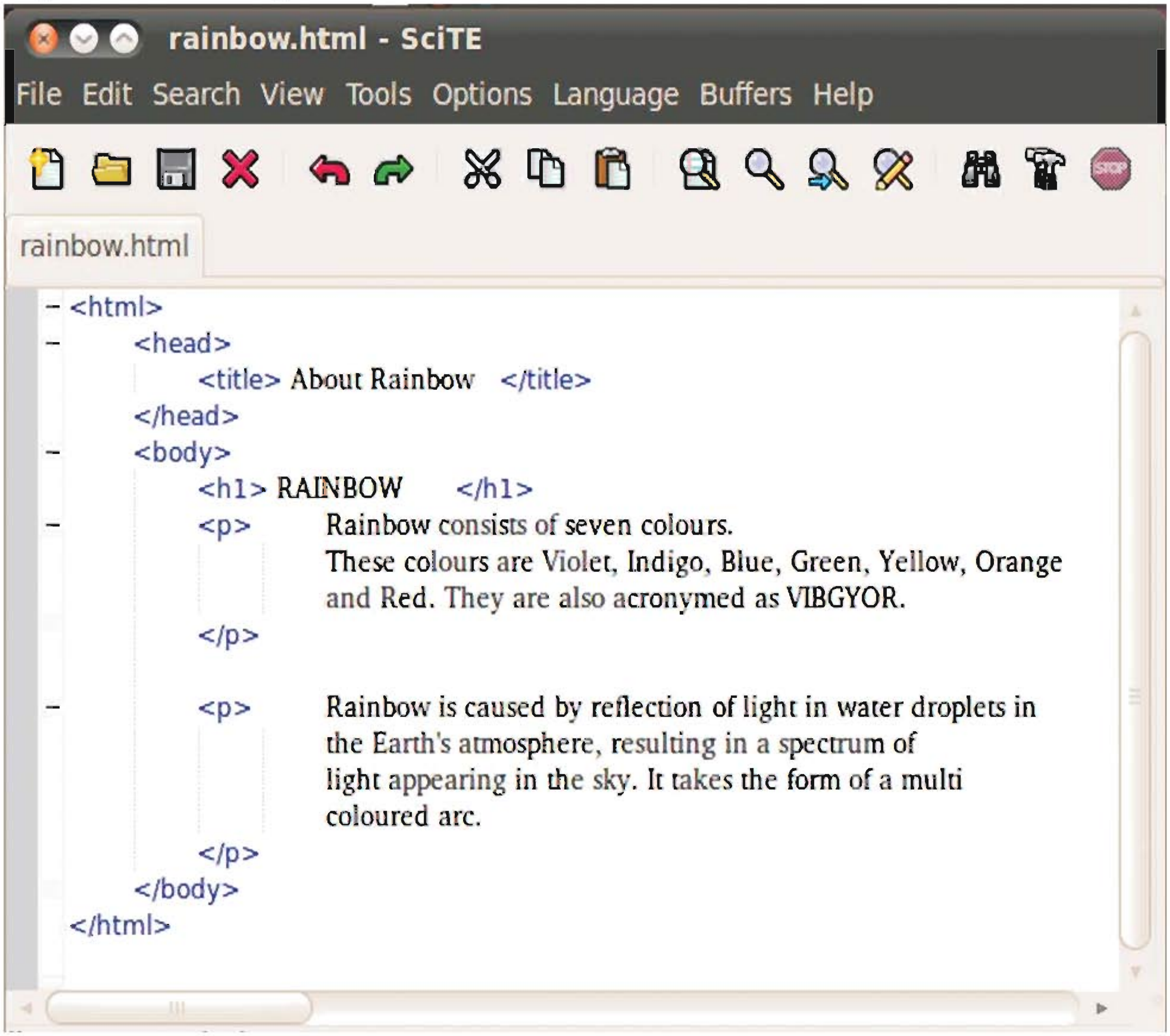
અન્ય એડિટર SciTEનો ઉપયોગ

SciTE એ લખાણને સુધારવા માટેનું એક ટેક્સ્ટ એડિટર (Text Editor) છે, જે Scintilla [http://www.scintilla.org] નામના "ફ્રી સોર્સ કોડ એડિટિંગ કમ્પોનન્ટ" (Free Source Code Editing Component) પર આધારિત છે. તે પૂરેપૂરા સોર્સ કોડ (source code) અને પરવાના સાથે આવે છે, જે તેનો કોઈ પણ નિઃશુલ્ક કે ધંધાકીય સૉફ્ટવેર ઉત્પાદન માટે ઉપયોગ કરવાની અનુમતિ આપે છે. SciTE નો દેખાવ આકૃતિ 1.10માં દર્શાવ્યા મુજબનો દેખાય છે.



આકૃતિ 1.10 : SciTE એડિટરનો દેખાવ

અગાઉ મેઘધનુષ્ય (rainbow) વિશેની માહિતી દર્શાવવા માટે લખાયેલ HTML કોડ(જુઓ આકૃતિ 1.3.)ને આકૃતિ 1.11માં દર્શાવ્યા મુજબ SciTE એડિટરમાં પણ લખી શકાય.



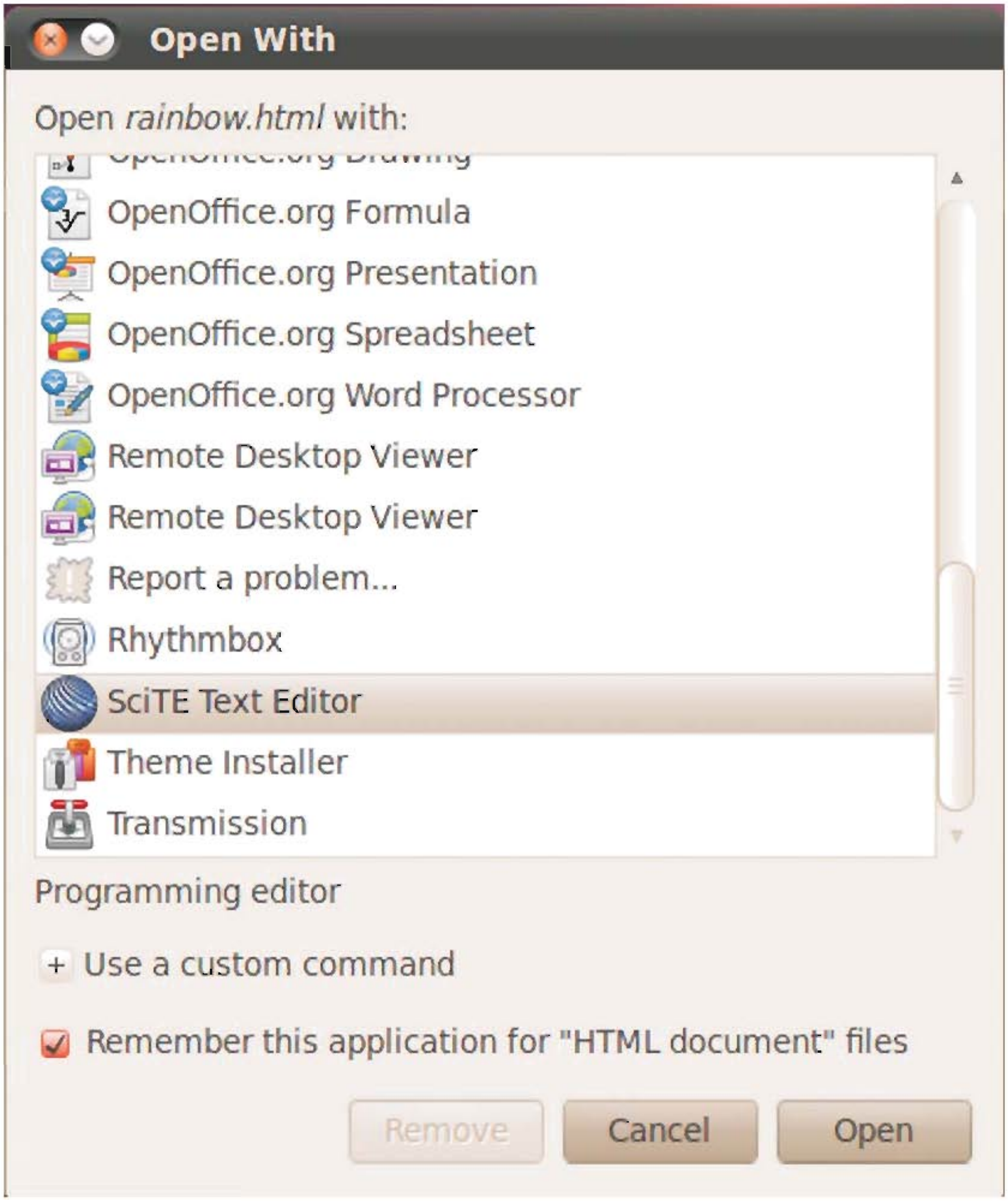
આકૃતિ 1.11 : SciTE એડિટરમાં HTML કોડ

આકૃતિ 1.11માં સ્ક્રીનના ઉપરના ભાગે દર્શાવાયેલ મેનુબાર વિકલ્પો ઉપર ધ્યાન આપો. મેનુ વિકલ્પો ફાઈલ પરની ક્રિયાઓ, લખાણને સુધારવાની સવલતો, માહિતી શોધવા, નિહાળવા તેમજ અન્ય મૂળભૂત સવલતો ધરાવે છે. મેનુ વિકલ્પોની તરત નીચેની લીટીમાં કેટલીક મૂળભૂત ક્રિયાઓ જેવી કે, new, open, save, edit, search વગેરે માટેના આઇકોન રજૂ કરે છે. SciTE એડિટરમાં હયાત કોડને તમે નીચે જણાવેલ પગલાં અનુસરીને ખોલી શકશો.

પગલું 1 : જે ફાઈલને ખોલવી હોય તેને મેળવો.

પગલું 2 : તેના પર જમણું બટન ક્લિક કરો અને Open With પસંદ કરો. આકૃતિ 1.12માં દર્શાવ્યા મુજબનો સ્ક્રીન દેખાશે.

પગલું 3 : SciTE ટેક્સ્ટ એડિટર પસંદ કરો. તે ફાઈલને SciTE એડિટરમાં ખોલશે. SciTE દ્વારા દર્શાવાતા ઇન્ડેન્ટ (indentation) માટેના નિર્દેશો અને ટેગના રંગોની નોંધ લો. આ રીતે ઇન્ડેન્ટ અને ટેગ વિવિધ રંગોમાં હોવાને લીધે લખાણ અને ટેગ જુદા તરી આવે છે અને કોડને વાંચવામાં સરળતા બક્ષે છે.



આકૃતિ 1.12 : Open With ડાયલોગ બૉક્સ

પગલું 4 : બ્રાઉઝરમાં html કોડનું પરિણામ જોવા માટે Tools → Go પસંદ કરો અથવા F5 કી દબાવો.

HTMLની વર્તમાન આવૃત્તિ

HTMLના વિકાસકાળમાં તેનું ધ્યાન તેના ઉદ્દેશ્ય તરફ કેન્દ્રિત કર્યું, જેથી કરીને લખાણને ટાઈપ કરીને તૈયાર કરવાનું અને પ્રકાશિત કરવાનું અસરકારક રહે. સમયાંતરે HTMLમાં વધુ ને વધુ સવલતો ઉમેરાતી ગઈ જેથી કરીને તે નવા બ્રાઉઝર્સ, નવી ટેક્નોલૉજી અને સર્જકોની સતત વધતી જતી માંગને અનુરૂપ બની શકે અને ઉપયોગી થઈ શકે. HTMLની અદ્યતન આવૃત્તિને HTML 5.0 તરીકે ઓળખવામાં આવે છે અને તે અત્યારે હાલમાં લોકપ્રિય થઈ રહી છે.

આ પ્રકરણમાં આવરી લીધેલ ટૅગ

આ પ્રકરણમાં આપણે કોષ્ટક 1.3માં દર્શાવેલ ટૅગની ચર્ચા કરી છે.

ટૅગ (Tag)	વર્ણન (Description)
<a href> ... 	મૂળ ફાઈલને લિન્ક દ્વારા જોડવામાં આવેલી (referred) ફાઈલ સાથે સાંકળે છે.
 ..	લખાણને ઘાટા અક્ષરોમાં દર્શાવે છે.
<body>...</body>	HTML દસ્તાવેજનો મુખ્ય ભાગ (Body) નક્કી કરે છે. <html> ટૅગની જોડીની વચ્ચે દર્શાવાય છે.
  or 	લીટીનો અંત (Line break) દર્શાવે છે. તે અયુગ્મ ટૅગ છે.
...	પ્રથમ સ્તરનું મથાળું દર્શાવે છે.
...	દ્વિતીય સ્તરનું મથાળું દર્શાવે છે.
...	તૃતીય સ્તરનું મથાળું દર્શાવે છે.
...	ચોથા સ્તરનું મથાળું દર્શાવે છે.
...	પાંચમા સ્તરનું મથાળું દર્શાવે છે.
...	છઠ્ઠા સ્તરનું મથાળું દર્શાવે છે.
<head>...</head>	HTML દસ્તાવેજનો મુખ્ય વિભાગ દર્શાવે છે. <html> ટૅગની જોડીની વચ્ચે આવે છે.
<html>...</html>	આખા HTML દસ્તાવેજને આવરી લે છે.
<i> ..</i>	લખાણને ત્રાંસા (italics) સ્વરૂપે છાપે છે.
...	ફકરો દર્શાવે છે.
<pre>...</pre>	પૂર્વરૂપાંકિત (preformated) લખાણ દર્શાવે છે.
<s> ..</s>	છેકી નાંખેલ સ્વરૂપે લખાણ દર્શાવે છે.
<title>...</title>	દસ્તાવેજનું શીર્ષક દર્શાવે છે. <head> નામની ટૅગની જોડીની વચ્ચે આવે છે.
<tt> ..</tt>	લખાણને ટાઈપરાઈટર જેવા અક્ષરોમાં દર્શાવે છે.
<u> ..</u>	લખાણને નીચે લીટી દોરેલ હોય તેવા સ્વરૂપે દર્શાવે છે.

કોષ્ટક 1.3 : પ્રકરણ 1માં આવરી લેવાયેલ HTML ટૅગ

સારાંશ

આ પ્રકરણમાં આપણે HTMLનો ઉપયોગ કરીને વેબપેઇજ કેવી રીતે તૈયાર કરવાં તે શીખ્યા. HTML એ એવી મૂળભૂત સુવિધા છે, જે વેબની માહિતી કેવી રીતે તૈયાર થાય, પ્રકાશિત થાય અને વેબ બ્રાઉઝરનો ઉપયોગ કરી કેવી રીતે દર્શાવવી તે વર્ણવે છે. આ પ્રકરણમાં, HTML દસ્તાવેજના માળખાના બે મહત્ત્વના ભાગ જેવા કે, HEAD અને BODY સહિત કેટલાક મૂળ ટૅગ વર્ણવવામાં આવ્યા. માહિતીનો ઉપયોગ કરીને તમે સાદાં વેબપેઇજ તૈયાર કરી શકો. તૈયાર કરેલાં અનેક વેબપેઇજને તમે આ પ્રકરણમાં ચર્ચા કરેલ ઍન્કર ટૅગ વડે જોડી શકો. HTMLના મૂળભૂત ખ્યાલો અને HTMLનો ઇતિહાસ પૂરો પાડવા ઉપરાંત આ પ્રકરણમાં HTML દસ્તાવેજ તૈયાર કરવા gedit અને SciTE જેવા એડિટર વિશે પણ માહિતી આપવામાં આવી છે.

સ્વાધ્યાય

1. HTMLના ઇતિહાસ અને તેના વિકાસ વિશે ટૂંક નોંધ લખો.
2. HTML દસ્તાવેજનું સરળ ઉદાહરણ આપી HTML દસ્તાવેજના માળખાનું વર્ણન કરો.
3. HTML એલિમેન્ટ એટલે શું ? HTML એલિમેન્ટનું માળખું સમજાવો.
4. HTML દસ્તાવેજને તમારા ડેસ્કટોપ પર કેવી રીતે નિહાળી શકાય ?
5. કોઈ પણ ત્રણ બ્રાઉઝરની યાદી બનાવો.
6. HTML મથાળાની વિવિધ ઢબ ઉદાહરણ સાથે ચર્ચો.
7. HTMLના વિવિધ ફોર્મેટિંગ ટેગ ઉદાહરણ સાથે ચર્ચો.
8. HTMLની પૂર્વરૂપાંકિત (preformatted) ટેગની ઉદાહરણ સાથે ચર્ચા કરો.
9. HTMLના ઍન્કર ટેગ વિશે ટૂંક નોંધ લખો.
10. HTMLમાં પૂર્ણ અને સંબંધિત સરનામાની વ્યાખ્યા આપી ચર્ચા કરો.
11. **નીચેના પ્રશ્નો માટે આપેલા વિકલ્પોમાંથી યોગ્ય વિકલ્પ પસંદ કરો :**

(1) વેબની માહિતી દર્શાવવા માટે કઈ માર્ક-અપ ભાષાની જરૂર પડે છે ?

(a) CML (b) HTML (c) NML (d) WML

(2) નીચેનામાંથી વેબપેઇજ વર્ણવવા માટેની ભાષા કઈ ગણાય છે ?

(a) HTML (b) WML (c) NML (d) CML

(3) નીચેનામાંથી HTMLનું પૂરું નામ કયું છે ?

(a) Hot Text Manipulation Language (b) Hyper Text Manipulation Law

(c) Hyper Text Markup Language (d) Hidden Text Markup Language

(4) નીચેનામાંથી SGMLનું પૂરું નામ કયું છે ?

(a) Standardized General Markup Language

(b) System General Manipulation Law

(c) Standardized Genome Markup Law

(d) Standardized Gigabyte Markup Language

(5) નીચેનામાંથી HTML એલિમેન્ટને કોણ સંબોધે છે ?

(a) શરૂઆતની ટેગ, લખાણ અને અંતની ટેગ (b) ખૂણિયા કૌંસ

(c) લખાણ (d) આપેલામાંથી કોઈ પણ

(6) HTML એલિમેન્ટની સાથે વધારાની ગોઠવણ કરવા નીચેનામાંથી શું ઉપયોગમાં લેવાય છે ?

(a) સંખ્યા (નંબર) (b) ગુણધર્મો (એટ્રીબ્યુટ)

(c) ટિપ્પણી (કોમેન્ટ) (d) લખાણ (કન્ટેન્ટ)

(7) નીચેનામાંથી કયો વિકલ્પ ખાલી ટેગ રજૂ કરે છે, જેમાં કોઈ લખાણની જરૂર પડતી નથી ?

(a) Compete (b) Empty (c) Null (d) Void

(8) નીચેનામાંથી કયા ગુણધર્મનો પ્રકાર કોઈ પણ ટેગ સાથે આવી શકે ?

(a) અજોડ (Unique) (b) સાર્વત્રિક (Universal)

(c) બિન અગત્યનો (Trivial) (d) પ્રાથમિક (Preliminary)

(9) HTML દસ્તાવેજમાં કયા પ્રકારની માહિતી સમાવી શકાય ?

(a) મલ્ટીમીડિયા માહિતી (b) શાબ્દિક માહિતી

(c) ફાઈલનામનું પથનામ અને ફાઈલનું નામ (d) આપેલ તમામ

(10) HTML દસ્તાવેજને સુધારવા માટે નીચેના પૈકી કયું એડિટર છે ?

(a) SciTE (b) BriTE (c) LigHT (d) SpriTE

પ્રાયોગિક સ્વાધ્યાય

1. તમારી શાળા વિશેની પ્રારંભિક માહિતી આપતું વેબપેઇજ બનાવો. આ પાનાનું મથાળું "My School" આપો. જરૂરી ફૉર્મેટિંગ અને રજૂઆતકર્તા ટૅગનો ઉપયોગ કરો.

2. એક વેબપેઇજ બનાવો, જે તમારા ક્લાસનો પરિચય કરાવે. તેમાં તમારા વર્ગશિક્ષક, અન્ય વિષયના શિક્ષક અને જે વિષયો તમે ભણો છો તેની માહિતીનો સમાવેશ કરો. આ પાનાનું મથાળું "My School" આપો. જરૂરી ફૉર્મેટિંગ અને રજૂઆતકર્તા ટૅગનો ઉપયોગ કરો.

3. આ સ્વાધ્યાયના પ્રકરણમાં તમે તૈયાર કરેલ વેબપેઇજને બોલાવવા માટે આ સ્વાધ્યાયના પ્રશ્ન 1 માં તૈયાર કરેલ વેબપેઇજને સુધારો. ''હૉટ ટેક્સ્ટ''ને એ રીતે ગોઠવો કે, જેથી જ્યારે 'My School' પાના પરથી ક્લિક કરવામાં આવે ત્યારે તે 'My Class' પાના પર પહોંચી જાય.

HTMLમાં Head અને Body વિભાગ

Head વિભાગ

કોઈ પણ HTML દસ્તાવેજને Head અને Bodyના નામે ઓળખાતા બે ભાગમાં વહેંચવામાં આવે છે. HTML દસ્તાવેજનો head વિભાગ દસ્તાવેજ વિશેની જરૂરી માહિતી પ્રદાન કરે છે. Head વિભાગના લખાણને <head> અને </head> નામના બે ટૅગની વચ્ચે લખવામાં આવે છે. આ બન્ને ટૅગ અને તેની વચ્ચેના લખાણને head એલિમેન્ટ તરીકે ઓળખવામાં આવે છે.

Head વિભાગમાં સમાવી શકાતી પ્રથમ જરૂરી બાબત છે, શીર્ષક (title). અહીં એ ખાસ નોંધવું જોઈએ કે આ શીર્ષક વેબપેઇજના લખાણ તરીકે દર્શાવાશે નહીં. તે માત્ર વેબપેઇજને દર્શાવતી બ્રાઉઝર વિન્ડોના શીર્ષક તરીકે દર્શાવાય છે.

અહીં એ પણ ખાસ નોંધવું જોઈએ કે, જો head વિભાગમાં શીર્ષક લખવામાં ન આવ્યું હોય તો, મોટાભાગના બ્રાઉઝર દ્વારા બ્રાઉઝર વિન્ડોના શીર્ષક તરીકે URL પથ અથવા ફાઈલનું નામ દર્શાવાય છે. વધુમાં, title ટૅગને અંતે પૂરો થતો ટૅગ લગાડવાનું ન ભૂલવું જોઈએ. જો અંત ટૅગ લગાડવાનું ભૂલી જવાયું હશે તો આખા લખાણને શીર્ષક માની લેવામાં આવશે અને કદાચ એવું પણ બને કે, HTML દસ્તાવેજને બ્રાઉઝર વિન્ડોમાં લાવવામાં ન આવે.

Head વિભાગમાં HTML દસ્તાવેજ અને લખાણ બાબતે કેટલીક વધારાની માહિતી પણ આવે છે. વધારાની માહિતી આપતા ટૅગને મેટા-ટૅગ (meta-tag) તરીકે ઓળખવામાં આવે છે. મેટા-ટૅગનો ઉપયોગ મોટે ભાગે બ્રાઉઝર અને સર્ચ એન્જિન(Search Engine)ને સંબંધિત માહિતી સંગ્રહવા માટે થાય છે. યોગ્ય અને બંધબેસતાં મેટા-ટૅગનો ઉમેરો વેબપેઇજની પ્રકૃતિ વર્ણવે છે અને સર્ચ એન્જિન દ્વારા વેબપેઇજને અસરકારક રીતે શોધવાનું સરળ બનાવે છે.

વર્ણન

મોટાભાગના સર્ચ એન્જિન ઉપયોગકર્તાને જ્યારે શોધનું પરિણામ રજૂ કરશે ત્યારે તે વર્ણન દર્શાવશે. આવા વર્ણનની અનુપસ્થિતિના કિસ્સામાં, સર્ચ એન્જિન માત્ર શરૂઆતના થોડાક શબ્દો દર્શાવશે. મેટા-ટૅગના description ગુણધર્મનું ઉદાહરણ નીચે મુજબ છે:

<meta name="DESCRIPTION" content="About rainbow and its colours">

ચાવીરૂપ શબ્દો

આ ટૅગમાં આપવામાં આવેલ ચાવીરૂપ શબ્દો (Keywords) સર્ચ એન્જિન દ્વારા ઉપયોગમાં લેવાશે. અગત્યની લાક્ષણિકતાઓનાં નામ, વેબપેઇજનો ઉદ્દેશ અને અગત્યના મુદ્દાને કી-વર્ડ તરીકે રજૂ કરી શકાય. Keywords વર્ણવતા મેટા-ટૅગ ગુણધર્મ નીચે મુજબ છે:

<meta name="KEYWORDS" content="Rainbow, VIBGYOR">

લેખક

નીચેનો મેટા-ટૅગ ગુણધર્મ, વેબપેઇજના સર્જક કે લેખક (Author) વિશેની માહિતી પ્રદાન કરે છે.

<meta name="AUTHOR" content="M K Gandhi">

ટિપ્પણી (Comments)

ટિપ્પણી (કોમેન્ટસ્) HTML કોડમાં વધારાની માહિતી આપવાની તક આપે છે. કોમેન્ટસ્ પરિણામ સ્વરૂપે દર્શાવવા માટે નથી હોતી. એટલે કે, HTMLના સોર્સ કોડ સિવાય કોઈ પણ જગ્યાએ ટિપ્પણી જોઈ શકાશે નહીં. ટિપ્પણી મૂકવા માટે આપણે '<!--' અને '-->' ટૅગનો ઉપયોગ કરવો પડે. '<!--' ટૅગ ટિપ્પણીની શરૂઆત દર્શાવે છે જ્યારે '-->' ટૅગ દ્વારા ટિપ્પણીનો અંત દર્શાવાય છે. ટિપ્પણીને "આમુખ" (prologue) તરીકે પણ ઓળખવામાં આવે છે. <!-- This is a comment --> એ ટિપ્પણીનું એક ઉદાહરણ છે.

અન્ય મેટા-ટૅગ

એવા પણ કેટલાક મેટા-ટૅગ છે, જે વેબપેઇજને અમુક સેકન્ડ પછી કોઈ ચોક્કસ URLને લઈ આવવાનું કહે અથવા તેને કહી દે કે પાનાં ન આવવાં જોઈએ. નીચેનું ઉદાહરણ દર પાંચ સેકન્ડ પછી વેબપેઇજને પુનઃ લાવી (reload કરી) વેબપેઇજને રીફ્રેશ (refresh) કરે છે.

```
<meta http-equiv="REFRESH" content="5">
```

નીચેનું ઉદાહરણ http://test.com/ નામના URL ના લખાણને દર પાંચ સેકન્ડે પુનઃ તાજું કરશે.

```
<meta http=equiv="refresh" content="5" URL='http://test.com/'">
```

HTML base

<base> ટૅગ વેબપેઇજની તમામ સંબંધિત URL માટે આધારરૂપ URL/ટાર્ગેટ નિશ્ચિત કરે છે.

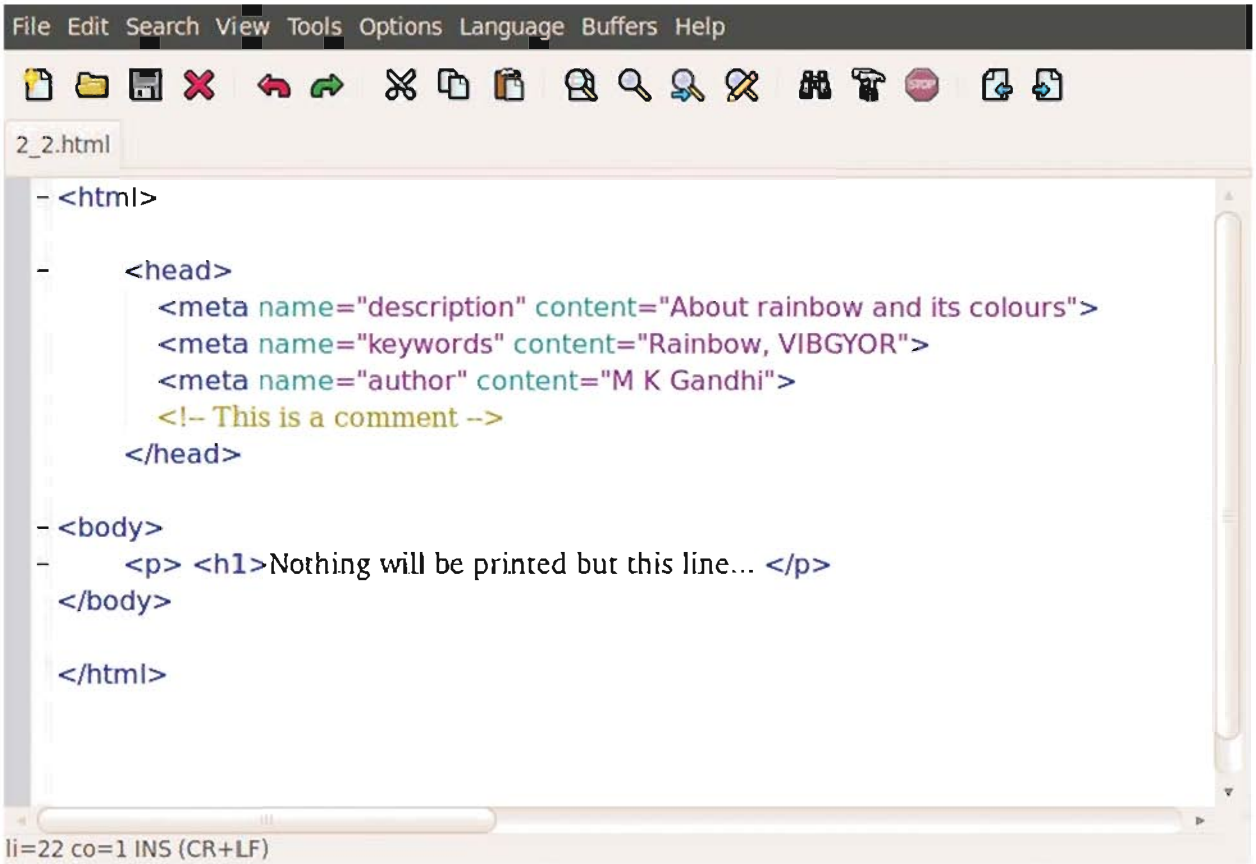
```
<head>
<base href="http://test.com/ " >
</head>
```

ટેબલ 2.1માં સારાંશરૂપે કેટલાક અગત્યના HTML head એલિમેન્ટ રજૂ કરેલ છે.

ટૅગ	વર્ણન
<head>	દસ્તાવેજ વિશેની માહિતી વ્યાખ્યાયિત કરે છે.
<title>	દસ્તાવેજનું શીર્ષક વ્યાખ્યાયિત કરે છે.
<base>	વેબપેઇજની તમામ લિન્ક માટે એક સામાન્ય સરનામું અથવા ગંતવ્ય સ્થાન વ્યાખ્યાયિત કરે છે.
<link>	દસ્તાવેજ અને બાહ્ય સંસાધન વચ્ચે સંબંધ વ્યાખ્યાયિત કરે છે.
<meta>	HTML દસ્તાવેજ અંગેના મેટા-ડેટા વ્યાખ્યાયિત કરે છે.
<script>	ક્લાયન્ટ તરફની સ્ક્રિપ્ટ વ્યાખ્યાયિત કરે છે.
<style>	દસ્તાવેજ માટેની ઢબની માહિતી વ્યાખ્યાયિત કરે છે.

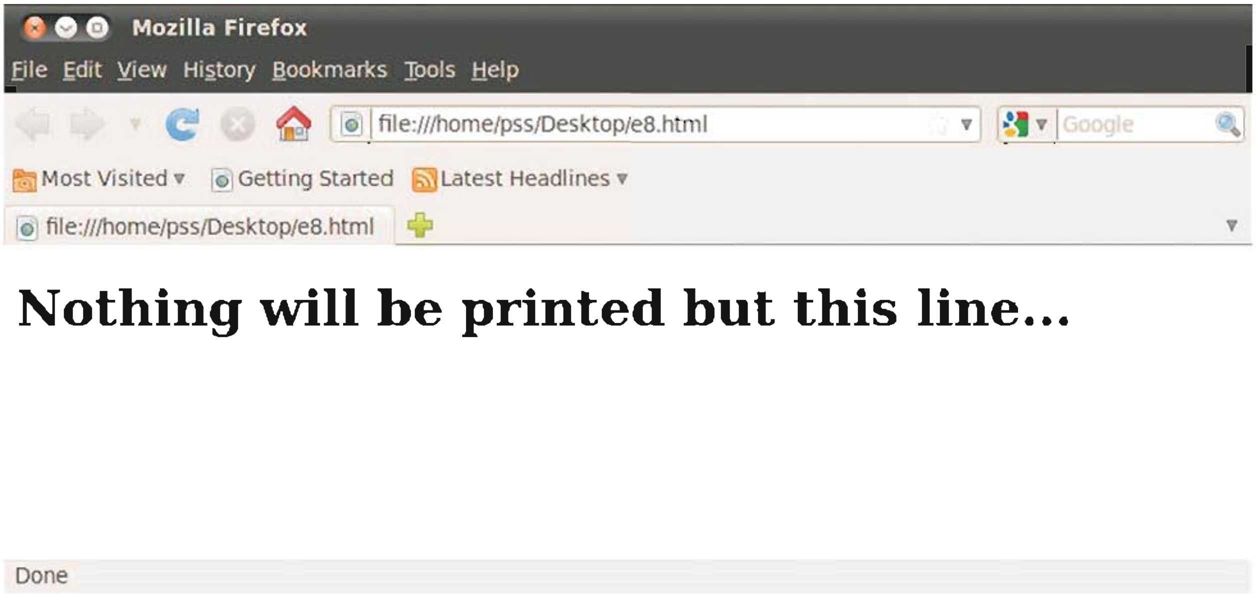
કોષ્ટક 2.1 : HTML દસ્તાવેજનું Head એલિમેન્ટ

head એલિમેન્ટ સાથે અવલોકન માટે આકૃતિ 2.1માં દર્શાવેલ HTML કોડને ધ્યાને લો.



આકૃતિ 2.1 : HTML ના Head એલિમેન્ટનો પ્રયોગ

ઉપરના કોડનો અમલ કરો અને બ્રાઉઝરમાં તેનું પરિણામ જુઓ. તે બિલકુલ આકૃતિ 2.2 જેવું દેખાશે.



આકૃતિ 2.2 : આકૃતિ 2.1માં દર્શાવેલ HTML કોડનું પરિણામ

Body વિભાગ

HTML દસ્તાવેજનો મુખ્ય વિભાગ <body> અને </body> ટૅગની વચ્ચે લખવામાં આવે છે. HTMLનું body એલિમેન્ટ બ્રાઉઝરની અંદર જેને દર્શાવવાની હોય તે બાબતોના સંગ્રાહક તરીકે કાર્ય કરે છે.

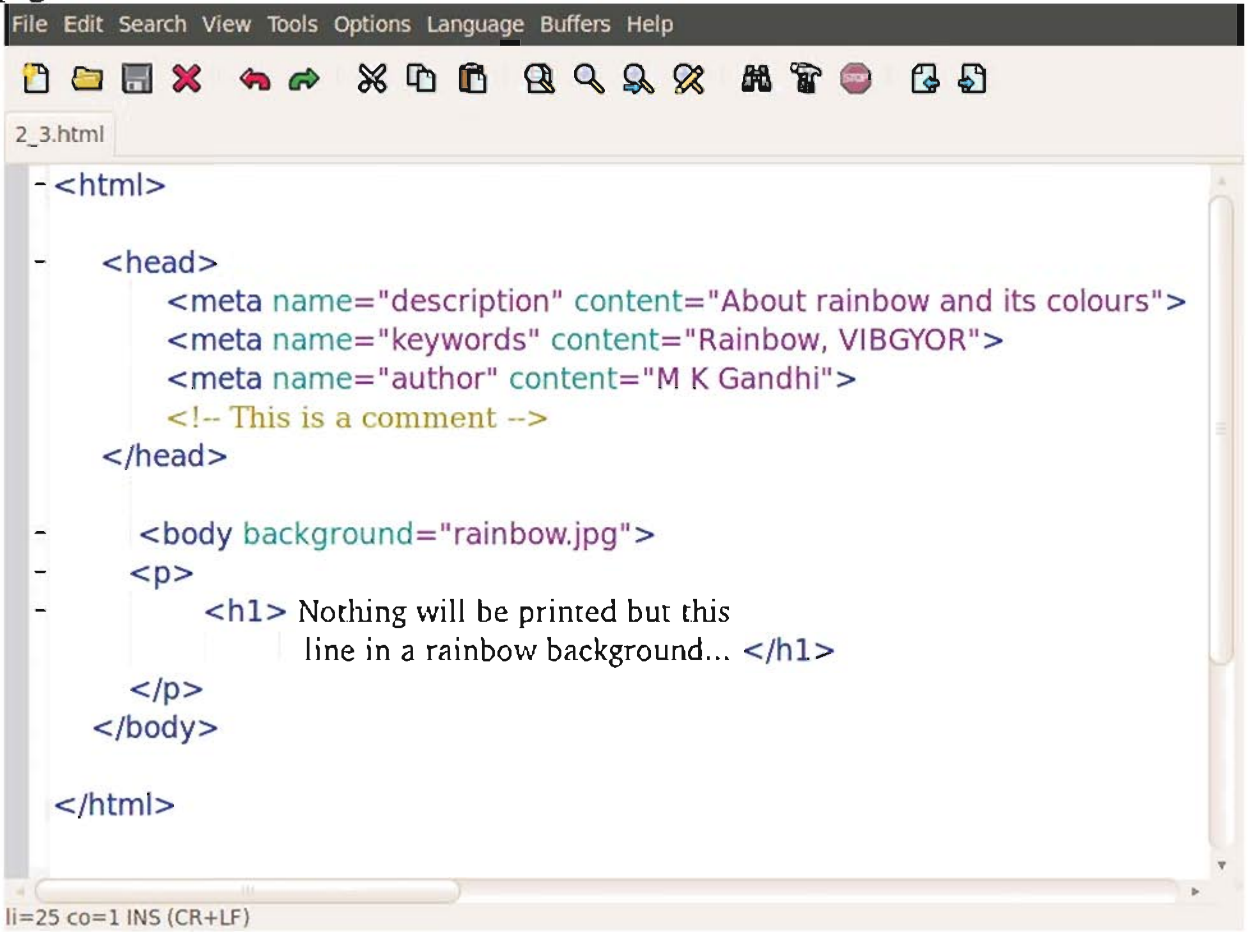
આગળના પ્રકરણમાં આપણે મથાળાં, ફકરા અને લખાણના સ્વરૂપાંકન માટેનાં કેટલાંક ટૅગ વિશે વાત કરી કે જે body વિભાગમાં આવી શકે. સામાન્ય રીતે, આ ટૅગની વચ્ચે કંઈક લખાણ જરૂરી હોય છે. એવા પણ કેટલાક ટૅગ હોય છે જેની વચ્ચે કોઈ લખાણની જરૂર હોતી નથી, આવા ટૅગને ખાલી ટૅગ (Empty Tag) કહે છે. આવા ખાલી ટૅગનું ઉદાહરણ છે
. તો ચાલો, HTML દસ્તાવેજના body વિભાગમાં ઉપયોગમાં લેવાતા કેટલાંક વધુ ટૅગ અને ગુણધર્મો જોઈએ.

બૅકગ્રાઉન્ડ છબી

અત્યાર સુધી, આપણે એવા HTML કોડ જોયા છે, જે સાદા સફેદ બૅકગ્રાઉન્ડ પર લખાણ દર્શાવે. સજાવટસભર અને રંગીન બૅકગ્રાઉન્ડ પર લખાણને રજૂ કરવા Body ટૅગ સાથે આપણે background ગુણધર્મનો પણ ઉપયોગ કરી શકીએ. ટેક્સ્ટનાં બૅકગ્રાઉન્ડ માટે આપણે આપણા કમ્પ્યૂટર પર ઉપલબ્ધ કોઈ પણ છબીનો ઉપયોગ કરી શકીએ. HTML પાના માટેના બૅકગ્રાઉન્ડ ગોઠવવા આપણે નીચે મુજબની ટૅગનો ઉપયોગ કરી શકીએ :

```
<body background ="rainbow.jpg">
```

અહીં background શબ્દ એ એક ગુણધર્મનું નામ છે. ગુણધર્મને એક ચોક્કસ કિંમત આપવી પડે. આ કિંમત તરીકે આપણે એવી ફાઈલનું નામ આપવું જોઈએ (મોટેભાગે છબીની ફાઈલ), કે જેને આપણે બૅકગ્રાઉન્ડ તરીકે દર્શાવવા ઇચ્છીએ છીએ. આપણા કિસ્સામાં તે rainbow.jpg છે. આપણે કોઈ પણ માન્ય છબી સ્વરૂપ જેવાં કે jpg, bmp, png અને tiff આપી શકીએ.



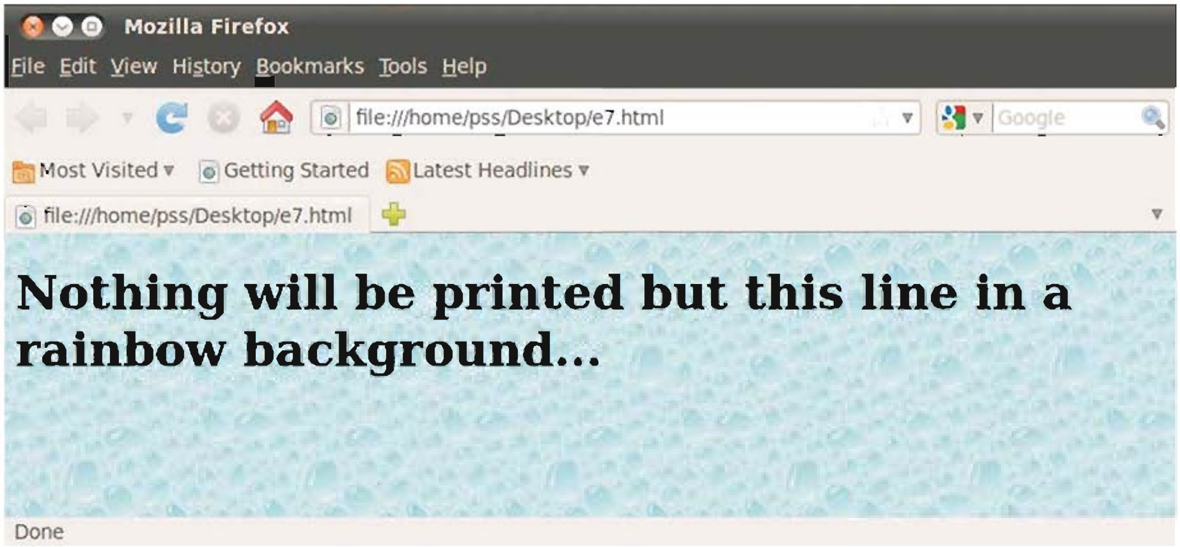
આકૃતિ 2.3 : બૅકગ્રાઉન્ડ છબી મૂકવા માટેનો HTML કોડ

તો ચાલો, આકૃતિ 2.3માં દર્શાવ્યા મુજબના HTML કોડ લખીએ કે જે, rainbow.jpg નામની છબી વડે બૅકગ્રાઉન્ડ રચે. એ નોંધવું જોઈએ કે, અહીં તમે આ rainbow.jpgને બદલે તમારી પસંદગીની કોઈ પણ છબી ફાઈલને પણ ગોઠવી શકો.

જ્યારે આપણે HTML કોડની અંદર છબી ફાઈલનો ઉપયોગ કરીએ છીએ, ત્યારે આપણે એ ખાતરી કરી લેવી

જોઈએ કે તે ફાઈલનું નામ, તેનું અનુલંબન અને તેનો પથ યોગ્ય રીતે લખવામાં આવ્યો છે કે નહિ. જો ફાઈલનું નામ દર્શાવવા બાબતે કોઈ ભૂલ થશે તો પાનાંના બૅકગ્રાઉન્ડ તરીકે કંઈ પણ દેખાશે નહીં.

આકૃતિ 2.3માં દર્શાવ્યા મુજબ કોડ ટાઈપ કરો અને તમારી ઈચ્છાનુસાર ફાઈલનાં નામ સાથે સાચવી દો. (અહીં તે e7.html તરીકે સચવાઈ છે.) બ્રાઉઝરમાં કોડને નિહાળો. બૅકગ્રાઉન્ડ છબી ઉમેરવાનું પરિણામ આકૃતિ 2.4 દર્શાવે છે.



આકૃતિ 2.4 : બૅકગ્રાઉન્ડ તરીકે છબી ઉમેરવી

ઘણી વખત, બૅકગ્રાઉન્ડની છબી તેના લખાણ કરતાં વધુ આકર્ષક હોય છે. આપણે એ બાબતની પણ ખાસ કાળજી લેવી જોઈએ કે વાચકનું ધ્યાન લખાણ પરથી હટીને બૅકગ્રાઉન્ડ પર કેન્દ્રિત ન થઈ જાય. બૅકગ્રાઉન્ડ તરીકે રજૂ કરાનાર છબી સુરમ્ય હોવી જોઈએ અને લખાણને વાંચવા પ્રેરે તેવી હોવી જોઈએ નહીં કે, લખાણ ઢંકાઈ જાય તેવી. વળી, જો બૅકગ્રાઉન્ડ છબી વધુ પડતી મોટી હશે તો તેને બ્રાઉઝરમાં આવવામાં ઘણો સમય લાગશે. અહીં એ પણ નોંધવું જોઈએ કે, જો બૅકગ્રાઉન્ડ માટે ઉપયોગમાં લેવાનારી છબીનું કદ સ્ક્રીનના કદ કરતાં નાનું હશે તો જ્યાં સુધી આખો સ્ક્રીન ભરાઈ નહીં જાય ત્યાં સુધી કદાચ તે છબીની એક પછી એક પ્રતિકૃતિઓ બનતી રહેશે. સામાન્ય રીતે જ્યારે તમે વેબપેઈજને સ્ક્રોલ કરશો ત્યારે બૅકગ્રાઉન્ડ છબી સ્ક્રોલ થશે. સિવાય કે, તમે તેને નીચે દર્શાવ્યા મુજબ સ્થિર કરી દો.

```
<body background="rainbow.gif" bgproperties="fixed">
```

બૅકગ્રાઉન્ડ રંગ

બૅકગ્રાઉન્ડમાં છબી દર્શાવવાને બદલે બૅકગ્રાઉન્ડ રંગનો ઉપયોગ કરીને પણ આપણે વેબપેઈજને આકર્ષક બનાવી શકીએ. ધારો કે, આપણે આપણા વેબપેઈજનું બૅકગ્રાઉન્ડ પીળા રંગનું કરવું હોય તો નીચે મુજબની ટૅગથી કામ થઈ જશે.

```
<body bgcolor="#FFFF00">
```

અહીં, bgcolor એ બૅકગ્રાઉન્ડ રંગ નિર્ધારિત કરતો ગુણધર્મ છે. અહીં એ ખાસ નોંધ લો કે, આપણે નીચે દર્શાવ્યા પ્રમાણે બૅકગ્રાઉન્ડ છબી અને રંગ એકસાથે ગોઠવી શકીએ.

```
<body background="rainbow.jpg" bgcolor="#FFFF00">
```

આવા કિસ્સામાં, જ્યાં સુધી બ્રાઉઝરમાં સંપૂર્ણપણે છબી આવી ન જાય ત્યાં સુધી બૅકગ્રાઉન્ડ રંગ દર્શાવાશે. આવી અસર સામાન્ય રીતે ધીમા કમ્પ્યૂટરમાં દેખાશે, ઝડપી કમ્પ્યૂટર પર તો કદાચ આપણે કોઈ અસર જોઈ ન શકીએ તેવું બને.

ટેલિવિઝન અને કમ્પ્યૂટરના વી.ડી.યુ.ના સ્ક્રીન જેવા ઈલેક્ટ્રોનિક માધ્યમમાં કોઈ પણ રંગ ત્રણ મૂળભૂત રંગ, લાલ (Red), લીલો (Green) અને વાદળી (Blue) નામના ત્રણ રંગોના સંયોજનરૂપે જ ગણાય છે. તેને ટૂંકમાં RGB તરીકે પણ ઓળખવામાં આવે છે.

યાદ કરો, બાળપણમાં તમે બે કે વધુ રંગોનું મિશ્રણ કરી નવો રંગ બનાવતા હતા ! શક્ય છે કે કદાચ જાંબલી રંગ બનાવવા તમે વાદળી અને લાલ રંગનું મિશ્રણ કર્યું હોય !

કમ્પ્યૂટરમાં રંગોને સોળઅંકી પદ્ધતિ મુજબ 00 ડિગ્રીથી FF ડિગ્રી સુધીના સંકેત આપવામાં આવે છે. આમ કોઈ રંગ રજૂ કરવા કેટલોક લાલ, કેટલોક લીલો અને કેટલોક વાદળી (RGB) રંગને રજૂ કરતો છ આંકડાનો કોડ બનાવવો પડે. એટલે કે, દરેક રંગ માટેના બે આંકડા, લાલ રંગ માટેનો કોડ FF0000 થશે. અહીં લાલ રંગને FF કિંમત (દશાંશ સંખ્યા 255ની સમાન સોળઅંકી સંખ્યા), લીલા રંગને 00 અને વાદળી માટેની કિંમત પણ 00 છે. આ ત્રણેય અંકોની મર્યાદા 00થી FF સુધીની છે. (એટલે કે, 0થી 255 સુધી, કુલ 256 કિંમતો) લાલ, લીલા અને વાદળીની 0થી 255 સુધીની કિંમતોનું સંયોજન (256 × 256 × 256) 16 કરોડ વિવિધ રંગો આપે છે. સોળઅંકી સ્વરૂપે રંગોની શબ્દમાળા દર્શાવવાથી આપણે વિવિધ રંગ દર્શાવી શકીએ. આકૃતિ 2.5માં રંગોના કેટલાંક ઉદાહરણો આપ્યાં છે.

FFFFFF→	White colour	→
FF0000→	Red colour	→
FFFF00→	Yellow colour	→
000000→	Black colour	→
FF00FF→	Pink/Magenta colour	→
0000FF→	Blue colour	→
00FF00→	Green colour	→

આકૃતિ 2.5 : રંગોનાં કેટલાંક ઉદાહરણો

આના બદલે તમે નીચે દર્શાવ્યા પ્રમાણે રંગોના નામોનો ઉલ્લેખ પણ કરી શકો:

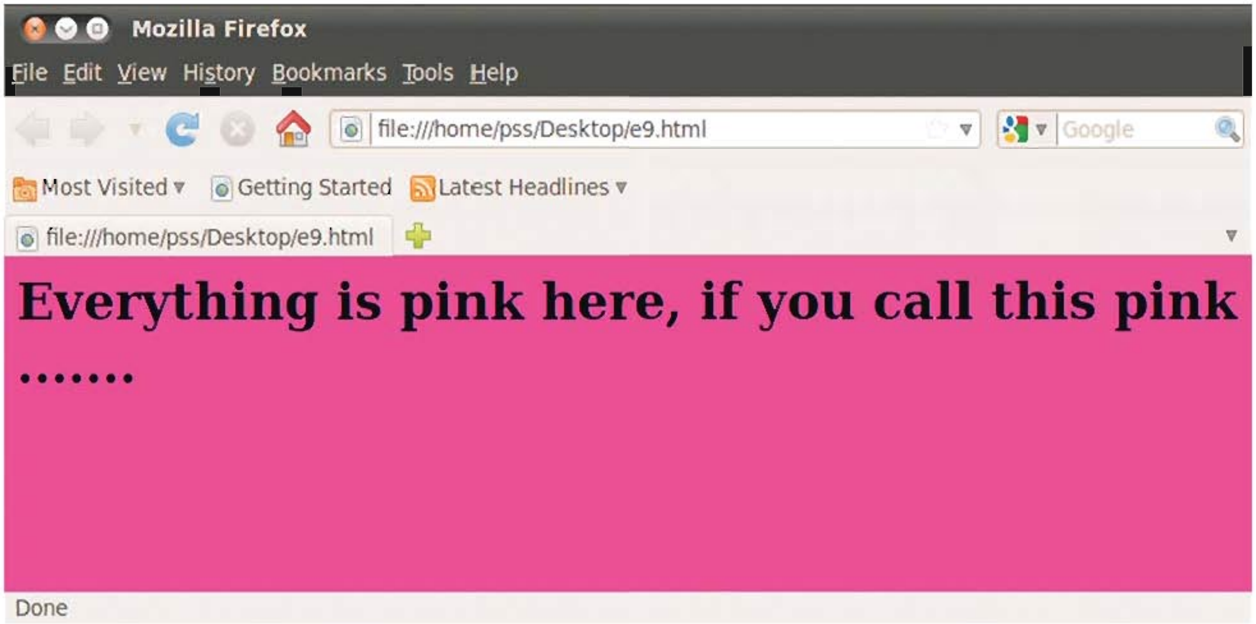
```
<body background="rainbow.jpg" bgcolor="Green">

<body background="rainbow.jpg" bgcolor="Chocolate">
```

રંગોની શબ્દમાળામાં ફેરફાર કરીને તમે અનેકવિધ રંગના બૅકગ્રાઉન્ડ તૈયાર કરી શકો. કોડ-લિસ્ટિંગ 2.1માં દર્શાવેલ HTML કોડનો પ્રયત્ન કરી જુઓ.

```
<html>
    <body bgcolor="#FF00FF">
        <h1>Everything is pink here, if you call this pink .......</h1>
    </body>
</html>
```

કોડ-લિસ્ટિંગ 2.1 : બૅકગ્રાઉન્ડ ઉમેરવા માટેના HTML કોડ

જ્યારે બ્રાઉઝરમાં નિહાળવામાં આવશે ત્યારે સ્ક્રીન આબેહૂબ આકૃતિ 2.6 જેવો દેખાશે.



આકૃતિ 2.6 : બૅકગ્રાઉન્ડ રંગનો ઉપયોગ

લખાણનો રંગ

જેવી રીતે આપણે body ટૅગની સાથે ગુણધર્મ(એટ્રીબ્યુટ)ના ઉપયોગ દ્વારા બૅકગ્રાઉન્ડ રંગ અને છબીને વ્યાખ્યાયિત કરી તેવી જ રીતે, body ટૅગની સાથે text ગુણધર્મ દ્વારા લખાણ(text)ના રંગને વ્યાખ્યાયિત કરી શકીએ. ધારો કે, આપણે નીચે મુજબ એક ઉદાહરણ લખીએ છીએ.

<body text="#FF0000" >

FF0000 કિંમત વડે text ગુણધર્મનો ઉપયોગ કરવાથી લખાણનો રંગ લાલ થશે. ધારો કે, તમારે ગુલાબી રંગના બૅકગ્રાઉન્ડ ઉપર પીળા રંગના અક્ષરોમાં લખાણ દર્શાવવું હોય તો, તમે નીચે મુજબની માત્ર એક સૂચનામાં આમ કરી શકો.

<body bgcolor="#FF00FF" text="#FFFF00">

લિન્કનો રંગ

દર્શાવાતા વેબપેઇજની અંદર ઘણીવાર તમે કોઈ જુદા સ્થાન / પાન પર નિર્દેશ કરતા વાદળી રંગના શબ્દો જોયા હશે. જ્યારે આવા શબ્દ પર ક્લિક કરવામાં આવે તો તે અન્ય પાન પર લઈ જાય છે. જ્યારે તમે મૂળ પાન પર પાછા આવો ત્યારે તમે એ અવલોકન કર્યું હશે કે આ લખાણનો રંગ બદલાઈ જતો હોય છે. તમે લિન્કની મુલાકાત લીધી હોવાને કારણે વેબપેઇજ દ્વારા આવી લિન્કને મુલાકાત લેવાઈ ગયેલ લિન્ક (visited link) તરીકે ઓળખવામાં આવે છે. આવી મુલાકાત લીધેલ લિન્કને vlink તરીકે ઓળખવામાં આવે છે. જ્યારે જે લિન્કની મુલાકાત લેવાય છે, તે લિન્કને સક્રિય લિન્ક (active link) તરીકે ઓળખવામાં આવે છે. ઍક્ટિવ લિન્કને alink તરીકે ઓળખવામાં આવે છે. મુલાકાત લેવાઈ ગયેલ લિન્ક અને સક્રિય લિન્કને જુદી પાડવા જુદા જુદા રંગોનો ઉપયોગ કરાય છે. નીચેનું ઉદાહરણ જુઓ :

<body alink="#00FF00">

<body vlink="#FF0000">

વૈકલ્પિક રીતે તમે આમ પણ લખી શકો.

<body vlink="pink">

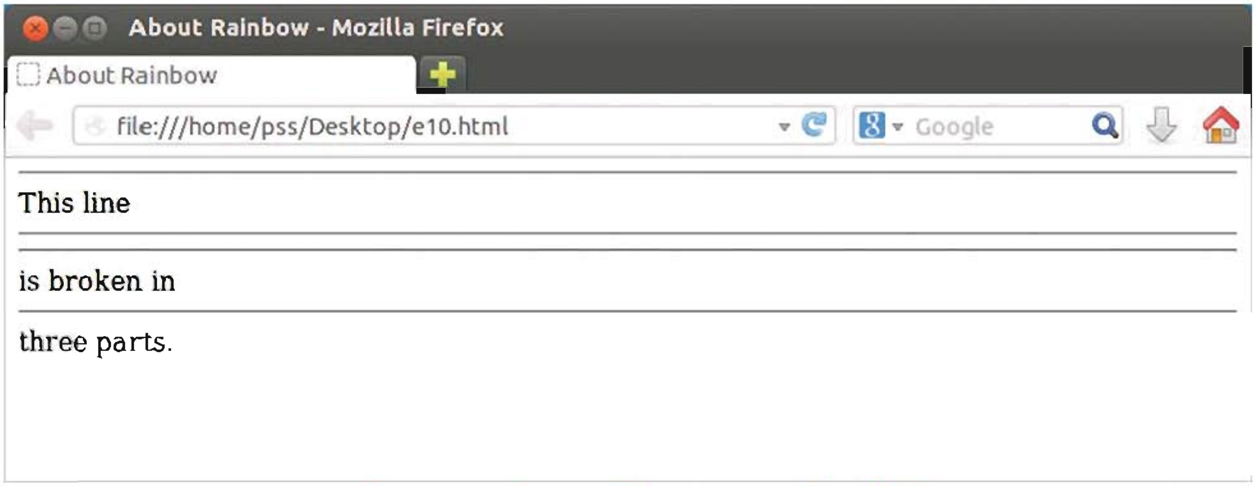
ઉપરના ઉદાહરણમાં ઉપયોગમાં લેવાયેલ alink ગુણધર્મ દસ્તાવેજમાંની સક્રિય લિન્કનો રંગ ગોઠવે છે. સક્રિય લિન્ક જે વખતે ક્લિક કરવામાં આવે તે વખતની લિન્કની પરિસ્થિતિ રજૂ કરે છે. ઉપરના ઉદાહરણમાં ઉપયોગમાં લેવાયેલ vlink ગુણધર્મ દસ્તાવેજની અંદરની હાઈપરલિન્કનો રંગ ગોઠવે છે, જેની મુલાકાત લેવાઈ ગઈ છે.

આડી લીટી (Horizontal line)

<hr /> એલિમેન્ટ પાનાને સમાંતર આડી લીટી દોરે છે.
 એલિમેન્ટની જેમ જ તે ખાલી એલિમેન્ટ છે. લખાણને બે વિભાગમાં વહેંચીને જુદું પાડવા તેનો ઉપયોગ થાય છે. તેને આડી લીટી પણ કહે છે. આકૃતિ 2.7માં દર્શાવેલ ઉદાહરણ જુઓ:

આકૃતિ 2.7 : આડી લીટીનું નિદર્શન

આકૃતિ 2.7માં લખેલ કોડનું પરિણામ આકૃતિ 2.8માં દર્શાવાયું છે.



આકૃતિ 2.8 : બ્રાઉઝર આ રીતે <hr> કમાન્ડને દર્શાવે છે.

આડી લીટી સાથે ઉપયોગમાં લઈ શકાય તેવા પાંચ ગુણધર્મો છે. આ ગુણધર્મો કોષ્ટક 2.2માં આપેલા છે.

ગુણધર્મ	વર્ણન
size	લીટીની જાડાઈ નક્કી કરે છે. તમે 3, 5 કે એથીય વધુ 10નું કદ ધરાવતા પિક્સેલ (pixel)નો ઉપયોગ કરી શકો.
color	Mozilla Firefox જેવા એક્સપ્લોરર (Explorer)માં લીટીનો રંગ નક્કી કરે છે. લીટીનો રંગ સોળ અંકી પદ્ધતિમાં પણ વ્યાખ્યાયિત કરી શકાય છે.
width	લીટીની લંબાઈ નક્કી કરે છે. આ લંબાઈ બ્રાઉઝરની વિન્ડોની પહોળાઈની ટકાવારી તરીકે પણ નક્કી કરી શકાય છે. પહોળાઈની કિંમત સામાન્ય રીતે 100% હોય છે.
align	લીટી જમણી બાજુ, ડાબી બાજુ કે મધ્યમાં ગોઠવાય તે નક્કી કરે છે.
noshade	છાયાંકિત લીટીને બદલે નક્કર સાદી લીટી દર્શાવે છે.

કોષ્ટક 2.2 : આડી લીટીના ગુણધર્મો

ધારો કે, ઉપયોગકર્તા એવું ઇચ્છતો હોય કે 2ના કદની લીટી દર્શાવાય. જે સ્ક્રીનના કદના 25% ભાગ જ ઉપયોગમાં લે અને તે સ્ક્રીનની જમણી બાજુ ગોઠવાયેલી હોય તો નીચે મુજબ ટૅગ લખવો પડે.

```
<hr size=2 width="25%" align="right" >
```

અહીં એ નોંધવું જરૂરી છે કે HTML માં <hr> ટૅગ માટે બંધ થતો ટૅગ લખવો ફરજિયાત નથી. આપણે આડી લીટીને કોઈ છબી વડે પણ ભરી શકીએ. તેને માન્ય HTML કોડમાં સમાવીને બ્રાઉઝર દ્વારા નિહાળી શકીએ.

પૂર્વસ્વરૂપાંકિત લખાણનું ઉદાહરણ (Example of preformatted text)

એવી પરિસ્થિતિની કલ્પના કરો કે, જેમાં આપણે આવનારા કોઈ પ્રસંગ માટેની જાહેરાત દર્શાવવા ઇચ્છીએ છીએ. જાહેરાતનો મુસદ્દો (notice) કોષ્ટક 2.3માં દર્શાવ્યા મુજબનો છે.

Practicing Rainbow Colours

Attention Please !

The rainbow is made up of seven colours. When we mix all seven colours they will become a white colour. To demonstrate this practically, we have arranged a laboratory session. The detail of the laboratory session is as follows:

Date	:	6 August
Time	:	9 : 30 am
Place	:	First Floor, Lab-1
Instructor	:	I M Patel

Principal

કોષ્ટક 2.3 : જાહેરાતનો મુસદ્દો

તો ચાલો, સૌપ્રથમ કોષ્ટક 2.3 સ્વરૂપે આપણને આપવામાં આવેલ માહિતી ઉપર થોડું અવલોકન કરીએ.

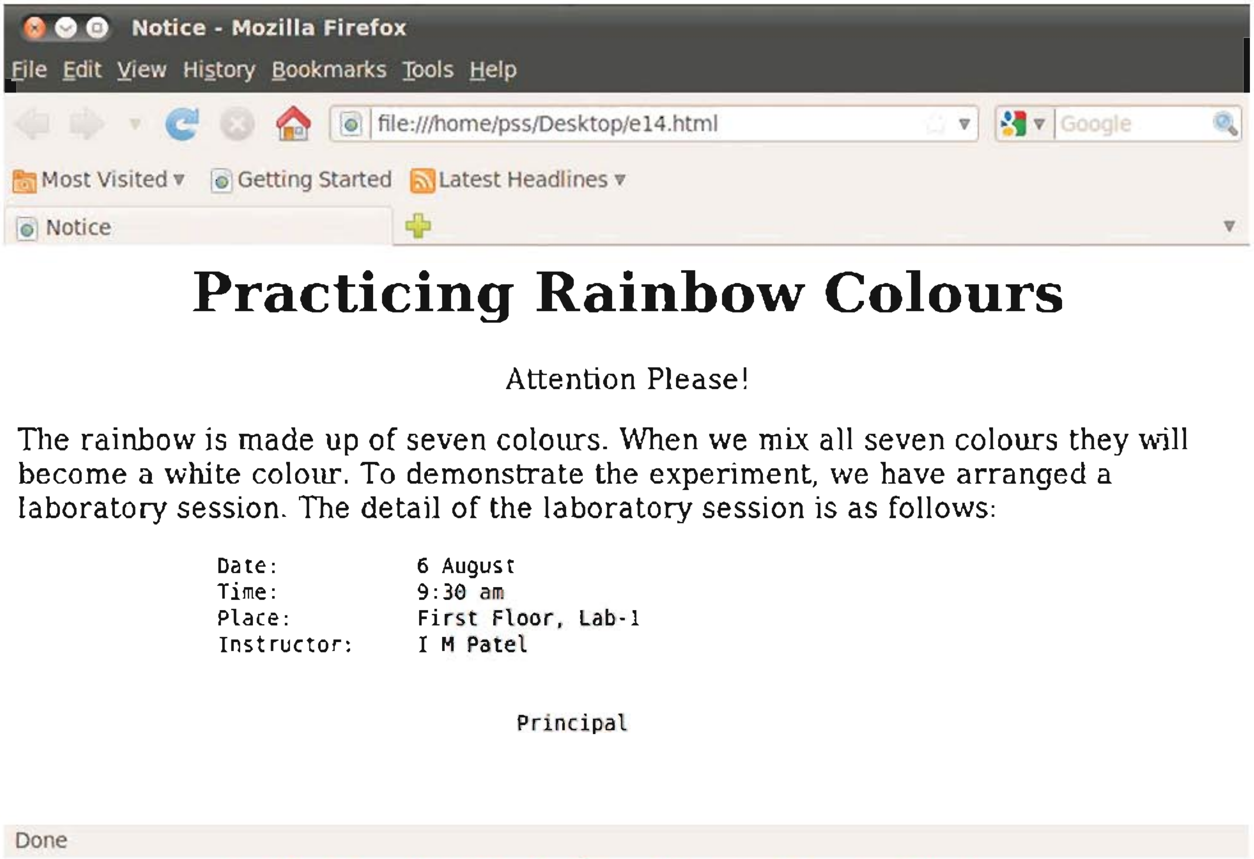
1. કોષ્ટક 2.3માં દર્શાવાયેલ લખાણ HTML દસ્તાવેજના body વિભાગમાં સમાવવું પડે.
2. પ્રથમ લીટી ('Practicing Rainbow Colours') એ મથાળું છે અને મધ્યમાં દર્શાવાયેલ છે. આને યોગ્ય રીતે દર્શાવવા આપણે align="center" ગુણધર્મ સાથે <h1> ટૅગનો ઉપયોગ કરવો પડે.
3. બીજી લીટી ('Attention Please !') એ સાદું લખાણ છે. પરંતુ તે નવી લીટી પર મધ્યમાં દર્શાવાયેલ છે તેથી આને એ રીતે દર્શાવવા આપણે align="center" ગુણધર્મ સાથે <p> ટૅગ વાપરીશું.
4. એ પછી પ્રયોગશાળામાં થનારા પ્રયોગ વિશે માહિતી આપતું સામાન્ય લખાણ છે. આ લખાણને આપણે સાદા paragraph ટૅગ દ્વારા દર્શાવીશું.
5. તારીખ, સમય, સ્થળ અને શિક્ષક વિશેની માહિતી આપતી પછીની ચાર લીટીઓ જુદી રીતે રજૂ કરવામાં આવી છે. આ પૂર્વસ્વરૂપાંકિત લખાણ (preformated text)નું ઉદાહરણ છે. <pre> અને </pre> ટૅગનો ઉપયોગ કરીને આપણે તેને બિલકુલ તેવી જ રીતે દર્શાવી શકીએ.

કોષ્ટક 2.3માં દર્શાવેલ લખાણને રજૂ કરતા HTML કોડ કોડ-લિસ્ટિંગ 2.2માં આપવામાં આવેલ છે.

```
<html>
<head>
<title> Notice </title>
</head>
<body>
<h1 align= "center "> Practicing Rainbow Colours </h1>
<p align =center>    Attention Please! </p>
<p>  The rainbow is made up of seven colours. When we mix all seven colours
     they will become a white colour. To demonstrate the experiment,
     we have arranged a laboratory session. The detail of the laboratory
     session is as follows :
</p>
<pre>
     Date        :  6 August
     Time        :  9 : 30 am
     Place       :  First Floor, Lab-1
     Instructor  :  I M Patel
                          Principal
</pre>
</body>
</html>
```

કોડ-લિસ્ટિંગ 2.2 : જાહેરાત દર્શાવવા માટેના HTML કોડ

કોડ લિસ્ટિંગ 2.2ની સૂચનાઓનું પરિણામ આકૃતિ 2.9માં દર્શાવાયું છે.



આકૃતિ 2.9 : HTMLમાં પૂર્વસ્વરૂપાંકિત (Preformatted) લખાણ

અક્ષરોનું સ્વરૂપાંકન (Formatting Characters)

લખાણને કેવી રીતે ઘાટા સ્વરૂપે (bold), ત્રાંસા અક્ષર સ્વરૂપે (italics) અને લખાણની નીચે લીટી સ્વરૂપે (underline) દર્શાવવા તે આપણે અગાઉના પ્રકરણમાં જોયું. જ્યારે આપણે લખાણને કેવા સ્વરૂપે રજૂ કરવું તે માટેની ઇચ્છિત ઢબ html કોડ સ્વરૂપે આપીશું તો બ્રાઉઝર આ સૂચનાઓને ચુસ્તપણે અનુસરશે અને લખાણને કહ્યા પ્રમાણે રજૂ કરશે. આવા ટૅગ ભૌતિક ઢબ (physical style) ટૅગ તરીકે ઓળખાય છે. કોષ્ટક 2.4માં ભૌતિક ઢબ માટેના ટૅગ રજૂ કરવામાં આવ્યા છે.

ટૅગ	વર્ણન
 ... 	આ ટૅગની વચ્ચે ટાઇપ કરેલ લખાણને ઘાટા સ્વરૂપે દર્શાવે છે.
<i> ... </i>	આ ટૅગની વચ્ચે ટાઇપ કરેલ લખાણને ત્રાંસા અક્ષર સ્વરૂપે દર્શાવે છે.
<strike> ... <strike>	આ ટૅગની વચ્ચે ટાઇપ કરેલ લખાણ જાણે કે લખ્યા પછી છેકી નાંખ્યું હોય તેવા સ્વરૂપે દર્શાવવા લખાણની ઉપર લીટી દર્શાવશે.
_{...}	આ ટૅગની વચ્ચે ટાઇપ કરેલ લખાણને તેની આગળના લખાણ કરતાં નાના અક્ષરોમાં અને થોડું નીચે દર્શાવે છે.
^{...}	આ ટૅગની વચ્ચે ટાઇપ કરેલ લખાણને તેની આગળના લખાણ કરતાં નાના અક્ષરોમાં અને થોડું ઉપર દર્શાવે છે.
<tt> ... </tt>	આ ટૅગની વચ્ચે ટાઇપ કરેલ લખાણને ટાઇપરાઇટર મશીન જેવા ચોક્કસ અક્ષરો સ્વરૂપે દર્શાવે છે.
<u> ... </u>	આ ટૅગની વચ્ચે ટાઇપ કરેલ લખાણની નીચે લીટી દોરાયેલ સ્વરૂપે દર્શાવે છે.

કોષ્ટક 2.4 : ભૌતિક ઢબ માટેના ટૅગ (Phsical Style Tag)

લખાણના સ્વરૂપાંકન માટે અન્ય રસ્તા પણ છે. સ્વરૂપાંકન માટે બ્રાઉઝરને ભૌતિક સૂચનાઓ આપવાને બદલે આપણે બ્રાઉઝરને માત્ર કહેવાનું હોય છે કે, આપણે શું જોઈએ છે ? આ રીતનું સ્વરૂપાંકન કરી આપતી ટૅગને તાર્કિક સ્ટાઈલ માટેના ટૅગ (logical style tag) કહે છે. નીચે આપેલા ટૅગ એ કેટલાંક તાર્કિક સ્ટાઈલના ટૅગનાં ઉદાહરણ છે.

 અને

આ ટૅગની જોડી દ્વારા લખાણને ભારપૂર્વક દર્શાવવામાં આવે છે. મહત્ત્વની બાબત જેવી કે "કરવાં જ પડનારાં કાર્યો"(Must be done) અને "અગત્યનું" (Important) આ ટૅગનો ઉપયોગ કરીને લખી શકાય. આ ટૅગનું ઉદાહરણ નીચે મુજબ છે:

What a beautiful rainbow...!

 અને

ટૅગની આ જોડી દ્વારા લખાણને વધુ ભારપૂર્વક રજૂ કરવામાં આવશે. આ ટૅગનું ઉદાહરણ નીચે મુજબ છે.

** What a beautiful rainbow...!**

ઉપરના ઉદાહરણ દ્વારા <em> અને <strong> દ્વારા દર્શાવાયેલ લખાણ વચ્ચેનો તફાવત જુઓ. આવા તાર્કિક સ્વરૂપાંકન કરતા ટૅગ કોષ્ટક 2.5માં આપવામાં આવ્યા છે.

ટૅગ	વર્ણન
<dfn>	લખાણને નિર્ધારિત ઢબે રજૂ કરે છે.
<em>	કોઈ ચોક્કસ લખાણ પર ભાર મૂકવા માટે
<cite>	પુસ્તકનું શીર્ષક, ચલચિત્ર વગેરે જેવા મહત્ત્વના લખાણને રજૂ કરવા.
<code>	કમ્પ્યૂટર પ્રોગ્રામિંગ કોડ દર્શાવવા આ લખાણ ચોક્કસ કદના ફોન્ટમાં દર્શાવાશે.
<strong>	આ ટૅગ લખાણને વધુ ભારપૂર્વક દર્શાવે છે. લખાણ વધુ ઘાટા અક્ષર સ્વરૂપે દર્શાવાય છે.

કોષ્ટક 2.5 : તાર્કિક સ્વરૂપાંકન ટૅગ (Logical Style tag)

Font ટૅગ

અક્ષરોના ચોક્કસ જાતના મરોડ અને કદ ગોઠવવા font ટૅગનો ઉપયોગ કરવામાં આવે છે. કેટલાંક ઉદાહરણ નીચે આપવામાં આવેલ છે.

```
<p><font size="3" color="red">This is some text!</font></p>

<p><font size="2" color="blue">This is some text!</font></p>

<p><font face="verdana" color="green">This is some text!</font></p>
```

અહીં તમે એ નોંધ્યું હશે કે, આપણે સોળઅંકી સંખ્યાને બદલે green, red, blue એમ રંગના સીધા નામનો ઉપયોગ કરેલ છે. ઉપરના ઉદાહરણમાં font ટૅગની સાથે ગુણધર્મોનો પણ ઉપયોગ કર્યો છે. આ ગુણધર્મો છે size અને color. જેમાં size એ લખાણમાં અક્ષરોનું કદ નક્કી કરે છે અને color એ લખાણ ક્યા રંગના અક્ષરોમાં દર્શાવાય તે નક્કી કરે છે.

HTML કોડની શરૂઆતમાં જ આખા વેબપેઇજ માટે font અને તેના વિવિધ ગુણધર્મો નક્કી કરી દેવાનું શક્ય છે. આવું કરવા નીચે મુજબ લખવું પડેઃ

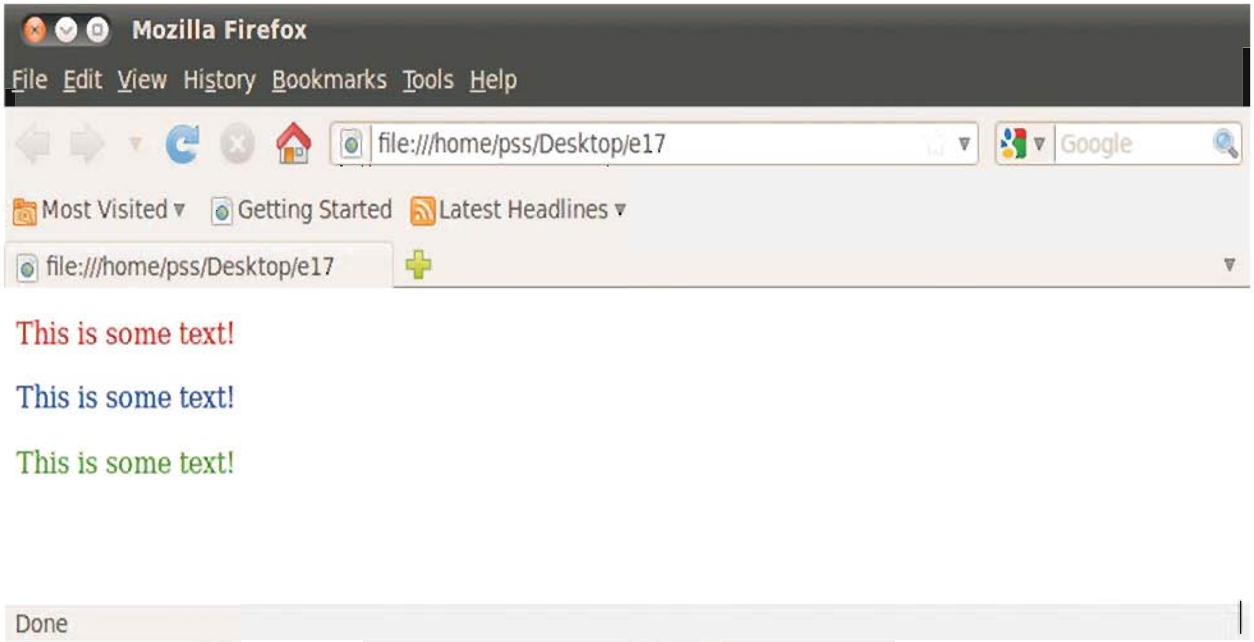
```
<basefont face="Arial" size="16">
```

કોડ લિસ્ટિંગ 2.3માં દર્શાવ્યા પ્રમાણે પૂરા HTML કોડ લખીને ઉપરોક્ત કોડને ચકાસી શકાય.

```
<html>

<body>
        <basefont face = Arial size=16>
        <p><font color="red">This is some text!</font></p>
        <p><font color="blue">This is some text!</font></p>
        <p><font color="green">This is some text!</font></p>
</body>

</html>
```

કોડ લિસ્ટિંગ 2.3 : font ટૅગને ચકાસવા માટે HTML કોડનો નમૂનો

ઉપરોક્ત કોડને ટાઇપ કરી બ્રાઉઝર દ્વારા નિહાળવાનો પ્રયત્ન કરો. આ HTML કોડનું પરિણામ આબેહૂબ આકૃતિ 2.10માં દર્શાવ્યા મુજબ દેખાશે.



આકૃતિ 2.10 : આધારરૂપ ફૉન્ટ અને રંગ

નોંધ :

ધ્યાનથી જુઓ આકૃતિ 2.10માં દેખાતું ફાઈલનું નામ e17.htmlને બદલે e17 છે. સામાન્ય રીતે લિનક્સ ફાઈલના અનુલંબનનો ઉપયોગ કરતું નથી. જ્યારે ફાઈલના નામની પાછળ અનુલંબનનો ઉપયોગ કરવામાં આવે છે ત્યારે તે ફાઈલની અંદરના લખાણ અથવા તેનો ઉપયોગ દર્શાવે છે. ઉદાહરણ તરીકે આપણા કિસ્સામાં HTML અનુલંબન એવું કહે છે કે આ HTML પ્રકારની ફાઈલ છે. ઈચ્છિત પરિણામ મેળવવા લિનક્સ પર ઉપયોગકર્તાએ ફાઈલનું અનુલંબન દર્શાવવાની જરૂર રહેતી નથી. માત્ર જે તે ફાઈલને ખોલવા યોગ્ય પ્રોગ્રામનો ઉપયોગ કરાયો છે કે કેમ તેની ખાતરી કરવાની રહે છે.

ખાસ ચિહ્નો

ખાસ ચિહ્નો જેવાં કે, <, &, ©, વગેરે ચિહ્ન પૂર્વગ તરીકે & (એમ્પરસન્ડ ચિહ્ન) સાથે વેબપેઇજ પર સમાવી શકાય છે. આવી પ્રતીકાત્મક સંજ્ઞાને બદલે આપણે જે તે ચિહ્ન માટેની ASCII કિંમત પણ રાખી શકીએ. આની જરૂર એટલા માટે પડે છે કેમ કે, HTML દસ્તાવેજમાં સીધેસીધા '<' અને '>' ચિહ્નનો ઉપયોગ તો કરી ન શકાય. કારણ કે, તે તો ટૅગ સાથે ભળી જવાની શક્યતા છે. કોષ્ટક 2.6માં દર્શાવેલ ખાસ ચિહ્નોનો ઉપયોગ કરો.

ચિહ્ન	વર્ણન	સાંકેતિક સંજ્ઞા	કિંમત
<	ના કરતાં નાનું (Less than)	lt	60
>	ના કરતાં મોટું (Greater than)	gt	62
&	એમ્પરસન્ડ (Ampersand)	amp	38
©	કૉપીરાઇટ (Copy right)	–	169
¼	એક ચતુર્થાંશ (One quarter)	–	188
½	અડધું (Half)	–	189
¾	ત્રણ ચતુર્થાંશ (Three quarters)	–	190
®	રજિસ્ટર્ડ ટ્રેડમાર્ક (Registered trademark)	erg	174

કોષ્ટક 2.6 : ખાસ ચિહ્ન માટે સાંકેતિક સંજ્ઞા અને ASCII કિંમત

HTML કોડમાં, ના કરતાં ઓછું (<) અને ના કરતાં વધુ (>) ચિહ્નના ઉપયોગને રજૂ કરતું ઉદાહરણ...

```
<p align="center"> &lt Basic Information &gt</p>
```

ઉપરોક્ત કોડ વેબપેઇજ પર નીચે મુજબ લખાણ દર્શાવશે:

< Basic Information >

આ પ્રકરણમાં આવરી લેવાયેલ ટૅગ

આ પ્રકરણમાં આપણે કોષ્ટક 2.7માં દર્શાવેલ ટૅગની ચર્ચા કરી છે.

ટૅગ	વર્ણન
<!-- ...-->	HTML કોડમાં ટિપ્પણીના લખાણને ઉમેરવા
<b>	લખાણને ઘાટા અક્ષરોમાં દર્શાવે છે.
<base>	વેબપેઇજમાં તમામ સંબંધિત URL માટે આધારરૂપ URL / ગંતવ્ય દર્શાવે છે.
<basefont>	આખા વેબપેઇજ માટેના સામાન્ય (default) ફૉન્ટ વ્યાખ્યાયિત કરે છે.
<body>	HTML દસ્તાવેજ માટે body વિભાગ વ્યાખ્યાયિત કરે છે. તેની સાથે background image, background colour, text colour, link colour વગેરે ગુણધર્મ ઉપલબ્ધ બને છે.

ટેગ	વર્ણન
<cite>	પુસ્તકનું શીર્ષક, ફિલ્મનું નામ વગેરે જેવા મહત્ત્વના શબ્દોને રજૂ કરવા આવું લખાણ ત્રાંસા અક્ષર (italics) સ્વરૂપે દર્શાવાય છે.
<code>	કમ્પ્યૂટર પ્રોગ્રામિંગ કોડના વિભાગને રજૂ કરે છે. આવું લખાણ નિયત કદના ફૉન્ટમાં દર્શાવાય છે.
<dfn>	લખાણને નિર્ધારિત સ્ટાઈલે રજૂ કરવા આવું લખાણ ત્રાંસા અક્ષર (italics) સ્વરૂપે દર્શાવાય છે.
<em>	લખાણ ભારપૂર્વક રજૂ કરાય છે.
<font>	ચોક્કસ પ્રકારના અને કદના ફૉન્ટ ગોઠવવા ફૉન્ટ ટૅગ ઉપયોગમાં લેવાય છે.
<head>	HTML દસ્તાવેજનો Head વિભાગ દર્શાવવા.
<hr>	આડી લીટી વ્યાખ્યાયિત કરે છે. તેની સાથે size, colour, width, alignment, noshade વગેરે જેવા ગુણધર્મો હોય છે.
<i>	લખાણને ત્રાંસા અક્ષર સ્વરૂપે રજૂ કરે છે.
<link>	દસ્તાવેજ અને બાહ્ય સંસાધન વચ્ચે સંબંધ વ્યાખ્યાયિત કરે છે.
<meta>	સર્ચ એન્જિન અને અન્ય યુટીલીટી પ્રોગ્રામને author, keywords, description, purpose વગેરે વિષે વધારાની માહિતી પૂરી પાડે છે.
<script>	ક્લાયન્ટ બાજુની સ્ક્રિપ્ટ વ્યાખ્યાયિત કરે છે.
<strike>	લખાણને જાણે કે લખ્યા પછી છેકી નાંખ્યુ હોય તેવી રીતે દર્શાવે છે.
<strong>	લખાણને વધુ ભારપૂર્વક દર્શાવવામાં આવે છે.
<style>	દસ્તાવેજ માટે સ્ટાઈલ અંગેની માહિતી વ્યાખ્યાયિત કરે છે.
<sub>	આ ટૅગની વચ્ચે ટાઇપ કરેલ લખાણને તેની આગળના લખાણ કરતાં નાના અક્ષરોમાં અને થોડું નીચે દર્શાવે છે.
<sup>	આ ટૅગની વચ્ચે ટાઇપ કરેલ લખાણને તેની આગળના લખાણ કરતાં નાના અક્ષરોમાં અને થોડું ઉપર દર્શાવે છે.
<tt>	આ ટૅગની વચ્ચે ટાઇપ કરેલ લખાણને ટાઇપરાઇટર મશીન જેવા ચોક્કસ અક્ષરો સ્વરૂપે દર્શાવે છે.
<u>	આ ટૅગની વચ્ચે ટાઇપ કરેલ લખાણની નીચે લીટી દોરાયેલ સ્વરૂપે દર્શાવે છે.

કોષ્ટક 2.7 : પ્રકરણ 2માં ચર્ચેલ HTML ટૅગ

એક ઉદાહરણ

તો ચાલો, રોજિંદા જીવનનું એક ઉદાહરણ લઈએ. ધારો કે, તમારે તમારી શાળાની એક સાદી વેબસાઇટ બનાવવી છે. વેબસાઇટ શાળાના સંસાધનો અને પ્રવૃત્તિઓની રજૂઆત કરતી હોય. શાળાની વેબસાઇટનું મુખ્ય પાનું અનુક્રમણિકા (index) અથવા હોમપેજ તરીકે ઓળખી શકાય. પ્રથમ વેબપેઇજ પર શાળાનું નામ પ્રકાશિત થશે. આ હોમપેજ પર શાળાનું ટ્રસ્ટ સાથેનું જોડાણ, નોંધણી નંબર, સંપૂર્ણ સરનામું અને સંપર્કસૂત્રની વધારાની માહિતી પણ મૂકી શકાય. તદ્ઉપરાંત હોમપેજને અન્ય વિવિધ વેબપેઇજ જેવાં કે, 'શૈક્ષણિક પ્રવૃત્તિ', 'ફોટો ગેલેરી', 'કાર્યક્રમો અને પ્રસંગો' વગેરે સાથે જોડાણ (લિન્ક) હોઈ શકે.

આ વેબપેઇજનું વર્ણન નીચે મુજબ આપેલ છે:

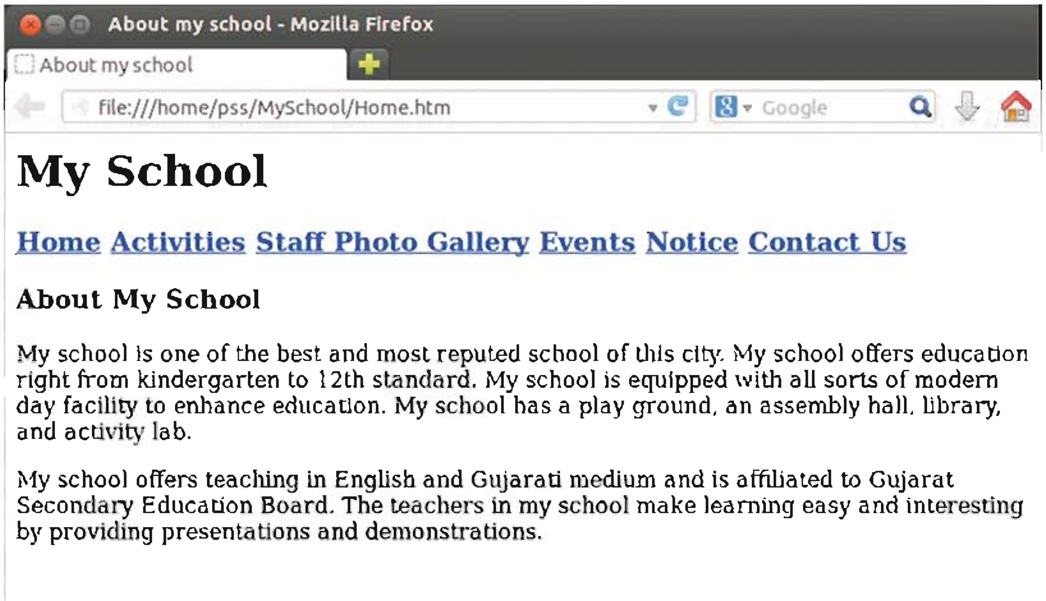
- મારી શાળાનું હોમ પેજ : શાળાની વેબસાઈટના મુખ્ય પાના પર અન્ય પાનાના મેનૂ ઉપરાંત શાળાનું નામ, શાળાના ટ્રસ્ટની ટૂંકમાં માહિતી, શાળાનો નોંધણી નંબર, કોઈ સરકારી વિભાગ સાથે શાળાનું જોડાણ અને શાળાના સંપર્ક માટેની જરૂરી વિગતોનો સમાવેશ થશે.
- શિક્ષકગણ અને શૈક્ષણિક પ્રવૃત્તિ : આ પાના પર શિક્ષકગણનો ટૂંકો પરિચય રજૂ થશે. જેમાં દરેક શિક્ષકનું નામ, શૈક્ષણિક લાયકાત, કયા વર્ગમાં તેઓ શીખવે છે તે, અને તેમનું વૈશિષ્ટીકરણ તેમજ તેમનું ઇ-મેઇલ સરનામું હશે.
- ફોટો ગેલરી : આ પાનું વર્ગના ફોટા, ગ્રંથાલય, પ્રયોગશાળા અને તાજેતરના પ્રસંગો દર્શાવે છે. જ્યારે છબીઓનું સંચાલન કરવાનું શીખી જાઓ ત્યારબાદ પછીથી આ વેબપેઇજ બનાવી શકો છો.
- પરિપત્રો અને સૂચનાપત્રો : આ પાનું હવે પછી યોજાનારા કાર્યક્રમો અને પ્રસંગો વિશેની માહિતી ધરાવે છે. તેમાં વિદ્યાર્થીઓ અને વાલીઓ માટેના પરિપત્રો અને સૂચનાપત્રો પણ મૂકાશે. ઉદાહરણ તરીકે, રમતોત્સવ, નિબંધસ્પર્ધા અંગેની સૂચના, શિષ્યવૃત્તિ કાર્યક્રમની જાહેરાત અને પરિણામ.

કોડ-લિસ્ટિંગ 2.4માં આપેલ HTML કોડ જુઓ.

```
<html>
<head>
     <title>About my school</title>
</head>
<body vlink="purple" >
<h1> My School   </h1>

<p>  <h3>
      <a href="Home.html">Home</a>
      <a href="Activities.html">Activities</a>
      <a href="Staff.html"> Staff </a>
      <a href="Gallery.html">Photo Gallery</a>
      <a href="Event.html">Events</a>
      <a href="Notice.html">Notice</a>
      <a href="Contact.html">Contact Us</a>
      </h3>
     </p>
     <h3>About My School</h3>

<p>  My school is one of the best and most reputed school of this city. My school
     offers education right from kindergarten to 12th standard. My school is equipped
     with all sorts of modern day facility to enhance education. My school has a
     play ground, an assembly hall, library, and activity lab. </h3>
</p>

<p>  My school offers teaching in English and Gujarati medium and is affiliated to
     Gujarat Secondary Education Board. The teachers in my school make learning
     easy and interesting by providing presentations and demonstrations. </h3>
</p>

</body>
</html>
```

કોડ-લિસ્ટિંગ 2.4 : My School વેબસાઈટના મુખ્ય પાના માટેના HTML કોડ

એડિટરનો ઉપયોગ કરી કોડ-લિસ્ટિંગ 2.4માં દર્શાવેલ HTML કોડ ટાઇપ કરો. તેને Home.html તરીકે સાચવો. આ કોડને બ્રાઉઝરમાં નિહાળો. એનું પરિણામ આબેહૂબ આકૃતિ 2.11માં દર્શાવ્યા મુજબનું દેખાશે.



આકૃતિ 2.11 : My School વેબસાઇટનું હોમપેજ

જો તમે Home સિવાયની કોઈ પણ લિન્ક પર ક્લિક કરશો તો ફાઈલ મળતી ન હોવા અંગેના ભૂલ સંદેશ દર્શાવાશે. આ ભૂલસંદેશ અત્યારે એટલા માટે દર્શાવાય છે, કારણ કે આપણે Home.html સિવાય અન્ય કોઈ ફાઈલ તૈયાર કરેલ નથી.

આવી ફાઈલો તૈયાર કરવા માટે નીચે આપેલાં પગલાંને અનુસરો :

- અન્ય ફાઈલ જેવી કે, activities, staff, photo gallery, notice અને contact માટેના HTML કોડ લખો. તેમને યોગ્ય નામ આપી તમે જ્યાં home.html ફાઈલને સાચવેલ છે તે જ ડિરેક્ટરીમાં બધાનો સંગ્રહ કરો. photo gallery માટેનું વેબપેઇજ બનાવવાનું તમે અત્યારે છોડી દઈ શકો છો. હવે પછીના પ્રકરણમાં જ્યારે તમે HTML દસ્તાવેજમાં છબીઓને સમાવવાનું શીખી જાવ એ પછી આ ઉદાહરણરૂપ પ્રૉજેક્ટને સુધારવા પ્રયત્ન કરજો. એ પછી તમે તમારી શાળાનું પ્રતીક (logo) પણ તમે ઉમેરી શકો. આ તબક્કે તમે માત્ર વેબપેઇજના બૅકગ્રાઉન્ડ તરીકે કોઈ ચિત્ર કે છબી મૂકવા સમર્થ છો.
- "Practicing Rainbow Colours" વિશે કોડ લિસ્ટિંગ 2.2માં આપેલ માર્ગદર્શન અનુસાર તમે એક સૂચનાપત્ર (notice) તો તૈયાર કરી જ દીધેલ છે. તમે આ ફાઈલનું નામ બદલીને Notice.html તરીકે સાચવી શકો.
- ફોર્મેટિંગ ટૅગ ઉપરાંત જરૂરી છબીઓ, બૅકગ્રાઉન્ડ અને લિન્કનો ઉપયોગ કરો. તમે લખાણને સુધારી પણ શકો. ઉદાહરણ તરીકે, હોમપેજ પર તમે તમારી શાળાનું નામ, શાળાના ટ્રસ્ટનું નામ, શાળાનો સ્થાપના દિન, શાળાના સંપર્ક માટેની માહિતી વગેરે લખી શકો.

સારાંશ

આ પ્રકરણમાં HTML દસ્તાવેજનાં head વિભાગમાં ઉપયોગમાં લેવાતા મેટા-ટૅગ (meta-tag) વિશે શીખ્યા. મેટા-ટૅગ એમની રીતે કંઈ પણ દર્શાવે નહીં, તેમ છતાં તે લેખક, ચાવીરૂપ શબ્દો, પાના વિશેનું વર્ણન સંબંધી માહિતી પ્રદાન કરે છે. આવી માહિતી સર્ચએન્જિન તથા અન્ય યુટિલિટી પ્રોગ્રામને ખૂબ જ મદદરૂપ નીવડી શકે છે. Head વિભાગની જેમ, HTML દસ્તાવેજના Body વિભાગમાં ઉપયોગમાં લેવાતા ટૅગ અને ગુણધર્મોની પણ ચર્ચા કરી. HTMLનું body વિભાગ ટૅગની ગોઠવણી સહિત ભૌતિક અને તાર્કિક સ્વરૂપાંકન ટૅગ અને ખાસ ચિહ્નો દર્શાવવાની સવલત પણ આપે છે. આપણે HTML દસ્તાવેજમાં ટિપ્પણી સર્વસામાન્ય (ડિફૉલ્ટ) ફૉન્ટ અને રંગ ઉમેરવાની રીત જોઈ. અંતમાં, આપણે એક નમૂનારૂપ શાળાની વિવિધ પ્રવૃત્તિઓ વર્ણવતા ઘણા વેબપેઇજઓ ધરાવતી એક નાની વેબસાઈટ પણ બનાવી.

સ્વાધ્યાય

1. કોઈ HTML દસ્તાવેજ માટે head વિભાગ અને તેમાં ઉપયોગમાં લેવાતા મેટા-ટૅગ વિશે ટૂંક નોંધ લખો.
2. HTML ટૅગનો ઉપયોગ કરી કેવી રીતે કોઈ છબીને વેબપેઇજ પર બૅકગ્રાઉન્ડ તરીકે ઉમેરી શકાય, તે વર્ણવો.
3. ઈલેક્ટ્રૉનિક માધ્યમોમાં રંગોની રજૂઆત વર્ણવો. "કલર કોડિંગ સ્કીમ" વિશે પણ ટૂંકમાં વર્ણવો.
4. HTMLમાં આડી લીટી માટેના <hr /> એલિમેન્ટની ચર્ચા કરો.
5. HTMLમાં font ટૅગ વર્ણવો.
6. HTML દ્વારા ખાસ ચિહ્નો કેવી રીતે રજૂ કરાશે તે વર્ણવો.
7. નીચે આપેલ પ્રશ્નો માટે આપેલા વિકલ્પોમાંથી યોગ્ય વિકલ્પ પસંદ કરી જવાબ આપો:

(1) નીચેનામાંથી ક્યા બે, HTML કોડના બે મૂળભૂત વિભાગો બનાવે છે ?

(a) Head અને Body (b) Physical અને Logical

(c) Code અને Browser (d) Meta-tags અને Body

(2) HTML દસ્તાવેજમાં નીચેનામાંથી ક્યા વિભાગમાં મેટા-ટૅગ લખવામાં આવે છે ?

(a) Body (b) Code

(c) Head (d) Special

(3) HTML પાનામાં જ્યારે મેટા-ટૅગ ઉપયોગમાં લેવાય ત્યારે, વેબપેઇજ વિશે નીચેનામાંથી શું વર્ણવવામાં આવે છે ?

(a) લેખક, ઉદ્દેશ્ય અને ચાવીરૂપ શબ્દો (b) માળખું

(c) સ્ટાઈલ (d) કદ

(4) વેબપેઇજનું શીર્ષક નીચેનામાંથી ક્યા ટૅગમાં આવરીને દર્શાવાય છે ?

(a) <p> અને </p> (b) <body> અને </body>

(c) <title> અને </title> (d) <h1> અને </h1>

(5) HTML દસ્તાવેજમાં નીચેનામાંથી ક્યા વિભાગમાં આપણે ટિપ્પણી ઉમેરી શકીએ ?

(a) Head (b) Body

(c) Head અને body બંને (d) Head અથવા body

(6) HTML કોડમાં રંગ દર્શાવવા માટે નીચેનામાંથી શું ઉપયોગમાં લેવાય છે ?

(a) સોળઅંકી પદ્ધતિમાં કલરકોડ (b) દશાંશ પદ્ધતિમાં કલરકોડ

(c) કલર મિક્સિંગ મૉડેલ (d) ટકાવારીમાં પિક્સેલ

(7) HTML કોડમાં મુલાકાત લેવાઈ ગયેલ (visited) લિન્ક ગોઠવવા નીચેનામાંથી શું ઉપયોગમાં લેવાય છે ?

(a) alink (b) vlink

(c) before link (d) after link

(૮) HTML કોડમાં ઍક્ટિવ લિન્ક ગોઠવવા માટે નીચેનામાંથી શું ઉપયોગમાં લેવાય છે ?

(a) alink (b) vlink

(c) before link (d) after link

(૯) HTMLમાં અક્ષરોના સ્વરૂપાંકન માટેના નીચેનામાંથી ક્યા બે મુખ્ય પ્રકારો છે ?

(a) ભૌતિક અને તાર્કિક (b) આંતરિક અને બાહ્ય

(c) કામચલાઉ અને કાયમી (d) Head અને Body

પ્રાયોગિક સ્વાધ્યાય

1. પ્રકરણમાં ચર્ચા કરેલ 'My School' વેબસાઇટ બનાવવાનો પ્રોજેક્ટ પૂરો કરો.

2. અન્ય શૈક્ષણિક સંસ્થાઓની વેબસાઇટ જુઓ અને તેની રચનાનો અભ્યાસ કરો. આ રચનાઓ માટે ટિપ્પણીઓ લખો.

3. તમારા વિશેનું વેબપેઇજ તૈયાર કરો. તમારા વિશે, તમારા પરિવાર વિશે, તમારા મિત્રો, તમારા શૈક્ષણિક ધ્યેય અને સંપર્ક માટે સરનામાનું વર્ણન કરો. વેબપેઇજની રચના તમે નીચે મુજબ વિચારી શકો.

I, me and myself…!

About Me My Family My Friends My Goal Contact Me

…. …. ….. ….

…… you may write about yourself here….

….. ….. …………… …..

અહીં બીજી લીટી પર નીચે લીટી દોરીને દર્શાવેલ શબ્દો એ હાઇપર લિન્ક છે, જે તમને જુદા પાન પર સાંકળશે. સક્રિય (ઍક્ટિવ) લિન્ક અને મુલાકાત લેવાઈ ગયેલ (visited) લિન્ક ગોઠવો. આછા રંગનું રંગીન બૅકગ્રાઉન્ડ બનાવો. જે રીતે અગાઉ આપણે 'My School' નામની વેબસાઇટ બનાવી તે પ્રમાણે આ એક સંપૂર્ણ વેબસાઇટ બનવી જોઈએ.

4. http://www.w3schools.com/ પર જાઓ. મેનુમાંથી HTML વિભાગ શોધી કાઢો અને "try it yourself" ને ચકાસી જુઓ. આ એક એડિટર છે જે તમારા HTML કોડને લઈને તે જાણે કે બ્રાઉઝરમાં નિહાળતા હોય તેવું દૃશ્ય રજૂ કરે છે.

HTMLમાં છબીઓનું વ્યવસ્થાપન

એવું કહેવાય છે કે, હજાર શબ્દો કરતાં એક ચિત્ર વધુ કહી જાય છે. ચિત્ર અથવા છબી મોટાભાગના વ્યક્તિઓને આકર્ષે છે અને પ્રેક્ષકોને અગત્યનો સંદેશ પહોંચાડે છે. જો વેબપેઇજમાં ચિત્ર કે છબી દર્શાવાશે તો ઉપયોગકર્તા તરત જ તેના પર ધ્યાન આપશે. વળી, એવું અવલોકન છે કે સામાન્ય રીતે ચિત્ર કે છબી સમજ અને સ્વીકૃતિની માત્રામાં વધારો કરે છે.

HTMLમાં <img> ટૅગનો ઉપયોગ કરીને વેબપેઇજમાં છબીઓ ઉમેરવામાં આવે છે. અગાઉના પ્રકરણમાં આપણે પણ બૅકગ્રાઉન્ડ તરીકે એક છબી ઉમેરી છે. વેબપેઇજ ઉપર છબી ઉમેરવાના બીજા પણ ઘણા રસ્તા છે. ઉદાહરણ તરીકે, નીચેની સૂચના વેબપેઇજમાં rainbow.jpg નામની છબી ઉમેરશે.

```
<img src="rainbow.jpg">
```

ઉપરોક્ત ઉદાહરણમાં src ગુણધર્મનો ઉપયોગ કરાયો છે, જેનું નીચે <img> ટૅગ સાથે ઉપયોગમાં લઈ શકાતા અન્ય ગુણધર્મોની સાથે વર્ણન કરવામાં આવ્યું છે.

image એલિમેન્ટ(image ટૅગ, છબી અને ગુણધર્મો સાથે)માં નવી લીટી (line break) આવતી નથી. માટે તે ઇનલાઇન એલિમેન્ટ (Inline Element) તરીકે ઓળખાય છે.

src ગુણધર્મ

src ગુણધર્મ છબીના સ્થાન અંગેની માહિતી પૂરી પાડે છે. અહીં સ્થાન એટલે છબીનું ઉદ્‌ગમસ્થાન. તે બ્રાઉઝરને જણાવે છે કે છબીની ફાઈલ ક્યાં પડેલી છે, એટલે કે તેને ક્યાં શોધવી. ઘણીવાર ઇમેજના સ્થાન માટે URLનો ઉપયોગ કરીએ છીએ. છબીનું મૂળ સ્થાન દર્શાવ્યા વગર બ્રાઉઝર માટે છબીને શોધવી અને દર્શાવવી અશક્ય બની જાય છે. "http://pritisajja.info/images/img1.jpg" તરીકે આપેલ URL પરથી છબીને સામેલ કરવા આપણે નીચે મુજબની ટૅગ લખી શકીએ.

```
<img src="http://pritisajja.info/images/img1.jpg">
```

આ રીતે દર્શાવવામાં આવેલ છબી ફાઈલ યોગ્ય છબી સ્વરૂપ જેમ કે, bmp, gif, tiff અથવા jpg સ્વરૂપની હોવી જોઈએ.

એક એવી પરિસ્થિતિની કલ્પના કરો કે, જેમાં તમે ઇન્ટરનેટમાં માહિતી જોઈ રહ્યા હો અને તમને કોઈ છબી ગમી જાય છે. તો કોઈ HTML કોડમાં તેને સામેલ કરવા તેનું સંપૂર્ણ URL તમે કેવી રીતે મેળવશો ? જવાબ એકદમ સરળ છે. પ્રથમ વિકલ્પ છે, તે છબી પર માઉસનું જમણું બટન ક્લિક કરી image linkની કૉપી કરવી અને બીજો વિકલ્પ છે જમણું બટન ક્લિક કરી છબીને તમારા કમ્પ્યૂટરની હાર્ડડિસ્કમાં સાચવી લેવી. બીજા વિકલ્પ મુજબ અમલ કરતાં પહેલાં છબીના કૉપીરાઇટ બાબતે સચેત રહેવું જરૂરી છે.

જ્યારે લખાણ સાથે <p> અથવા <h1> જેવા ટૅગનો ઉપયોગ કરાય ત્યારે આ ટૅગનું કામ માત્ર લખાણને ચોક્કસ ઢબે દર્શાવવાનું હોય છે, જ્યારે image ટૅગનો ઉપયોગ કરવામાં આવે ત્યારે, આપણે કોઈ લખાણ આપવાનું હોતું નથી, માત્ર છબીનું મૂળ સ્થાન અને સાથે કેટલાક જરૂરી ગુણધર્મો જ દર્શાવવાના હોય છે. એટલા માટે જ image ટૅગને ખાલી ટૅગ કહે છે. જો કે, જ્યારે બ્રાઉઝરમાં છબી ધરાવતું કોઈ HTML દસ્તાવેજ દર્શાવાય ત્યારે બ્રાઉઝરે તે છબીને પ્રાપ્ત કરવી જરૂરી છે. છબી તમારા સ્થાનિક કમ્પ્યૂટર અથવા સર્વરમાં ઉપલબ્ધ હોવી જરૂરી છે. જો તમે URLનો ઉપયોગ કર્યો હોય તો તમારું ઇન્ટરનેટ જોડાણ ચાલુ છે કે કેમ તે ચકાસો.

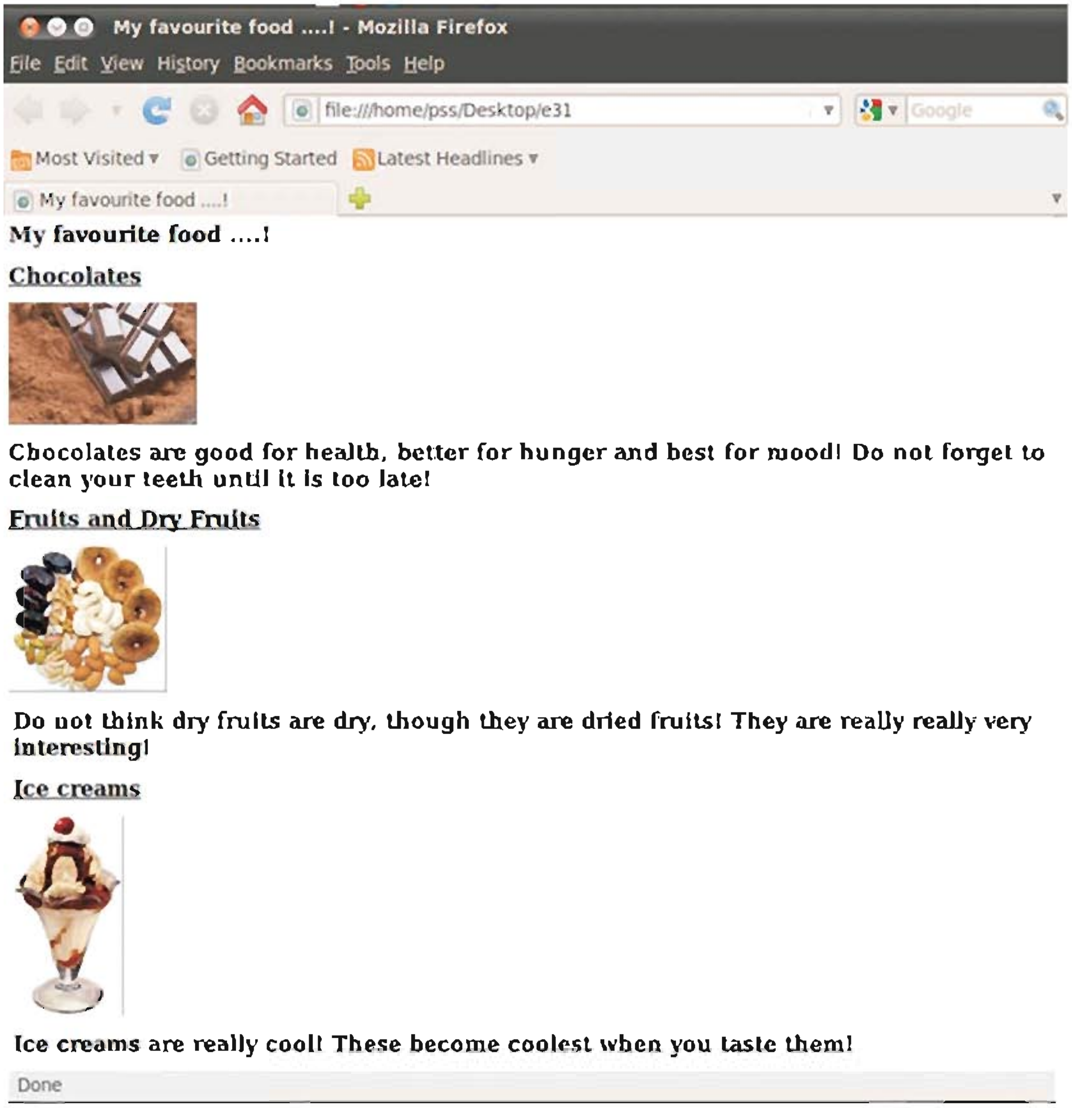
છબીઓ માટે એક અલાયદું ફોલ્ડર કે ડિરેક્ટરી બનાવવી એ એક સારી વ્યવસ્થા ગણાય. આ અભિગમ બહેતર છે, ખાસ કરીને જ્યારે વેબસાઈટ ઘણી મોટી હોય અને અનેક છબીઓનો ઉપયોગ કરતી હોય. HTMLમાં એક પછી એક ત્રણ છબીને ઉમેરવા માટેનો કોડ કોડ-લિસ્ટિંગ 3.1માં દર્શાવેલ છે.

```
<html>
<head>
      <title>My favourite food ….!</title>
</head>

<!-- ------------------------------------------------------------------------------------------ -->
<body>
     <h1> My favourite food ….!</h1>

<!-- ------------------------------------------------------------------------------------------ -->
     <h1> <u> Chocolates </u></h1>

<p>
<img src="chocolate.png">
<h1>
Chocolates are good for health, better for hunger and best for mood !
Do not forget to clean your teeth until it is too late !
</h1>
</p>

<!-- ------------------------------------------------------------------------------------------ -->
<h1><u> Fruits and Dry Fruits </u></h1>
<p>
<img src="fruits.jpg">
<h1>
      Do not think dry fruits are dry, though they are dried fruits !
      They are really really very interesting !
</h1>
</p>

<!-- ------------------------------------------------------------------------------------------ -->
<h1> <u> Ice creams </u> </h1>

<p>

<img src="icreams.jpg">

<h1>

      Ice creams are really cool ! These become coolest when you taste them !

</h1>

</p>


<!-- ------------------------------------------------------------------------------------------ -->

</body>

</html>
```

કોડ-લિસ્ટિંગ 3.1 : વેબપેઇજ પર અનેક છબીઓ ઉમેરવી

કોડ-લિસ્ટિંગ 3.1 કેટલીક ટપકાંવાળી લીટીઓ ધરાવે છે. આ ટપકાંવાળી લીટીઓ વેબપેઇજમાં દર્શાવાશે નહીં, કારણ કે, તેને કૉમેન્ટ ટૅગ (<!-- અને -->) માં આવરીને દર્શાવાઈ છે. આવી લીટીઓ HTML કોડને વાંચવામાં સરળતા બક્ષવા અને વધુ સારી રીતે સમજવા સ્વતંત્ર વિભાગોમાં વિભાજિત કરે છે. બ્રાઉઝર માત્ર માન્ય HTML વિધાનો જ સ્વીકારતું હોવાને કારણે આપણે ટપકાંવાળી લીટીને સીધેસીધી ટાઇપ કરી શકતાં નથી. આપણે આવી ટિપ્પણી જેવી લીટીઓને કૉમેન્ટ ટૅગમાં આવરીને દર્શાવવી જોઈએ.

કોડ-લિસ્ટિંગ 3.1માં આપેલ HTML કોડનું પરિણામ આકૃતિ 3.1માં દર્શાવાયું છે.



આકૃતિ 3.1 : વેબપેઇજમાં અનેક છબીઓ

બ્રાઉઝર આપેલા સ્થાન (source) પરથી એક પછી એક છબીઓ મેળવશે અને HTML કોડમાં જે ક્રમ દર્શાવ્યો હશે તે પ્રમાણે તેને ક્રમમાં દર્શાવશે. ઉપયોગકર્તા સામાન્ય રીતે તમામ છબીઓ સહિત પાનાને નિહાળે છે.

alt ગુણધર્મ

image ટૅગ સાથે alt ગુણધર્મ મૂળ છબીને વર્ણવે છે. આ ગુણધર્મને મૂકવો એ હંમેશાં સલાહભર્યું છે, કારણ કે તે છબીને વર્ણવે છે. alt ગુણધર્મનું ઉદાહરણ નીચે મુજબ છે:

```
<img src="rainbow.bmp" alt="This is a rainbow">
```

આ ગુણધર્મ અત્યંત ઉપયોગી છે. તેનું પહેલું કારણ એ છે કે, તે છબીને વર્ણવે છે, જે ઘણી બધી વખત જરૂરી હોય છે. અન્ય કારણ એ છે કે, ધારો કે, કોઈ પણ કારણોસર બ્રાઉઝર છબીને ન દર્શાવે તો પણ વાચક એટલું તો જાણતો જ હશે કે કયું ચિત્ર આવશે. ત્રીજું કારણ એ છે કે, જો વેબપેઇજ 'સ્ક્રીન રીડર' (Screen Reader)

જેવા સૉફ્ટવેર દ્વારા વાંચવામાં આવતું હશે તો છબીનું અર્થઘટન સરળ બની જશે. (સ્ક્રીન રીડર એ દૃષ્ટિની ખામી ધરાવતા લોકો માટે તૈયાર કરાયેલું એક વિશિષ્ટ સૉફ્ટવેર છે)

height અને width ગુણધર્મ

height અને width ગુણધર્મ નીચે દર્શાવ્યા પ્રમાણે છબીની ઊંચાઈ અને પહોળાઈ કેટલા પિક્સેલ છે તે દર્શાવે છે.

```
<img src="rainbow.bmp" alt="This is a rainbow" height="120" width="180">
```

અહીં, width ગુણધર્મ બ્રાઉઝરને જણાવે છે કે, વેબપેઇજ પર છબી કેટલી પહોળી દેખાવી જોઈએ. height ગુણધર્મ એ દર્શાવે છે કે છબી કેટલી ઊંચી હોવી જોઈએ. height અને width ગુણધર્મની કિંમત પિક્સેલની સંખ્યાને આધારે દર્શાવવામાં આવે છે. height અને width બન્ને એકસાથે ઉપયોગમાં લેવા જરૂરી નથી, તમે બેમાંથી કોઈ પણ એકનો ઉપયોગ કરી શકો. જો કે, કોઈ એક જ ગુણધર્મનો ઉપયોગ કરવાનું કંઈ ખાસ ઉપયોગી પણ નથી.

બ્રાઉઝરને છબીની લંબાઈ અને પહોળાઈ વિશે વધારાની માહિતી પૂરી પાડવાથી બ્રાઉઝરમાં છબીને લઈ આવવાનું (Load કરવાનું) અને રજૂઆત કરવાનું કાર્ય સરળ અને કાર્યક્ષમ બનશે. અહીં, બ્રાઉઝર જાણે છે કે, છબી માટે કેટલી જગ્યા ફાળવવી પડશે. અત્રે એ નોંધવું જોઈએ, કે તમારી છબી તમારા સ્ક્રીન કરતાં મોટી ન હોવી જોઈએ. જો કે, HTML કોડમાં મોટી છબીને અનુકૂળ રીતે સમાવવાનું શક્ય છે. વધુ ઊંચી કે નીચી ઊંચાઈ કે પહોળાઈની કિંમત આપીને તમારી છબીને વિસ્તૃત કરવી કે સંકોચવી એ પણ સલાહભર્યું નથી. આમ કરવાથી છબીની ગુણવત્તા અને દૃશ્યમાનતા નાશ પામશે.

align ગુણધર્મ

align ગુણધર્મ આપેલી છબીને પાનાની અંદર જ અથવા છબી ધરાવતા કોઈ પણ એલિમેન્ટમાં ગોઠવે છે. આ માટેનું ઉદાહરણ નીચે મુજબ છે :

```
<img src="chocolate.png" alt="This is a rainbow" align="right">
```

align ગુણધર્મની સાથે કોષ્ટક 3.1માં દર્શાવ્યા મુજબની કોઈ એક કિંમત આપી શકાય છે.

કિંમત	ઉદ્દેશ્ય
Top	છબી લખાણની હાલની લીટીના ઉપરના ભાગે ગોઠવાય છે.
Middle	છબી એ રીતે ગોઠવાય છે કે, છબીનો મધ્યભાગ હાલના લખાણ પાસે દેખાય છે.
Bottom	છબીનો છેક નીચેનો ભાગ લખાણની હાલની લીટી સાથે ગોઠવાય છે.
Left	છબી વિન્ડોની ડાબી બાજુ ગોઠવાય છે.
Right	છબી વિન્ડોની જમણી બાજુ ગોઠવાય છે.

કોષ્ટક 3.1 : align ગુણધર્મની શક્ય કિંમતો

border ગુણધર્મ

વેબપેઇજમાં છબીની ફરતે કિનારી દોરીને છબીને વધુ આકર્ષક બનાવી શકાય છે. સામાન્ય રીતે છબીઓને ચારબાજુ ફરતે કિનારી હોતી નથી.

છબીની ફરતે કિનારી બનાવવા માટે નીચે મુજબનો border ગુણધર્મ ઉપયોગમાં લઈ શકાય. border ગુણધર્મમાં borderની પહોળાઈ પિક્સેલમાં દર્શાવાય છે.

```
<img src="chocolate.png" alt="This is a rainbow" align="right" border ="20">
```

જ્યારે છબીને જોડાણ અપાયું હોય ત્યારે કેટલાંક બ્રાઉઝર (જેવાં કે, Internet Explorer) છબીને વધુ પ્રકાશિત (highlight) કરે છે. એટલે કે, જ્યારે છબી જોડાણ તરીકે ઉપયોગમાં લેવાઈ હશે ત્યારે તે કિનારી દ્વારા વધુ પ્રકાશિત કરાશે.

id ગુણધર્મ

id ગુણધર્મની મદદથી, તમે છબી માટે કોઈ ઓળખ (identifier) એટલે કે નામ આપી શકો. એ પછી જાવા (java) જેવી પ્રોગ્રામિંગ ભાષામાં લખાતી સ્ક્રિપ્ટમાં તે છબીને તે નામથી સંબોધી શકાય. id ગુણધર્મનો ઉપયોગ કેવી રીતે કરાય તે નીચેની ટૅગ દ્વારા દર્શાવાયું છે :

```
<img src="chocolate.png" id="choc_image">
```

ઈમેજની ચારેબાજુ ખાલી જગ્યા ઉમેરવી (Adding space around image)

ઈમેજની ચારેબાજુ ખાલી જગ્યા ઉમેરવા માટે vspace અને hspace ગુણધર્મનો ઉપયોગ કરવામાં આવે છે. છબીની ઉપર અને નીચેના ભાગે ખાલી જગ્યા ઉમેરવા vspace ગુણધર્મનો ઉપયોગ કરવામાં આવે છે એ જ રીતે, છબીની ડાબી અને જમણી બાજુએ ખાલી જગ્યા દર્શાવવા આ બન્ને ગુણધર્મોનો ઉપયોગ નીચેના ઉદાહરણમાં દર્શાવાયો છે:

```
<img src="rainbow.bmp" hspace="30" vspace="10">
```

જ્યારે કોઈ છબી લખાણની બિલકુલ અડોઅડ હોય અને જરાય જગ્યા ન દેખાતી હોય ત્યારે આ ગુણધર્મ ખૂબ જ ઉપયોગી નીવડે છે. છબીની ચારેય બાજુ જગ્યા છોડવાથી છબી જોવામાં વધુ સ્પષ્ટ બને છે અને વેબપેઇજ વધુ વાંચવા યોગ્ય બને છે. કોડ-લિસ્ટિંગ 3.2 એક એવું ઉદાહરણ દર્શાવે છે કે જેમાં ચૉકલેટની છબીની ફરતે થોડી જગ્યા છોડવામાં આવી છે.

```
<html>
<head>
    <title>My favourite food ....!</title>
</head>
<!-- ------------------------------------------------------------------ -->
<body>
    <h1> My favourite food ....!</h1>
<!-- ------------------------------------------------------------------ -->
<h1> <u> Chocolates </u></h1>
<p>
    <img src="chocolate.png" alt="Here comes the sweet image"  hspace="100"
vspace="200">
<h1>
    Chocolates are good for health, better for hunger and best for mood !
    Do not forget to clean your teeth until it is too late !
</h1>
</p>
<!-- ------------------------------------------------------------------ -->
</body>
</html>
```

કોડ-લિસ્ટિંગ 3.2 : hspace અને vspace વર્ણવતો HTML કોડ

જ્યારે તમે કોડ-લિસ્ટિંગ 3.2માં દર્શાવેલ HTML કોડને બ્રાઉઝરમાં નિહાળશો તો તે આકૃતિ 3.2 જેવો દેખાશે. hspace અને vspace ગુણધર્મ દ્વારા દર્શાવવામાં આવતી આડી અને ઊભી જગ્યા જોઈ શકાય છે. HTML કોડમાં થોડી વધુ છબીઓ ઉમેરો અને પછી જુઓ કે તમારું બ્રાઉઝર તે છબીઓને કેવી રીતે રજૂ કરે છે.

આકૃતિ 3.2 : hspace અને vspaceનું નિદર્શન

હવે તમે એ જોયું કે, hspace અને vspace ગુણધર્મ છબીની બન્ને બાજુ જગ્યા ઉમેરે છે. એટલે કે, જ્યારે hspaceનો ઉપયોગ કરવામાં આવે ત્યારે તે છબીની ડાબી અને જમણી એમ બન્ને બાજુ જગ્યા છોડે છે. એ જ રીતે, જ્યારે vspace ગુણધર્મનો ઉપયોગ કરવામાં આવે છે ત્યારે તે છબીની ઉપરની અને નીચેની બાજુએ જગ્યા છોડે છે. માત્ર કોઈ એક તરફ જ છબી ઉમેરાય તેવું કરવા માટે આપણે છબીને યોગ્ય રીતે સુધારવી પડે. એના વિકલ્પરૂપે મૂળ છબીની બાજુમાં એક ખાલી છબી છાપવાનું પણ વિચારી શકાય. જો કે, આ બીજા વિકલ્પમાં જગ્યા અને ગોઠવણી કરવાનું થોડું અઘરું બની જાય.

તમારા કમ્પ્યૂટરની સ્મૃતિમાંથી "chocolate.png" ફાઈલને દૂર કરવાનો પ્રયાસ કરો અને જુઓ કે શું થાય છે ? શું તે છબી માટે વૈકલ્પિક વર્ણન જેમ કે, "Here comes the sweet image" એવું દર્શાવશે ? તમને કદાચ આકૃતિ 3.3માં દર્શાવ્યા મુજબનું પરિણામ જોવા મળે.

આકૃતિ 3.3 : છબીની અનુપસ્થિતિમાં વૈકલ્પિક વર્ણન

ઓછી રિઝૉલ્યુશનવાળી છબી

વધુ રિઝૉલ્યુશનવાળી છબી ઘણી વધુ જગ્યા રોકે છે અને વેબપેઇજમાં રજૂ થવામાં ઘણી વાર લગાડે છે. વધુ રિઝૉલ્યુશનવાળી છબી વેબપેઇજમાં ધીમે ધીમે રજૂ થાય ત્યાં સુધી આપણે કામચલાઉ ઓછા રિઝૉલ્યુશનવાળી છબીને નીચે દર્શાવ્યા મુજબ વેબપેઇજમાં રજૂ કરી શકીએ :

```
<img src="rainbow.bmp" lowsrc="rainbow_low.bmp">
```

કોઈ પણ છબીને ઓછા રિઝૉલ્યુશનવાળી છબી બનાવવા માટે તમે છબીમાં સુધારા-વધારા કરવાની સવલત આપતા 'ઇમેજ એડિટિંગ ટૂલ'ની resize, crop વગેરે ટેકનિક્નો ઉપયોગ કરી શકો.

ઓછા રિઝૉલ્યુશનવાળી છબી મૂકવાના વિકલ્પના ગુણધર્મના ઉપયોગ દ્વારા આપણે બ્રાઉઝરની ગતિની સમસ્યા ઉકેલી શકીએ તેમ છતાં, આપણે વૈકલ્પિક છબી બનાવવી પડે અને તેને સાચવવી પડે. વળી, ઓછા રિઝૉલ્યુશનવાળી છબી ધૂંધળી અને અસ્પષ્ટ દેખાશે.

વિવિધ સ્વરૂપોની છબીઓને સમાવવી

અત્યાર સુધી આપણે Image ટૅગમાં bmp (Bit Mapped Picture) ફાઈલનો જ સમાવેશ કર્યો. પરંતુ, છબીના થોડાં અન્ય સ્વરૂપો પણ ઉપલબ્ધ છે, જેનો Image ટૅગમાં સમાવેશ કરી શકાય. છબી માટેનાં વિવિધ સ્વરૂપો પૈકી bmp, jpeg, png, tiff અને gif એ વધુ પ્રચલિત છબી સ્વરૂપો છે. આવાં છબીસ્વરૂપોનું વર્ણન કોષ્ટક 3.2માં દર્શાવેલ છે.

છબી ફાઈલ સ્વરૂપ	વર્ણન
BMP	Windows અને OS/2 માટેની Bitmap Graphic file
GIF	Graphic Interchange Format file
JPEG	Joint Photographic Expert Group file
JPG	JPEG/JIFF Image file
PNG	Portable Network Graphics file
TIFF	Tag Image File Format

કોષ્ટક 3.2 : છબી સ્વરૂપો

ડિજટલ કૅમેરા અને વેબપેઇજં સામાન્ય રીતે jpg/jpeg ફાઈલોનો ઉપયોગ કરે છે. કારણ કે, આ સ્વરૂપો ફાઈલમાં ડેટાનું નોંધપાત્ર રીતે સંકુચન કરે છે. ફોટોગ્રાફ જેવી (continuously toned) છબી માટે jpg અને jpeg છબી સ્વરૂપ આદર્શ છે. વ્યાપારી ધોરણે, tiff છબીસ્વરૂપ વધુ ઉપયોગમાં લેવાય છે, કારણ કે તે ઉચ્ચ ગુણવત્તા અને લઘુતમ નુકસાન સાથે છબીનું સારા એવા પ્રમાણમાં સંકુચન કરી આપે છે. Tiff એ ખરેખર નુકસાનરહિત સારા છબી સ્વરૂપોમાં એક છબીસ્વરૂપ છે. gif અને png જેવા અન્ય ફાઈલ સ્વરૂપો પણ નુકસાનરહિત સંકુચન ટેક્નોલૉજીનો ઉપયોગ કરે છે. જો કે, png સ્વરૂપ લખવા અને વાંચવા માટે પ્રમાણમાં ધીમું છે.

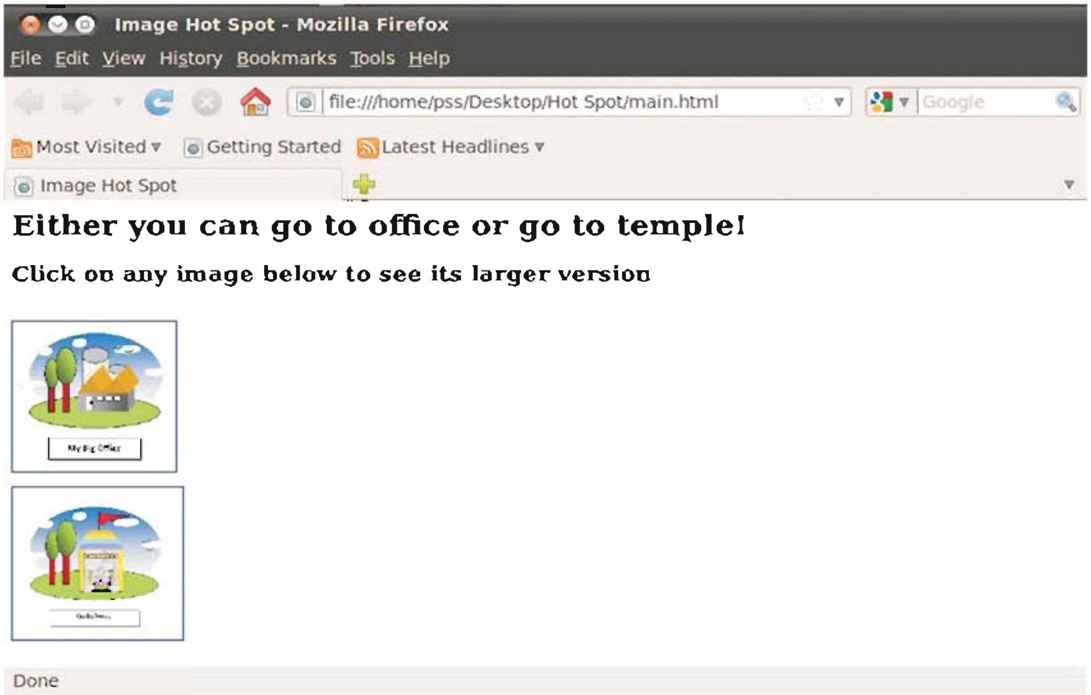
આકર્ષણ કેન્દ્ર હૉટ સ્પૉટ તરીકે છબી

ઘણી વખત પ્રમાણમાં મોટી છબીને દર્શાવવામાં બ્રાઉઝર થોડો સમય લગાડે છે. આ ઉપરાંત જ્યારે આપણી પાસે એક કે વધુ પાનાં પર અનેક છબીઓ હોય ત્યારે ઉપયોગકર્તા માટે મૂંઝવણરૂપ બને છે કે કઈ છબીને પ્રથમ ગણવી. શું મોબાઇલ ફોન, ઘરેણાં, પુસ્તકો અને એના જેવી ચીજવસ્તુઓ માટેની ઑનલાઇન ખરીદી માટેની વેબસાઇટ પર તમારું ધ્યાન પડ્યું છે ? એક જ વેબપેઇજ પર તમને કદાચ ઘણી ચીજવસ્તુઓની ટૂંકા વર્ણન સાથેની અનેક નાની છબીઓ જોવા મળશે. જ્યારે તમને એમાંથી કોઈ વસ્તુમાં રસ પડે તો તમે તે નાની છબી પર ક્લિક કરી શકો, જેથી તમને પસંદ પડેલી ચીજવસ્તુ માટે સારી ગુણવત્તાવાળી પ્રમાણમાં મોટી છબીની સાથે સાથે વસ્તુ વિશેની વિસ્તૃત માહિતી ધરાવતું વેબપેઇજું દર્શાવાય. આવું કરવા માટે વેબપેઇજ પરની દરેક નાની છબીને જોડાણ (લિન્ક) આપવી પડે. તમે એવું કહી શકો કે, અહીં છબીએ હૉટ ટેક્સ્ટ (hot text) જેવું કાર્ય કરે છે. જોકે, અહીં કોઈ ટેક્સ્ટ તો છે નહીં ! માટે ખરેખર આને હૉટ ટેક્સ્ટ તો કહી ન શકાય ! ખરુંને ! તેથી હૉટ ટેક્સ્ટ જેવું કામ કરતી આ છબીવાળી જગ્યા(spot)ને હૉટ સ્પૉટ (hot spot) કહે છે. આવી બે નાની છબી ધરાવતી અને બે જુદા જુદા વેબપેઇજ તરફ દોરી જતા HTML કોડ, કોડ-લિસ્ટિંગ 3.3માં રજૂ કરવામાં આવેલ છે.

```
<html>
        <head>
                <title> Image Hot Spot  </title>
        </head>
<!-- ---------------------------------------------------------------------------- -->
        <body>
                <h1> Either you can go to office or go to temple!</h1>

                <p> <h2>Click on any image below to see its larger version </h2> </p>

                <br>
<!-- ----------- adding links to the images --------------------------------------- -->

        <a href="big_office.html"><img src="small_office.bmp"></img></a></p>

        <a href="big_temple.html"><img src="small_temple.bmp"></img></a></p>
<!-- ---------------------------------------------------------------------------- -->
        </body>
</html>
```

કોડ-લિસ્ટિંગ 3.3 : હૉટસ્પૉટ તરીકે છબી માટેના HTML કોડ

main.html ફાઈલમાં ઉપરોક્ત કોડ ટાઈપ કરી સંગ્રહ કરો. જો તમે આ કોડને બ્રાઉઝર દ્વારા નિહાળશો તો તે આબેહૂબ આકૃતિ 3.4માં દર્શાવ્યા મુજબ દેખાશે.



આકૃતિ 3.4 : હૉટ સ્પૉટ તરીકે છબીનો ઉપયોગ

કોડ-લિસ્ટિંગ 3.3માં દર્શાવેલ HTML કોડ ઉપરાંત પણ આપણે ચાર છબીની જરૂર પડશે. આપણને small_office.bmp નામની ઑફિસની એક નાની છબીની જરૂર પડશે અને big_office.bmp નામની ઑફિસની એક મોટી છબી ફાઈલની પણ જરૂર પડશે. એ જ રીતે, આપણને small_temple.bmp નામની મંદિરની એક નાની છબી અને big_temple.bmp નામની મંદિરની એક મોટી છબીની પણ જરૂર પડશે. આ માટે તમારે નવી છબીઓ તૈયાર કરવી પડે અથવા હયાત છબીઓનાં નામ બદલીને પણ ઉપયોગમાં લઈ શકો.

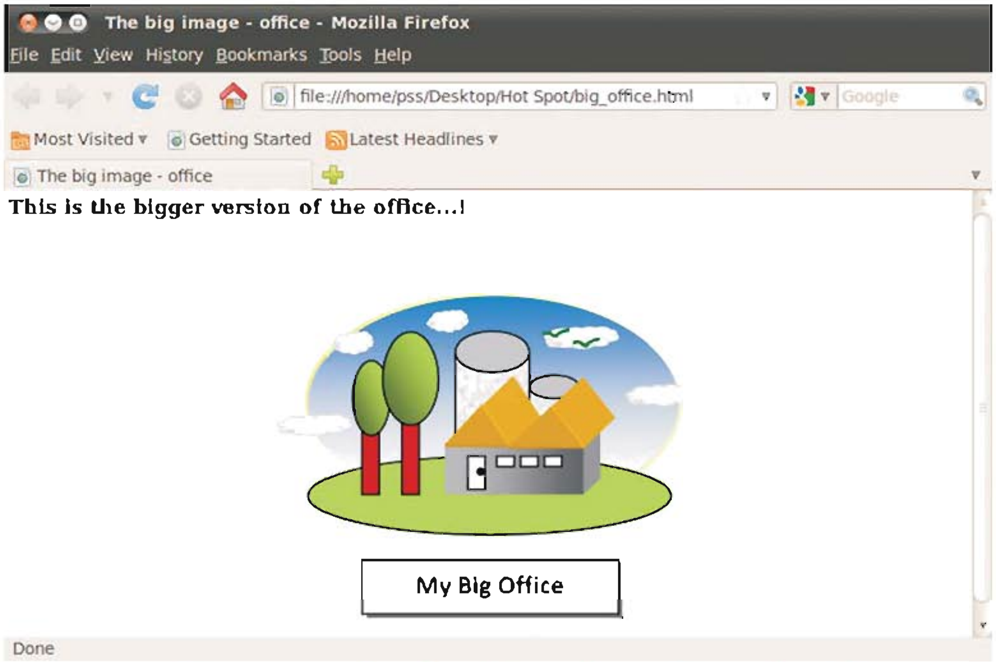
જો ઑફિસ માટે (small_office.bmp) અને મંદિર માટે (small_temple.bmp) ફાઇલ ઉપલબ્ધ હશે અને જો તમે main.htmlને બ્રાઉઝર દ્વારા નિહાળશો, તો આકૃતિ 3.4માં દર્શાવ્યા મુજબનું પરિણામ તમે નિહાળી શકશો.

એકવાર બ્રાઉઝરમાં તમે main.html પાનામાં બન્ને છબીઓને યોગ્ય રીતે નિહાળી લો, પછી છબીઓ પર ક્લિક કરવાનો પ્રયત્ન કરી શકો. જ્યારે તમે પ્રથમ છબી પર ક્લિક કરશો ત્યારે નાની છબીને હૉટ સ્પૉટ ગણીને તેની નીચે આપેલ href ટૅગ તમને બીજા પાના (big_office.html) તરફ લઈ જશે.

```
<a href="big_office.html"><img src="small_office.bmp"></img></a></p>
```

અત્રે એ નોંધ લો કે, ઑફિસની મોટી છબી સાથે તમારી big_office.html તૈયાર હોવી જ જોઈએ.

પ્રથમ છબી પર ક્લિક કરશો ત્યારે તમને આકૃતિ 3.5માં દેખાય છે તેવી સ્ક્રીન જોવા મળશે.



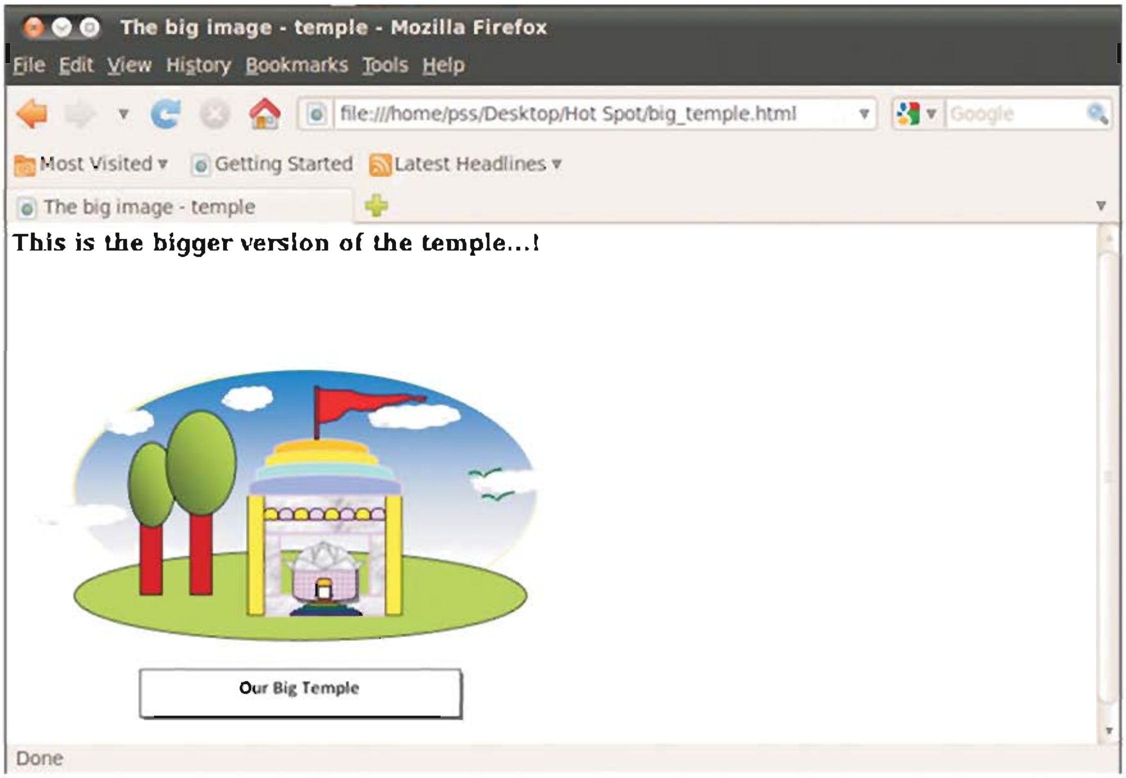
આકૃતિ 3.5 : જ્યારે તમે પહેલી છબી પર ક્લિક કરો ત્યારે

big_office.html તૈયાર કરવા માટેના HTML કોડ કોડ-લિસ્ટિંગ 3.4માં આપેલ છે.

```
<html>
    <head>
        <title> The big image - office </title>
    </head>
<!-- -------------------------------------------------------------- -->
    <body>
        <h1> This is the bigger version of the office...! </h1>
        <br>
        <br>
        <br>

        <img src="big_office.bmp"></img>
<!-- -------------------------------------------------------------- -->
</body>
</html>
```

કોડ-લિસ્ટિંગ 3.4 : પ્રથમ છબી માટેના કોડ

એ જ રીતે, જ્યારે આકૃતિ 3.4માં દર્શાવેલી બીજી છબી પર તમે ક્લિક કરશો ત્યારે તમે આકૃતિ 3.6માં દર્શાવ્યા મુજબનું પરિણામ નિહાળી શકશો :



આકૃતિ 3.6 : જ્યારે તમે બીજી છબી પર ક્લિક કરો ત્યારે

આકૃતિ 3.6 માટેનો HTML કોડ (big_temple.html) કોડ-લિસ્ટિંગ 3.5માં વર્ણવેલ છે.

```
<html>
    <head>
        <title> The big image - temple </title>

    </head>
<!-- --------------------------------------------------- -->
<body>
    <h1> This is the bigger version of the temple...! </h1>
    <br>
    <br>
    <br>
    <img src="big_temple.bmp"></img>
<!-- --------------------------------------------------- -->
    </body>
</html>
```

કોડ-લિસ્ટિંગ 3.5 : બીજી છબી માટેના કોડ

ઉપરોક્ત ઉદાહરણને પૂર્ણ કરવા તમારે નીચે મુજબ ચાર છબીઓ અને ત્રણ HTML ફાઈલની જરૂર પડી :

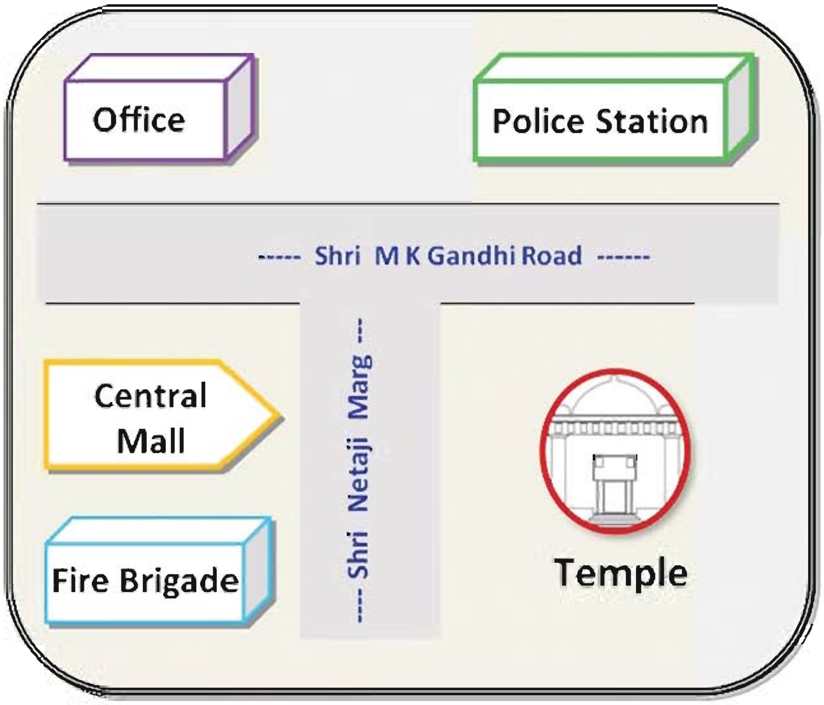
1.	Small_office.bmp	→	bmp સ્વરૂપમાં ઑફિસની એક નાની છબી (અથવા અન્ય કોઈ પણ માન્ય છબી સ્વરૂપ)
2.	Small_temple.bmp	→	bmp સ્વરૂપમાં મંદિરની એક નાની છબી
3.	Big_office.bmp	→	bmp સ્વરૂપમાં ઑફિસની એક મોટી છબી
4.	Big_temple.bmp	→	bmp સ્વરૂપમાં મંદિરની એક મોટી છબી
5.	Main.html	→	બે નાની છબી સાથેની HTML કોડની ફાઈલ, અન્ય બે HTML ફાઈલ સાથે અનુસંધાન અને થોડું લખાણ
6.	Big_office.html	→	ઑફિસની એક મોટી છબી સાથેની HTML કોડની ફાઇલ
7.	Big_temple.html	→	મંદિરની એક મોટી છબી સાથેની HTML કોડની ફાઇલ

મુખ્ય પાનામાં તમે કેટલીક વધુ નાની છબીઓ ઉમેરવાનો પ્રયત્ન કરી શકો. નાની છબીઓને થમ્બનેઇલ (thumbnail) તરીકે પણ ઓળખવામાં આવે છે. જ્યારે તમે વેબપેઇજમાં પદ્ધતિસર ગોઠવાયેલી આવી નાની છબીઓ પર ક્લિક કરશો ત્યારે દરેક છબી તેની વિસ્તૃત (અથવા પ્રમાણમાં મોટી) આવૃત્તિ તરફ દોરી જશે.

Image Map

આગળના વિભાગમાં, આપણે છબી સાથે એક જોડાણ (લિન્ક) કર્યું. એક જ છબી સાથે અનેક જોડાણ કરવાનું પણ શક્ય છે. અહીં દરેક જોડાણ જુદા જુદા સંબંધ તરફ નિર્દેશ કરે છે. એક જ છબીમાં એકથી વધુ જોડાણ બનાવવા આપણે છબીની અંદર એક કરતાં વધુ ક્લિક કરી શકાય તેવાં સ્થાન તૈયાર કરવાં પડશે. આવા દરેક સ્થાનને 'હૉટ સ્પૉટ' (Hot spot) કહેવાય. એવું ધારો કે, તમારી પાસે ધંધાકીય કચેરી (કૉર્પોરેટ ઑફિસ), ખરીદ કેન્દ્ર (શૉપિંગ મૉલ), મંદિર, પોલિસમથક અને અગ્નિશામક દળમથક જેવાં વિવિધ સ્થાનો દર્શાવતો કોઈ વિસ્તારનો નકશો છે. આ દરેક સ્થાન ઉપર આપણે જોડાણ (લિન્ક) બનાવી શકીએ. ઉદાહરણ તરીકે, મંદિરના સ્થાન ઉપર આપણે એવું જોડાણ (લિન્ક) ગોઠવી શકીએ જે આપણને મંદિર વિશેની માહિતી ધરાવતા પાના તરફ લઈ જાય. આ રીતે જોડાણ કરાયેલ વેબપેઇજ કેટલાક ફોટા, સમાચારો, મંદિર વિશેનો ટૂંકો ઇતિહાસ અને મંદિરે કેવી રીતે પહોંચી શકાય તે માટેનું દિશાદર્શન પણ દર્શાવી શકે. જો તમારી પાસે કદાચ કોઈ નકશો (ધારો કે, ભારતનો નકશો) હોય તો નકશાના કેટલાક પ્રદેશો(ગુજરાત, મહારાષ્ટ્ર વગેરે જેવા રાજ્યો)ને 'હૉટ સ્પૉટ' બનાવી શકાય. જ્યારે કોઈ વ્યક્તિ આવા કોઈ હૉટ સ્પૉટ પર ક્લિક કરે ત્યારે નવું વેબપેઇજ ખોલી શકાય.

આકૃતિ 3.7માં દર્શાવેલી છબીને ધ્યાને લો. છબીમાં એક મંદિર, એક ધંધાકીય કચેરી, એક ખરીદ કેન્દ્ર, એક અગ્નિશામકદળ મથક અને એક પોલિસમથક છે. આ ઉપયોગી કેન્દ્રો "શ્રી એમ. કે. ગાંધી માર્ગ" અને "શ્રી નેતાજી માર્ગ" નામના બે મુખ્ય રસ્તા પર આવેલ છે. આ પાંચેય મકાનો અને બન્ને રસ્તાઓ પર જોડાણ ગોઠવેલ છે. એટલે કે, સાત જુદા જુદા વિસ્તાર પર સાત જોડાણ ગોઠવ્યાં છે. અગાઉ જણાવ્યા અનુસાર જે જગ્યા પર જોડાણ ગોઠવવામાં આવે તે જગ્યાને 'હૉટ સ્પૉટ' કહે છે. હૉટ સ્પૉટ મોટા અને સ્પષ્ટ દેખાય તેવા હોવા જોઈએ, જેથી ઉપયોગકર્તા આસાનીથી તેમને ઓળખી શકે અને તેના પર ક્લિક કરી શકે. અન્યથા હૉટ સ્પૉટ પસંદ કરવામાં અને જોડાણને અનુસરવામાં ઉપયોગકર્તાને તકલીફ પડશે. વધુમાં, છબી એવી માહિતી રજૂ કરતી હોવી જોઈએ કે, આવી જગ્યા / હૉટ સ્પૉટ ઉપર ક્લિક કરવાથી ઉપયોગકર્તાને પસંદ કરેલ જગ્યા વિશે વિસ્તૃત માહિતી દર્શાવતા નવા વેબપેઇજ તરફ લઈ જવામાં આવશે. ઉદાહરણ તરીકે જ્યારે આકૃતિ 3.7માં દર્શાવેલી છબી બ્રાઉઝરમાં દર્શાવાય ત્યારે ઉપયોગકર્તાને "click on temple to know more..." એવો સંદેશ બતાવી શકાય. છબી બનાવી શકતા યોગ્ય "ઇમેજ ક્રિએશન ટૂલ"નો ઉપયોગ કરી આકૃતિ 3.7માં દર્શાવ્યા મુજબની છબી તૈયાર કરો અને તેને city.bmp એવું નામ આપો.



આકૃતિ 3.7 : શહેરના બજારનું દૃશ્ય

હૉટ સ્પૉટના સ્થાન છબીના ઉપરની બાજુના ડાબા ખૂણેથી x અને y પ્રતિષ્ઠાપકો (co-ordinates) માપીને દર્શાવવામાં આવે છે. આ પ્રતિષ્ઠાપકો બે રીતે ઉપયોગમાં લેવામાં આવે છે :

- હૉટ સ્પૉટ ક્યાં આવેલા છે તે દર્શાવવા
- ક્લિક મારફતે ઉપયોગકર્તા દ્વારા પૂરાં પાડવામાં આવેલ પ્રતિષ્ઠાપકોને સરખાવવા.

ઉપયોગકર્તા બાજુ, બ્રાઉઝર ઉપયોગકર્તાની ક્લિકના પ્રતિષ્ઠાપકોને ઓળખી કાઢે છે અને ઉપયોગકર્તાને દર્શાવેલ વેબપેઈજ઼ પર લઈ જાય છે. આ માટે Image ટૅગમાં usemap ગુણધર્મ સાથે <map> અને <area> ટૅગ ઉપયોગમાં લેવાય છે. map ટૅગને name નામનો એક ગુણધર્મ હોય છે. name ગુણધર્મની કિંમત usemap ગુણધર્મની કિંમત સાથે મળવી જોઈએ. નીચેનું ઉદાહરણ જુઓ:

```
<img src="city.bmp" width="700" height="600" alt="city" usemap="#roadmap">
```

અહીં "#roadmap" એ આપણે બનાવેલા usemapના આઈડેન્ટિફાયરનું નામ છે. <map> અને </map> ટૅગની વચ્ચે હોટ સ્પોટ પ્રતિષ્ઠાપકો અને તેમનું સંબંધિત જોડાણ (લિન્ક) દર્શાવવાની હોય છે. અહીં મંદિરના પ્રતિષ્ઠાપકો (518, 378, 70) છે અને તે વર્તુળાકાર છે. આપણે આ માહિતીને map ટૅગમાં નીચે મુજબ આવરી લઈ શકીએ.

```
<map name="roadmap">
<area shape="circle" coords="518,378,70" alt="Temple" href="Temple.html">
<!-- —About co-ordinates of other hot spots --------- -->
</map>
```

ઉપરોક્ત HTML વિભાગની બીજી લીટીમાં આપેલ કો-ઑર્ડિનેટ્સ આકૃતિ 3.7માં દર્શાવેલ city.bmp ફાઈલનું મંદિર રજૂ કરે છે. મંદિર વર્તુળાકાર દર્શાવાયું હોવાને લીધે shapeની કિંમત આપણે "circle" આપી છે. ધારો કે, મંદિર લંબચોરસ આકારમાં દર્શાવવાનું હોત તો shape ની કિંમત તરીકે "rect" લખવું પડે. area ટૅગ દ્વારા સ્થાનનો આકાર

દર્શાવાય તે જરૂરી છે. circle, rectangle અને polygon એ માન્ય આકારો છે. લંબચોરસ rect દ્વારા; વર્તુળ circle દ્વારા અને બહુકોણ poly દ્વારા દર્શાવાય છે. એના બદલે પૂરું નામ જેમ કે, rectangle પણ ઉપયોગમાં લેવાય છે. જુદા જુદા આકારો સાથે area ટૅગને વર્ણવતું નીચેનું ઉદાહરણ જુઓ.

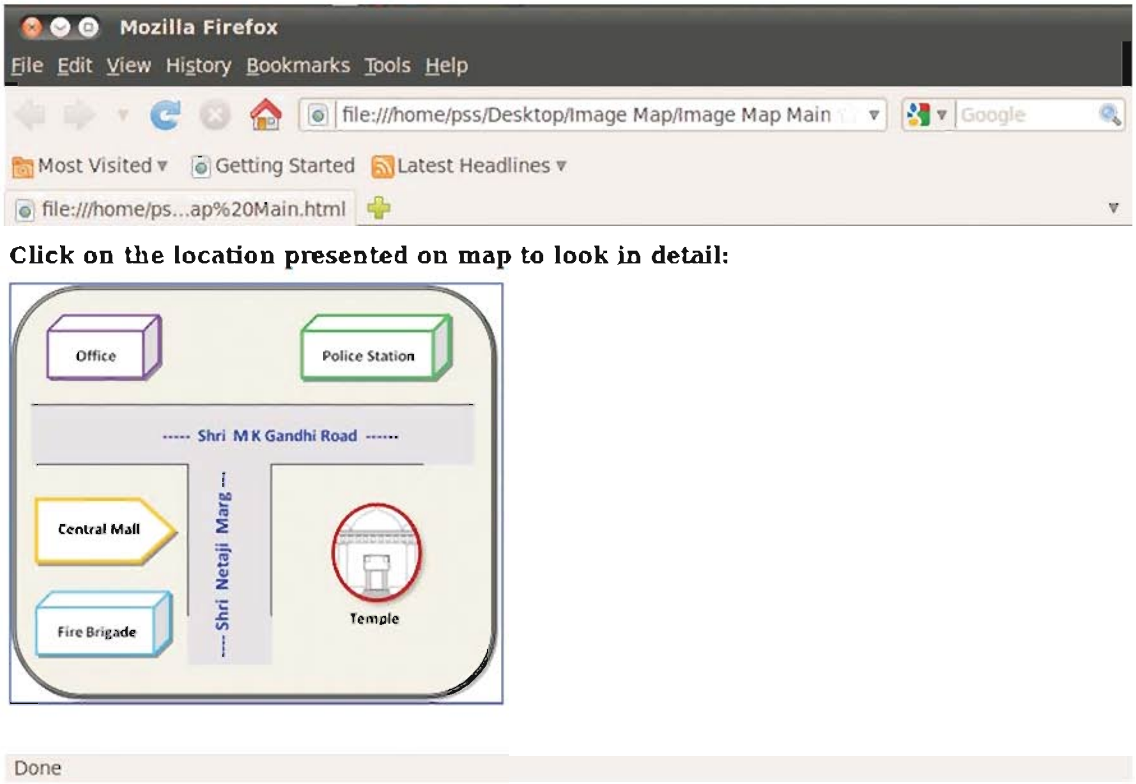
- **<area shape="poly" coords="32,301,183,301,239,352,188,399,32,399" alt="Central Mall" href="CentralMall.html">**
- **<area shape="rect" coords="32,432,233,532" alt="Fire Brigade" href="FireBrigade.html">**
- **<area shape="circle" coords="518,378,70" alt="Temple" href="Temple.html">**

આકૃતિ 3.7માં દર્શાવેલ શહેરના દશ્યનો ઉપયોગ કરીને ઇમેજ મૅપ (Image map) તૈયાર કરવા માટેનો સંપૂર્ણ HTML કોડ, કોડ-લિસ્ટિંગ 3.6માં દર્શાવાયો છે.

```
<html>
<body>
<p>Click on the location presented on map to look in detail:</p>
<!-- --------------------------------------------------------------- -->
<img src="city.bmp" width="696" height="593" alt="Map" usemap="#roadmap">

<!-- ------------------------------------------------------------------------------- -->
<map name="roadmap">

<area shape="rect" coords="46,37,219,141" alt="Office" href="Office.html">

<area shape="rect" coords="407,38,632,142" alt="Police Station" href= "PoliceStation.html">

<area shape="poly" coords="32,301,183,301,239,352,188,399,32,399" alt="Central Mall"
href="CentralMall.html">

<area shape="rect" coords="32,432,233,532" alt="Fire Brigade" href="FireBrigade.html">

<area shape="circle" coords="518,378,70" alt="Temple" href="Temple.html">

</map>
<!-- ------------------------------------------------------------------------------------- -->
</body>
</html>
```

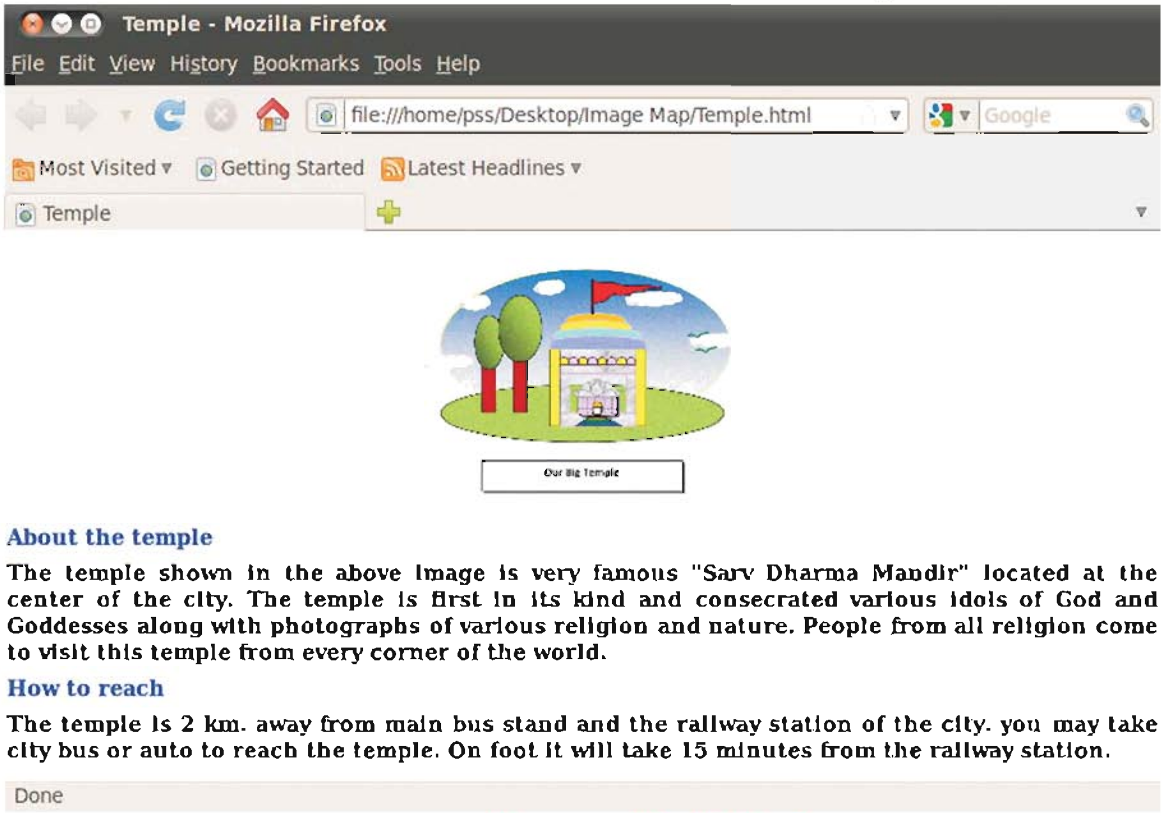
કોડ-લિસ્ટિંગ 3.6 : image map માટેના HTML કોડ

બ્રાઉઝરમાં કોડ લિસ્ટિંગ 3.6નું પરિણામ આકૃતિ 3.8માં દર્શાવ્યા મુજબનું દેખાશે :



આકૃતિ 3.8 : બ્રાઉઝરમાં ઈમેજ મેપ

જ્યારે તમે આકૃતિ 3.8માં દર્શાવેલ મંદિરની વર્તુળાકાર છબી પર ક્લિક કરશો ત્યારે તમને મંદિર વિશેની વિગતો દર્શાવતા નવા પાના પર લઈ જવામાં આવશે. મંદિરની વિગતો રજૂ કરતી અન્ય ફાઈલ દર્શાવતી આકૃતિ 3.9 જુઓ.



આકૃતિ 3.9 : મંદિરની વિગતો

આકૃતિ 3.9માં દર્શાવેલ વેબપેઇજ બનાવવા માટે જેની જરૂર પડે તે કોડ, કોડ-લિસ્ટિંગ 3.7 દર્શાવે છે :

```
<html>
<head> <title> Temple </title> </head>

<body>
      <center>
            <img src="Temple.gif"   align="middle"> </img>
      </center>

<!-- ------------------------------------------------------------------------------------ -->

<h1> <font color="blue">About the temple </font> </h1>

      <p align="justify"> <b>

            The temple shown in the above image is very famous "Sarv Dharma Mandir"

            located at the center of the city. The temple is first in its kind and

            consecrated various idols of God and Goddesses along with photographs

            of various religion and nature. People from all religion come to visit

            this temple from every corner of the world.

</b>
</p>
<!-- ------------------------------------------------------------------------------------ -->
<h1> <font color="blue">  How to reach </font> </h1>

      <p align =justify> <b>

            The temple is 2 km. away from main bus stand and the railway station of the city.

            you may take city bus or auto to reach the temple. On foot it will take 15 minutes

            from the railway station.
      </b>
      </p>
</body>
</html>
```

કોડ-લિસ્ટિંગ 3.7 : મંદિરની વિગતો માટેનો કોડ

એ જ રીતે, તમે કચેરી (Office.html), પોલિસમથક (PoliceStation.html), અગ્નિશામક દળ (FireBrigade.html) અને સેન્ટ્રલ મોલ (CentralMall.html) માટે HTML ફાઈલો તૈયાર કરી શકો. બધી જ ફાઈલો સ્વતંત્ર રીતે કામ કરે છે તે ચકાસી લો અને આખા શહેરનું સમગ્રતયા ચિત્ર દર્શાવતી મુખ્ય HTML ફાઈલમાંથી ક્લિક કરીને તેમનો પ્રયોગ કરી જુઓ.

મલ્ટિમીડિયા ફાઈલોને જોડવી

HTML દસ્તાવેજમાં ઍન્કર ટૅગનો ઉપયોગ કરીને નીચે દર્શાવ્યા પ્રમાણે વીડિયો તેમજ ઑડિયો ફાઇલને સંકલિત કરી શકાય છે:

** follow this recipe and enjoy delicious food...! **

આગળની લીટીમાં દર્શાવેલ ઉદાહરણમાં "food.mp4" નામની એક ચલચિત્ર (movie) ફાઈલ છે. તમારી પાસે પણ વીડિયો મુવી ફાઈલ તૈયાર હોવી જોઈએ. ઉપયોગકર્તા જ્યારે વેબપેઇજ પર દર્શાવાયેલ "Enjoy this... !" શબ્દો પર ક્લિક કરશે ત્યારે યોગ્ય સૉફ્ટવેરમાં વીડિયો ફાઈલ દર્શાવાશે. જો ઉલ્લેખ કરાયેલી મૂવી ફાઈલ અન્ય વેબસાઇટ કે અન્ય સ્થળે હશે તો સૌપ્રથમ તેને ઉપયોગકર્તાના કમ્પ્યૂટરની ડિરેક્ટરી પર કામચલાઉ ઇન્ટરનેટ ફાઈલ તરીકે સંગ્રહવામાં આવશે. મૂવી ફાઈલનો અમલ કરતી વખતે જો જરૂરી સૉફ્ટવેર નહીં મળે તો, બ્રાઉઝર તમને કમ્પ્યૂટર અથવા ઇન્ટરનેટ પરથી વૈકલ્પિક સૉફ્ટવેર પસંદ કરવા પૂછશે.

HTML કોડમાં ધ્વનિ ફાઈલને ઉમેરવા માટેનું ઉદાહરણ નીચે આપેલ છે :

** Enjoy this song...! **

.mov ફાઈલ સ્વરૂપ ઉપરાંત વીડિયો ફાઈલ માટે અન્ય ફાઈલ સ્વરૂપો પણ છે. એમાંનાં થોડાં સ્વરૂપોમાં .avi, .wmv, .mpg અથવા .mpeg, અને .swf ગણી શકાય. આમાંના કેટલાંકને ફાઈલનો અમલ કરવા માટે વીડિયો પ્લેયર ડાઉનલોડ કરવાની જરૂર પડે છે.

સારાંશ

આ પ્રકરણમાં આપણે HTML દસ્તાવેજમાં છબીને કેવી રીતે ઉમેરવી તે શીખ્યા. HTML દસ્તાવેજમાં એક કે વધુ છબી ઉમેરવા ઉપરાંત align, hspace અને vspace જેવા ગુણધર્મોનો ઉપયોગ કરીને છબીઓને સ્ક્રીન પર કેવી રીતે ગોઠવવી તે પણ આપણે શીખ્યા. છબીને હૉટ સ્પૉટ કેવી રીતે બનાવવી તે પણ આપણે જોયું. માત્ર એક આખી છબીને જ નહીં બલ્કે, છબીના વિવિધ ભાગો(જેમ કે ભૌગોલિક નકશો)ને image map લાક્ષણિકતાનો ઉપયોગ કરીને જુદા જુદા વેબપેઇજઓ સાથે જોડી શકાય તે પણ આપણે જાણ્યું. અંતમાં, HTML સાથે ધ્વનિ અને વીડિયો ફાઈલ કેવી રીતે ઉમેરવી તે વિશેની માહિતી પણ મેળવી. આ પ્રકરણમાં આપવામાં આવેલી માહિતીનો ઉપયોગ કરીને, વિદ્યાર્થી HTML દસ્તાવેજમાં માત્ર સાદું લખાણ અને આંકડાઓ જ રજૂ કરતાં શીખવાને બદલે તે વેબપેઇજમાં વીડિયો, એનિમેશન, છબીઓ અને વિવિધ ધ્વનિને સંમિલિત કરવા માટે પણ સક્ષમ બનશે.

સ્વાધ્યાય

1. HTML દસ્તાવેજમાં છબીઓને કેવી રીતે ઉમેરી શકાય તે યોગ્ય ઉદાહરણ આપીને સમજાવો.
2. HTMLના image ટૅગના વિવિધ ગુણધર્મો પર ટૂંક નોંધ લખો.
3. હોટસ્પોટની વ્યાખ્યા આપો. HTMLમાં છબીને કેવી રીતે હોટસ્પોટ બનાવી તેનો ઉપયોગ કરી શકાય તે ચર્ચો.
4. Image map પર ટૂંક નોંધ લખો.
5. નીચેના પ્રશ્નો માટે આપેલા વિકલ્પોમાંથી યોગ્ય વિકલ્પ પસંદ કરી જવાબ આપો:

   (1) HTML દસ્તાવેજમાં વીડિયો ફાઈલ ઉમેરવા નીચેના પૈકી ક્યો ટૅગ ઉપયોગમાં લેવાય છે ?

   (a) img (b) image (c) href (d) ime

(2) image એલિમેન્ટ લીટી તોડતા (line break કરતાં) નથી, આને નીચેનામાંથી કયા નામે ઓળખવામાં આવે છે ?

(a) ઑનલાઇન ઇમેજ (b) ઇનલાઇન ઇમેજ

(c) આઉટલાઇન ઇમેજ (d) બ્લૅન્ક ઇમેજ

(3) નીચેનામાંથી કયું માન્ય છબી ફાઈલ સ્વરૂપ છે ?

(a) Img (b) Move (c) Mp3 (d) Png

(4) જ્યારે આપણે છબીના alt ગુણધર્મનો ઉપયોગ કરીશું ત્યારે નીચેનામાંથી શું આપવામાં આવશે ?

(a) વૈકલ્પિક વર્ણન (b) Alt કી વ્યાખ્યાયિત થાય

(c) વૈકલ્પિક છબી (d) વૈકલ્પિક HTML જોડાણ

(5) નીચેનામાંથી કયો ગુણધર્મ છબીની ઊંચાઈ અને પહોળાઈની કિંમત પિક્સેલમાં દર્શાવે છે ?

(a) Img src (b) Height અને width

(c) H અને V (d) આપેલા કોઈ પણ

(6) નીચેનામાંથી કયું માન્ય છબી સ્વરૂપ નથી ?

(a) Imv (b) Png (c) Bmp (d) Gif

(7) આખી છબીને જોડાણ (લિન્ક) તરીકે દર્શાવવા નીચેનામાંથી કયો ખ્યાલ ઉપયોગમાં લેવાય છે ?

(a) હૉટ સ્પૉટ તરીકે છબી (b) હૉટ ટેક્સ્ટ

(c) સક્રિય જોડાણ (એક્ટિવ લિન્ક) (d) આપેલા કોઈ પણ

(8) નીચેનામાંથી Image mapના બે પ્રકારો કયા છે ?

(a) દુકાનદાર બાજુ અને ઉપયોગકર્તા બાજુ (b) સર્વર બાજુ અને ક્લાયન્ટ બાજુ

(c) વેપારી બાજુ અને પૂરા પાડનાર (સપ્લાયર) બાજુ (d) આપેલા બધા

(9) image map ઉમેરવા માટે નીચેનામાંથી કઈ ટૅગ ઉપયોગમાં લેવાય છે ?

(a) Image name (b) Htemp (c) Map (d) Alt

પ્રાયોગિક સ્વાધ્યાય

1. આ પ્રકરણના કોડ લિસ્ટિંગ 3.1માં ચર્ચેલ ઉદાહરણને ધ્યાને લો. તે ચૉકલેટ, સૂકો મેવો અને આઇસ્ક્રીમની છબી છાપે છે. HTML કોડને એ રીતે આગળ વધારો કે જેથી જો તમે ચૉકલેટની છબી પર ક્લિક કરો તો તે તમને ચૉકલેટ વિશેની માહિતી ધરાવતા નવા HTML પાના પર લઈ જાય. તમે ચૉકલેટની થોડી વધુ છબી, ચૉકલેટ વિશેની કેટલીક હકીકતો, ચૉકલેટનો ઇતિહાસ અને ઘરે ચૉકલેટ બનાવવાની વિધિ (રેસીપી) પણ ઉમેરી શકો.

જો તમે સૂકા મેવાની છબી પર ક્લિક કરો. તો તે તમને સૂકા મેવા વિશેની વધુ માહિતી ધરાવતા HTML પાના પર લઈ જશે. એ જ રીતે, જો તમે આઇસ્ક્રીમની છબી પર ક્લિક કરો તો તે તમને વિવિધ આઇસ્ક્રીમ વર્ણવતા નવા HTML પાના પર લઈ જશે.

2. Index.html નામનું પાનું તૈયાર કરો. તમારી જાણીતી ત્રણ હસ્તીઓ(personalities)ની નાની છબી મૂકો. આ હસ્તી તમારી શાળાના શિક્ષક, લેખકો, તમારા મિત્રો, ખેલાડીઓ હોઈ શકે. આ ઉપરાંત ઉપરોક્ત ત્રણેય વ્યક્તિઓની છબીની મોટા કદની છબી સાથે તેમના વિશેની વધુ માહિતી ધરાવતા ત્રણ વેબપેઇજ પણ બનાવો. Index પાના પરની દરેક નાની છબીઓ પર એવી રીતે જોડાણ (લિન્ક) ગોઠવો કે જેથી પસંદ કરેલ વ્યક્તિની સંપૂર્ણ માહિતી ધરાવતા પાના પર લઈ જાય. ત્રણેય નાની છબી સાથે તમારે ત્રણ જુદાં જોડાણ (લિન્ક) કરવાં પડશે.

   (Hint : છબીમાંથી જોડાણ કરવા, <a> એલિમેન્ટ ઉમેરો, અને <a> એલિમેન્ટના href ગુણધર્મમાં મોટી છબીનું જોડાણ મૂકો.)

3. તમારા કુટુંબનું વૃક્ષ (family tree) તૈયાર કરવા, ઉપરોક્ત પ્રથમ ઉદાહરણમાં ચર્ચા કરેલ અભિગમનો ઉપયોગ કરો. તમારા દાદા અને દાદીની એક કરતાં વધુ છબીઓ સાથેનું વેબપેઇજ તૈયાર કરો. અન્ય પાનાં પણ એ રીતે તૈયાર કરો કે જેથી તમે જ્યારે તમારા દાદાની છબી પર ક્લિક કરો, ત્યારે થોડી વધુ સંબંધિત છબીઓ સાથે નવું વેબપેઇજ દર્શાવાય.

4. આ પ્રકરણમાં ચર્ચા કર્યા મુજબ શહેરનું સ્થળ, સ્થિતિ, ચિત્ર રજૂ કરતો image map તૈયાર કરો.

5. ફોટો ગેલેરી ઉમેરીને પ્રકરણ 2માં તમે તૈયાર કરેલ સ્કૂલ વેબસાઇટ પ્રોજેક્ટ પૂરો કરો. મુખ્ય વેબપેઇજ પર તમે તમારી શાળાનો લોગો મૂકો.

6. કોઈ હયાત વીડિયો ફાઈલનો ઉપયોગ કરો અને તેને વેબપેઇજમાં સામેલ કરો. તેને વિકલ્પરૂપે તમે તમારા મોબાઇલ અથવા અન્ય કોઈ સાધન દ્વારા એક વીડિયો ફાઈલ તૈયાર કરી શકો અને તેને વેબપેઇજમાં સામેલ કરી શકો.

4 HTMLમાં યાદી અને કોષ્ટકનો ઉપયોગ

યાદીનું વ્યવસ્થાપન

માહિતી રજૂ કરતી વખતે, ઘણી વખત આપણે વિવિધ અલગ-અલગ મુદ્દાઓ પણ રજૂ કરવાની જરૂર પડતી હોય છે. આવા મુદ્દાઓને ફકરા સ્વરૂપે સળંગ લખવાને બદલે આવા જુદા જુદા મુદ્દા ક્રમ સંખ્યા સહિત કે તેના વગર યાદી સ્વરૂપે રજૂ કરવામાં આવે છે. મુદ્દાઓને આ રીતે રજૂ કરવાથી રજૂઆતની સ્પષ્ટતામાં વધારો થાય છે અને તે વાચકને વાંચવામાં અને સમજવામાં સરળતા બક્ષે છે.

તો ચાલો, એક ઉદાહરણ જોઈએ. જ્યારે તમે બજારમાંથી અમુક ચીજો ખરીદવા ઇચ્છતા હોવ ત્યારે તમારે તેને યાદ રાખવાની જરૂર પડે છે. આકૃતિ 4.1 આવી કેટલીક વસ્તુઓની સરસ રીતે યાદી દર્શાવે છે, જે ખરીદી કરવા જતી વખતે તમે સાથે રાખી શકો છો.

Items to be purchased

- Chocolates
- Notebook
- Practice book
- Ballpen
- Markers

આકૃતિ 4.1 : યાદીનું ઉદાહરણ

HTML યાદીના મૂળભૂત ત્રણ પ્રકાર છેઃ (1) ક્રમબદ્ધ યાદી (ordered list), (2) અક્રમિક યાદી (unordered list) અને (3) વર્ણનાત્મક યાદી (descriptive list). આ તમામ પ્રકાર આ પ્રકરણના આગળ આવતા વિભાગોમાં ચર્ચવામાં આવેલ છે.

અક્રમિક યાદી (Unordered List)

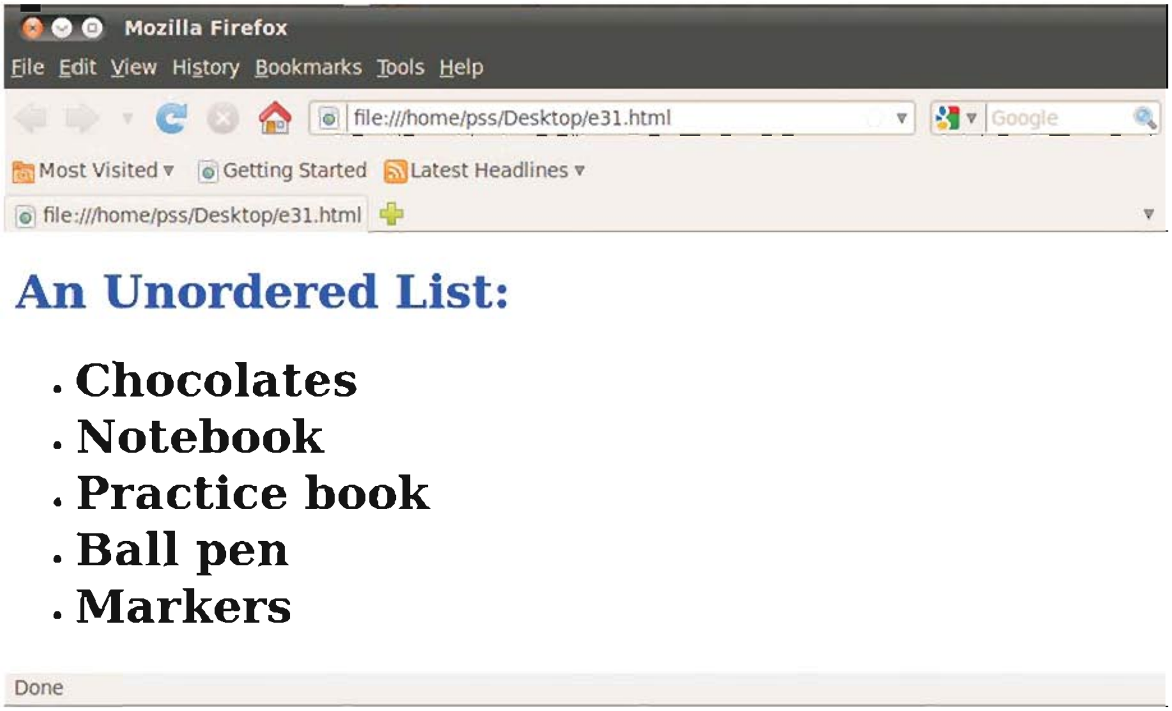
આકૃતિ 4.1માં દર્શાવ્યા મુજબ, અક્રમિક યાદીમાં કોઈ ચિહ્ન વડે યાદીના મુદ્દા દર્શાવાય છે. આ ચિહ્નને 'બુલેટ' (bullet) પણ કહે છે. આકૃતિ 4.1માં દર્શાવેલ યાદી એ બુલેટ તરીકે '●' સાથેની અક્રમિક યાદી છે. HTMLમાં અક્રમિક યાદી તૈયાર કરવા માટે અને ટૅગની જોડી ઉપયોગમાં લેવાય છે. યાદીના એક પછી એક દરેક મુદ્દા અને ટૅગની જોડીમાં આવરીને દર્શાવાય છે. અક્રમિક યાદી તૈયાર કરતાં HTML ઉદાહરણ કોડ-લિસ્ટિંગ 4.1માં દર્શાવેલ છે.

```
<html>
<!-- ---------------------------------------------------------- -->
<body>
<font color ="Blue">
        <h1>An Unordered List:</h1>
</font>
<!-- ---------------------------------------------------------- -->
<ul>
<font size="6">
```

```
<b>
        <li>Chocolates</li>
        <li>Notebook</li>
        <li>Practice book</li>
        <li> Ball pen </li>
    <li>Markers </li>
</b>
</font>
</ul>
<!-- -------------------------------------------------------------- -->
</body>
</html>
```

કોડ-લિસ્ટિંગ 4.1 : અક્રમિક યાદી

SciTE એડિટરમાં કોડ-લિસ્ટિંગ 4.1માં દર્શાવેલ કોડ ટાઈપ કરીને બ્રાઉઝરમાં તેનું પરિણામ ચકાસો. તેનું પરિણામ આબેહૂબ આકૃતિ 4.2માં દર્શાવ્યા મુજબ દેખાશે.



આકૃતિ 4.2 : બ્રાઉઝરમાં અક્રમિક યાદી

બુલેટ બદલવી

સામાન્ય રીતે ભરેલા ગોળ ચિહ્નને બુલેટ તરીકે લેવાય છે. પરંતુ જો આપણે એ ચિહ્ન બદલવું હોય તો, તમારે type નામના ગુણધર્મનો ઉપયોગ કરવો પડે. બુલેટના વિવિધ પ્રકારોમાં ભરેલ વર્તુળ, ખાલી વર્તુળ (disc) અને ચોરસ હોઈ શકે. નીચેનું ઉદાહરણ જુઓ:

```
<ul type="square">
```

બુલેટ તરીકે ખાલી ગોળ ચિહ્નનો ઉપયોગ કરવા માટે આપણે નીચે મુજબ ટેગ લખી શકીએ :

```
<ul type="circle">
```

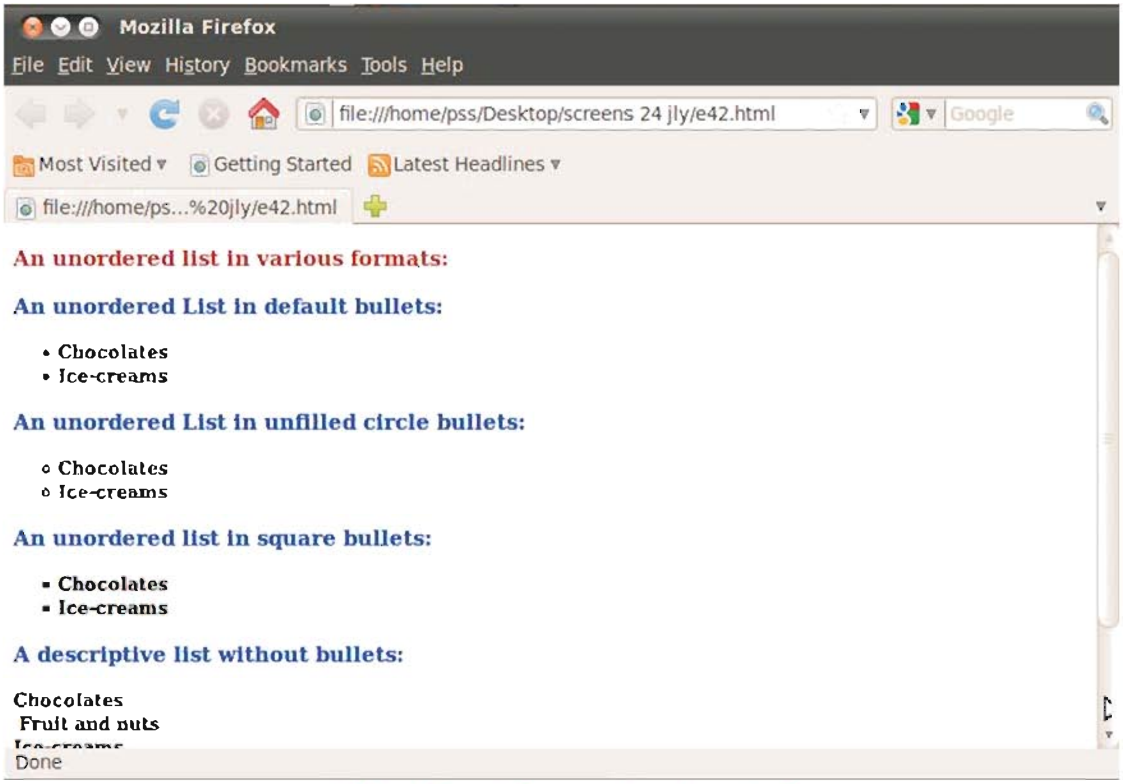
કોડ-લિસ્ટિંગ 4.1માં આપેલા HTML કોડને બદલીને વારાફરતી જુદી જુદી જાતની બુલેટનું અવલોકન કરો.

બુલેટ વગરની યાદી

જો કોઈ કિસ્સામાં તમારે યાદીમાં મુદ્દાની આગળ કોઈ પણ જાતની બુલેટ ન દર્શાવાય તેવું ઇચ્છતા હોય તો, તમે <dl> અને </dl> નો ઉપયોગ કરીને વર્ણનાત્મક યાદી તૈયાર કરી શકો. અહીં, 'dl' એ description listનું ટૂંકું નામ છે. વર્ણનાત્મક યાદીની અંદર વિવિધ મુદ્દા રજૂ કરવા આપણે દરેક નવા મુદ્દાને <dt> અને </dt> ટૅગનો ઉપયોગ કરવો પડે. એટલે કે, મુદ્દા તરીકે Chocolate દર્શાવવા માટે નીચે મુજબ ટેગ લખી શકાય:

```
<dl>
        <dt> Chocolates </dt>
</dl>
```

એમાંય વળી, પેટાવિગત દર્શાવવા માટે આપણે વર્ણનાત્મક ટૅગ <dd> અને </dd> નામની જોડીનો ઉપયોગ કરી શકીએ. એટલે કે, આપણે ચોકલેટ અને આઇસ્ક્રીમની વિવિધ જાતો ઉમેરી શકીએ. આકૃતિ 4.3 વેબપેઇજમાં જુદી જુદી જાતની બુલેટનો ઉપયોગ દર્શાવે છે.



આકૃતિ 4.3 : બુલેટના જુદા જુદા પ્રકારો

આકૃતિ 4.3 જેવું પરિણામ દર્શાવવા માટેના HTML કોડ કોડ-લિસ્ટિંગ 4.2માં દર્શાવેલ છે.

```
<html>
<body>
<font color ="Brown">
      <h1>An Unordered List in various formats: </h1>
</font>
<!-- ---------------------------------------------------- -->
```

```
<font color ="Blue">    <h1>An unordered List in default bullets: </h1>      </font>
<ul>
<font size="6">
<b>
        <li>Chocolates     </li>
        <li>Ice-creams     </li>
        </b>
</font>
</ul>
<!-- ------------------------------------------------------- -->
<font color ="Blue">   <h1>An unordered List in unfilled circle bullets: </h1>  </font>
<ul type="circle">
<font size="6">
<b>
        <li>Chocolates     </li>
        <li>Ice-creams     </li>
</b>
</font>
</ul>
<!-- ------------------------------------------------------- -->
<font color ="Blue">  <h1>An unordered list in square bullets: </h1> </font>
<ul type="square">
<font size="6">
<b>
        <li>Chocolates     </li>
        <li>Ice-creams     </li>
        <li>Dry fruits     </li>
</b>
</font>
</ul>
<!-- ------------------------------------------------------- -->
<font color ="Blue">
        <h1>A descriptive list without bullets: </h1>
</font>
<dl>
<font size="6">
<b>
        <dt>Chocolates     </dt>
                <dd> Dark </dd>
                <dd> Fruit and nuts </dd>
                <dd> Milk  </dd>
```

```
        <dt>Ice-creams      </dt>
              <dd> Vanilla </dd>
              <dd> Chocolate chips </dd>
        <dt>Dry fruits      </dt>
              <dd> Almonds </dd>
              <dd> Cashew nuts   </dd>
</b>
</font>
</dl>
<!-- ------------------------------------------------------------ -->
</body>
</html>
```

કોડ-લિસ્ટિંગ 4.2 : વિવિધ પ્રકારની બુલેટ્સ માટેના HTML કોડ

યાદીનું મથાળું

યાદીનું મથાળું આપવા માટે આપણે ફક્ત <lh> ટૅગનો ઉપયોગ કરવો પડે. સ્વાભાવિક રીતે જ, મથાળું યાદીની પહેલાં જ આવવું જોઈએ. આમ, <li> અને ટૅગની જોડી લખતાં પહેલાં જ આપણે જરૂરી મથાળા સાથે <lh> ટૅગનો ઉપયોગ કરવો પડે. નીચે આપેલ ઉદાહરણ જુઓ :

```
<lh> My Shopping List </lh>
```

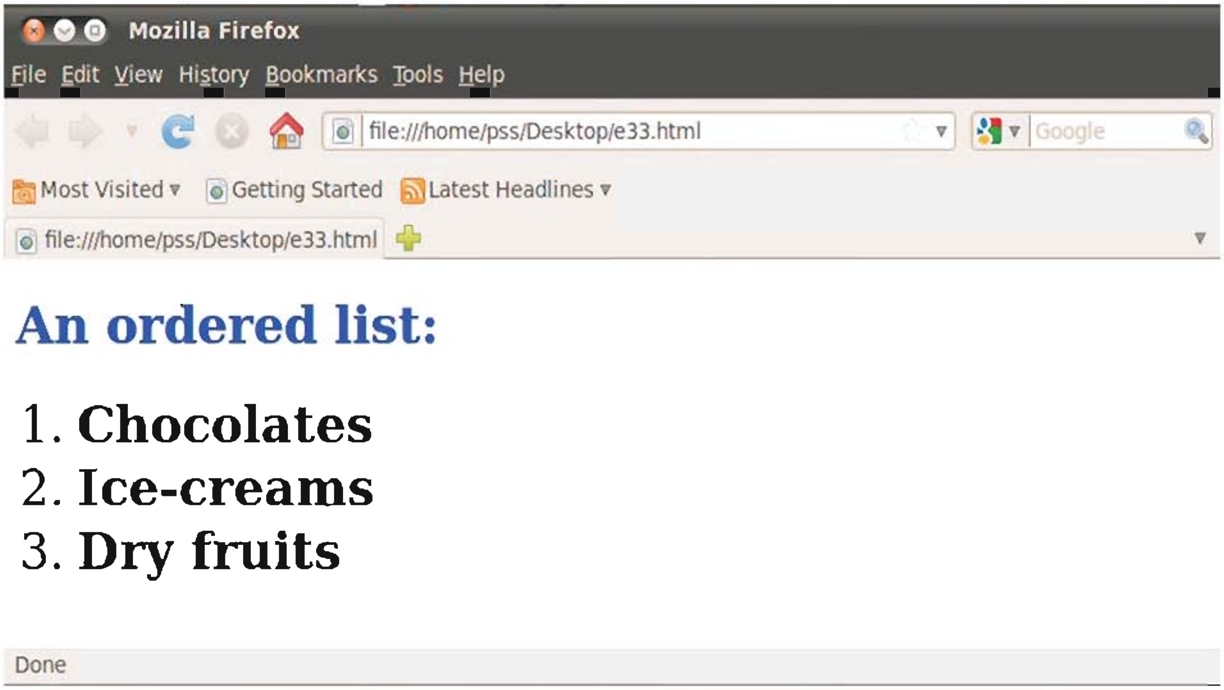
ક્રમબદ્ધ યાદી

ક્રમબદ્ધ યાદીમાં બુલેટ્સને બદલે ક્રમ દર્શાવતા આંકડા કે અક્ષરો ધરાવતા મુદ્દાઓ હોય છે. HTML માં આવી ક્રમબદ્ધ યાદી બનાવવા માટે અને ટૅગનો ઉપયોગ થાય છે. હરહંમેશની જેમ યાદીમાંના દરેક મુદ્દાને અને ટૅગની વચ્ચે આવરીને દર્શાવાય છે. ક્રમબદ્ધ યાદી બનાવતું HTML ઉદાહરણ કોડ-લિસ્ટિંગ 4.3માં દર્શાવેલ છે.

```
<html>
<body>
<font color ="Blue">
        <h1>An ordered list: </h1>
</font>
<!-- ------------------------------------------------------------------------ -->
<ol >
<font size="6">
<b>
        <li>Chocolates      </li>
        <li>Ice-creams      </li>
        <li>Dry fruits      </li>
</b>
</font>
</ol>
<!-- ------------------------------------------------------------------------ -->
</body>
</html>
```

કોડ-લિસ્ટિંગ 4.3 : ક્રમબદ્ધ યાદી માટે HTML કોડ

કોડ-લિસ્ટિંગ 4.3નું પરિણામ આકૃતિ 4.4માં દર્શાવેલ છે.



આકૃતિ 4.4 : ક્રમબદ્ધ યાદી

સામાન્ય રીતે, ક્રમબદ્ધ યાદી હંમેશાં ક્રમ 1થી શરૂ થાય છે. તમે જો ઇચ્છો તો કોઈ ચોક્કસ ક્રમ સંખ્યાથી પણ યાદી શરૂ કરી શકો. ધારો કે, તમારે યાદીના પહેલાં મુદ્દાની શરૂઆત ક્રમ 6થી કરવી હોય તો તમે નીચે દર્શાવ્યા મુજબ <ol> ટૅગ સાથે start ગુણધર્મનો ઉપયોગ કરી શકો.

```
<ol start ="6">
```

ક્રમ 6થી યાદીની શરૂઆત કરવા માટે કોડ-લિસ્ટિંગ 4.3માં આપેલ HTML કોડને સુધારો.

અક્ષરક્રમ ધરાવતી ક્રમબદ્ધ યાદી

આંકડાની જગ્યાએ, આપણે ક્રમ દર્શાવવા અંગ્રેજી મૂળાક્ષરો 'A', 'B', 'C' અથવા 'a', 'b', 'c'. આપણે રોમન અંકોનો પણ ઉપયોગ કરી શકીએ. આવું ટૅગની સાથે type ગુણધર્મનો ઉપયોગ કરીને કરી શકાય. ટેબલ 4.1 type ગુણધર્મ માટેની શક્ય કિંમત રજૂ કરે છે.

કિંમત	વર્ણન
1	અંક
A	અંગ્રેજી મોટા મૂળાક્ષરો
a	અંગ્રેજી નાના મૂળાક્ષરો
I	રોમન મોટા અંકો
i	રોમન નાના અંકો

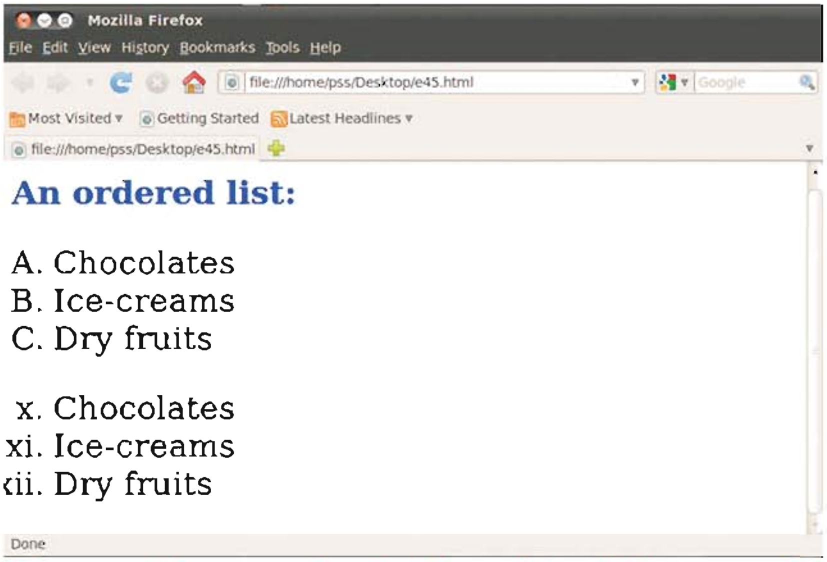
કોષ્ટક 4.1 : Type ગુણધર્મની કિંમતો

કોડ-લિસ્ટિંગ 4.4માં HTML કોડ આપવામાં આવેલ છે, આ કોડ એક અંગ્રેજી મૂળાક્ષર સાથેની અને એક ક્રમ 10થી શરૂ થતા રોમન અંક સાથેની એમ બે ક્રમબદ્ધ યાદીઓ રજૂ કરશે.

```
<html>
<body>
<font color ="Blue">
        <h1>An ordered list: </h1>
</font>
<!-- --------------------------------------------------------------------- -->
<font size="6">
</b>
        <ol type=A>
        <li>Chocolates    </li>
        <li>Ice-creams    </li>
        <li>Dry fruits    </li>
        </ol>
</b>
</font>
<!-- --------------------------------------------------------------------- -->
<font size="6">
</b>
        <ol type='i' start="10">
        <li>Chocolates    </li>
        <li>Ice-creams    </li>
        <li>Dry fruits    </li>
</ol>
</b>
</font>
<!-- --------------------------------------------------------------------- -->
</body>
</html>
```

કોડ-લિસ્ટિંગ 4.4 : બુલેટ તરીકે રોમન અંકો છાપવા માટેના HTML કોડ

HTML કોડ 4.4નું પરિણામ આકૃતિ 4.5માં આપવામાં આવેલ છે. આ રીતે તમે type ગુણધર્મની જુદી જુદી કિંમતો સાથે પ્રયત્ન કરી શકો છો.



આકૃતિ 4.5 : અંગ્રેજી મૂળાક્ષરો અને રોમન અંકો સાથેની ક્રમબદ્ધ યાદી

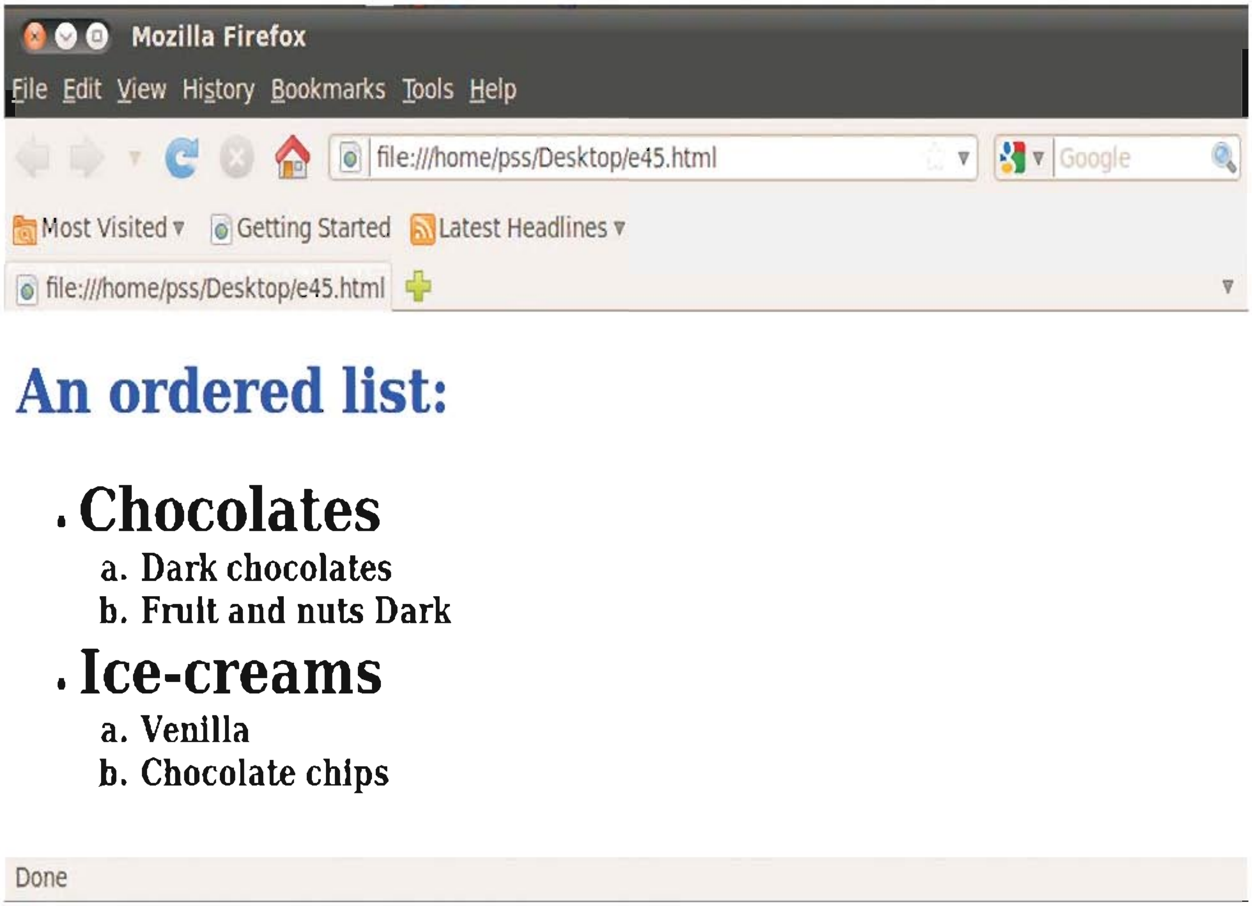
નેસ્ટેડ યાદી

કોઈ એક યાદીમાંય વળી બીજી યાદી સમાવવી પડે તેવી પણ જરૂરિયાત હોય છે. આને યાદીનું નેસ્ટિંગ અથવા નેસ્ટેડ યાદી કહે છે. વળી, આવા કિસ્સામાં પ્રથમ યાદી અને એની અંદર રહેલી બીજી યાદીનો પ્રકાર જુદો જુદો પણ હોઈ શકે. ઉદાહરણરૂપ HTML કોડ એ કોડ-લિસ્ટિંગ 4.5માં આપેલ છે.

```
<html>
<body>
<font color ="Blue">
        <h1>An ordered list: </h1>
</font>
<!-- --------------------------------------------------------------------- -->
<ul >
<font size= "6 ">
<b>
        <li>Chocolates      </li>
        <font size= "4">
        <ol type="a">
                <li> Dark chocolates            </li>
                <li> Fruit and nuts Dark     </li>
</ol>
</font>
<!-- --------------------------------------- -->
<li>Ice-creams      </li>
<ol type="a">
        <font size="4">
                <li> Venilla                    </li>
                <li> Chocolate chips           </li>
        </font>
</ol>
<!-- --------------------------------------------------------------------- -->
</b>
</font>
</ul>
</body>
</html>
```

કોડ-લિસ્ટિંગ 4.5 : નેસ્ટેડ યાદી માટેના HTML કોડ

કોડ-લિસ્ટિંગ 4.5નું પરિણામ આકૃતિ 4.6માં દર્શાવેલ છે.



આકૃતિ 4.6 : HTMLમાં નેસ્ટેડ યાદી

લખાણને વધુ માળખાગત રીતે રજૂ કરી શકે તેવાં માળખાં પણ છે. આવા એક માળખાને કોષ્ટક કહે છે. નીચેનો વિભાગ HTMLમાં કોષ્ટક રજૂ કરે છે.

HTMLમાં કોષ્ટકનું વ્યવસ્થાપન

માહિતીની યોગ્ય સ્વરૂપે રજૂઆત સમજવામાં અને ઉપયોગ કરવામાં સરળતા બક્ષે છે. શું તમે કોઈ ખરીદીના બિલને સળંગ ફકરા સ્વરૂપે છપાયેલું હોય તેવી કલ્પના કરી શકો છો. વિવિધ વિષયોના ગુણ રજૂ કરતા તમારા ગુણપત્રક(માર્કશીટ)નું શું ? તે આકૃતિ 4.7માં દર્શાવ્યા જેવી મળે તો તે તમને વાંચવા માટે ચોક્કસપણે અઘરી પડે. તે માત્ર વધુ પ્રયત્ન માગી લે તેવી જ નહીં બલ્કે વધુ સમય લગાડનારી પણ હશે. ઉપરાંત, તેના દ્વારા માહિતીનું ખોટું અર્થઘટન પણ થઈ જાય એવુંય બને.

Modern School Affiliated to Star Education, Gujarat Mr. Arvind B Patel Seat No 123 March 2013 Subjects: English, Gujarati, Science, Maths, SS*

Internal Marks (40): 20, 30, 25, 35, 25 External Marks (60): 35, 32, 48, 40, 35 Class: First Class SS*: Social Science

આકૃતિ 4.7 : વિચિત્ર પ્રકારના ગુણપત્રકનું ઉદાહરણ

હવે, આકૃતિ 4.8 જુઓ. દેખીતી રીતે જ તેમાં કોઈ ગૂંચવણ નથી. શાળાનું નામ, વિદ્યાર્થીનું નામ, તારીખ, તમામ વિષયોના ગુણ અને પરિણામ સ્પષ્ટપણે દેખાઈ આવે છે.

Modern School

Affiliated to Star Education, Gujarat

Mr. Arvind B Patel	Seat No 123	March 2013

Subject:	English	Gujarati	Science	Maths	SS *
Internal Marks (40):	20	30	25	35	25
External Marks (60):	35	32	48	40	35

Class: First class

SS: Social Science*

આકૃતિ 4.8 : યોગ્ય સ્વરૂપના ગુણપત્રકનો નમૂનો

માહિતીની સુનિયોજિત રજૂઆત માટે HTMLમાં કોષ્ટક વધુ કાર્યક્ષમ છે. કોષ્ટકમાં ખાનાંઓ સ્વરૂપે માહિતી રજૂ થાય છે. તમારા ગણિતની ગણતરીઓ માટે કદાચ તમે ઊભાં અને આડાં ખાનાં ધરાવતી વિશિષ્ટ નોંધપોથી વાપરી હશે. એ જ રીતે, કોષ્ટકમાં માહિતીને ઊભી અને આડી હાર સ્વરૂપે રજૂ કરી શકાય. હવે પછીના વિભાગમાં HTML ટૅગના ઉપયોગ દ્વારા વેબપેઇજ પર કેવી રીતે એક નાનું કોષ્ટક દર્શાવી શકાય. તે વર્ણવવામાં આવ્યું છે. એ પછી, કોષ્ટકનું સર્જન કરતાં દરેક ટૅગને વિગતવાર જોઈશું.

એક નાના કોષ્ટકનું સર્જન

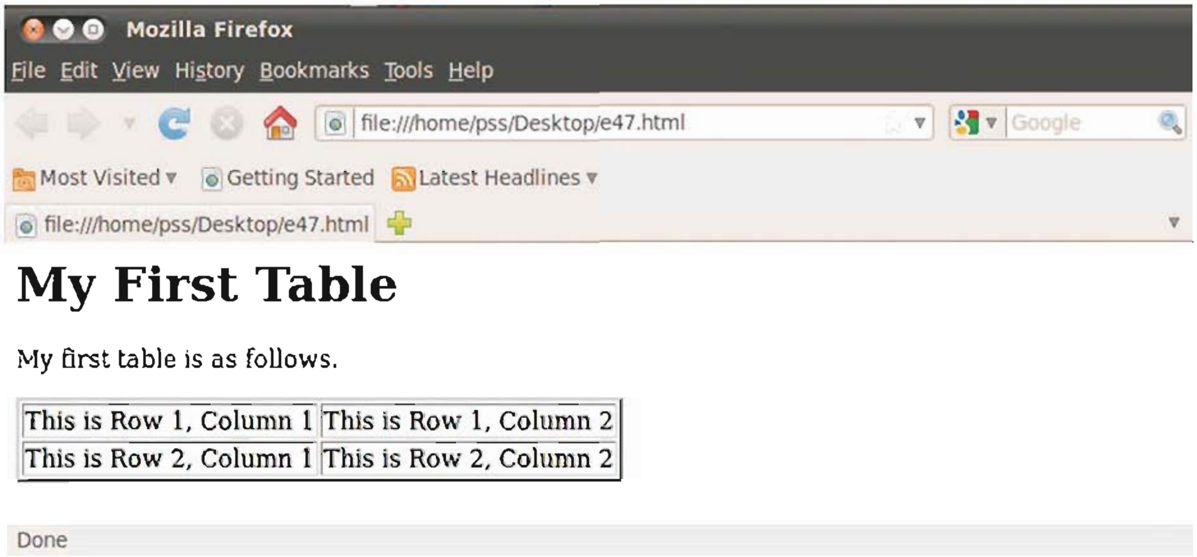
HTMLમાં કોષ્ટક તૈયાર કરવા <table> અને </table> ટૅગનો ઉપયોગ થાય છે. એટલે કે, કોષ્ટકની શરૂઆત અને અંત એ આ બે ટૅગ દ્વારા દર્શાવાય છે. <table> અને </table> ટૅગની વચ્ચે આપણે કોષ્ટકનું શીર્ષક, કોષ્ટકની કિનારી, કોષ્ટક માટે ઊભી અને આડી હરોળ જેવા ગુણધર્મોનો ઉપયોગ કરી શકીએ.

કોષ્ટકનું મુખ્ય લખાણ ઊભી હાર અને આડી હરોળ સ્વરૂપે ગોઠવાય છે. કોષ્ટકમાં આડી હરોળ <tr> ટૅગ દ્વારા દર્શાવવામાં આવે છે. કોષ્ટકની પ્રથમ આડી હરોળ મથાળા માટેની હોય છે, જે <th> ટૅગ દ્વારા દર્શાવાય છે. એ પછીની બાકીની આડી હરોળ <td> ટૅગ દ્વારા દાખલ કરવામાં આવે છે. કોડ-લિસ્ટિંગ 4.6 જુઓ, જેમાં ઉદાહરણરૂપ HTML કોડ દર્શાવવામાં આવેલ છે.

```
<html>
<body>
<h1>My First Table </h1>
<p>My first table is as follows.</p>
<table border="2">
<tr>
     <td> This is Row 1, Column 1 </td>
     <td> This is Row 1, Column 2 </td>
</tr>
<tr>
     <td> This is Row 2, Column 1 </td>
<td> This is Row 2, Column 2 </td>
</tr>
</table>
</body>
</html>
```

કોડ-લિસ્ટિંગ 4.6 : સાદા ટેબલ માટેના HTML કોડ

તમે તમારા એડિટરમાં આ કોડ લખીને બ્રાઉઝરમાં પરિણામ જોઈ શકો. બ્રાઉઝરમાં જોવાથી પરિણામ આકૃતિ 4.9 જેવું જોવા મળશે.



આકૃતિ 4.9 : વેબપેઇજ પર દર્શાવાયેલ એક સાદું HTML કોષ્ટક

કોડ-લિસ્ટિંગ 4.6માં દર્શાવેલ HTML કોડ થોડા ગુણધર્મો અને ભાગ સાથે table ટૅગનો ઉપયોગ કરે છે. Table ટૅગની અંદર ઉપયોગમાં લેવાયેલ પ્રથમ ગુણધર્મ પોતે જ 2 પિક્સેલ કદની કિનારી દર્શાવવા માટેનો border ગુણધર્મ છે. કોષ્ટકમાં એ પછી <tr> ટૅગ સાથે આડી હરોળ વ્યાખ્યાયિત કરે છે. આકૃતિ 4.9માં દર્શાવેલ કોષ્ટકને બે આડી હરોળ છે, આ માટે <tr> ટૅગની બે જોડી ઉપયોગમાં લેવી પડે.

આડી હરોળની અંદર <td> ટૅગની મદદથી ઊભી હરોળ વ્યાખ્યાયિત કરી શકાય છે. આ ઉદાહરણમાં ઉપયોગમાં લેવાયેલ ટૅગ અને ગુણધર્મોની ચર્ચા આ પ્રકરણમાં પછીથી કરવામાં આવેલ છે. ઉદાહરણનો ઉદ્દેશ કોષ્ટકની રચનામાં ઊભી અને આડી હરોળ સમજવા માટે છે. આ પ્રકારની વ્યવસ્થાને ખાનાવાળી માહિતી (grid type) તરીકે પણ ઓળખવામાં આવે છે. તો ચાલો, કોષ્ટક માટેના અન્ય ટૅગ વિશે વિસ્તૃત માહિતી મેળવીએ.

Table ટૅગ અને તેના ગુણધર્મો

કોષ્ટકને વ્યાખ્યાયિત કરવા માટે ઉપયોગમાં લેવાતો પ્રથમ ટૅગ એ table ટૅગ છે. <table> ટૅગમાં ઘણા ગુણધર્મો સમાવી શકાય છે. તેમાંના કેટલાક આજકાલ ખાસ પ્રચલિત નથી. જો કે, તેમ છતાં તમે એને વાપરી શકો. આવા ગુણધર્મો કોષ્ટક 4.2માં દર્શાવાયેલ છે.

align	આ ગુણધર્મ કોષ્ટકની ગોઠવણી માટે ઉપયોગી છે.
bgcolor	આ ગુણધર્મ કોષ્ટકના બૅકગ્રાઉન્ડ રંગ બાબતે જણાવે છે.
border	આ ગુણધર્મ કોષ્ટકની કિનારી બાબતે જણાવે છે.
cellpadding	ખાનાની ધાર (edges of cell) અને ખાનાની અંદરની માહિતી વચ્ચે નક્કી કર્યા મુજબની જગ્યા છોડવા આ ગુણધર્મ દ્વારા જણાવી શકાય છે.
cellspacing	આ ગુણધર્મ કોષ્ટકના દરેક ખાનાની વચ્ચે કેટલી જગ્યા રાખવી તે જણાવે છે.
dir	આ ગુણધર્મ કોષ્ટકમાં દર્શાવાતા લખાણની દિશા નક્કી કરવા માટે ઉપયોગી છે.
frame	આ ગુણધર્મ કોષ્ટકની છેક બહારની બાજુની કિનારીને નિયંત્રિત કરે છે.
rules	આ ગુણધર્મ કોષ્ટકની અંદરની કિનારીની રજૂઆતને નિયંત્રિત કરે છે.
summary	આ ગુણધર્મ કોષ્ટકનું વર્ણન રજૂ કરે છે.
width	આ ગુણધર્મ કોષ્ટકની પહોળાઈ કેટલી રાખવી તે જણાવે છે.

કોષ્ટક 4.2 : Table ટૅગના ગુણધર્મો

સામાન્યપણે ઉપયોગમાં લેવાતા કેટલાક ગુણધર્મો નીચે ચર્ચવામાં આવેલ છે :

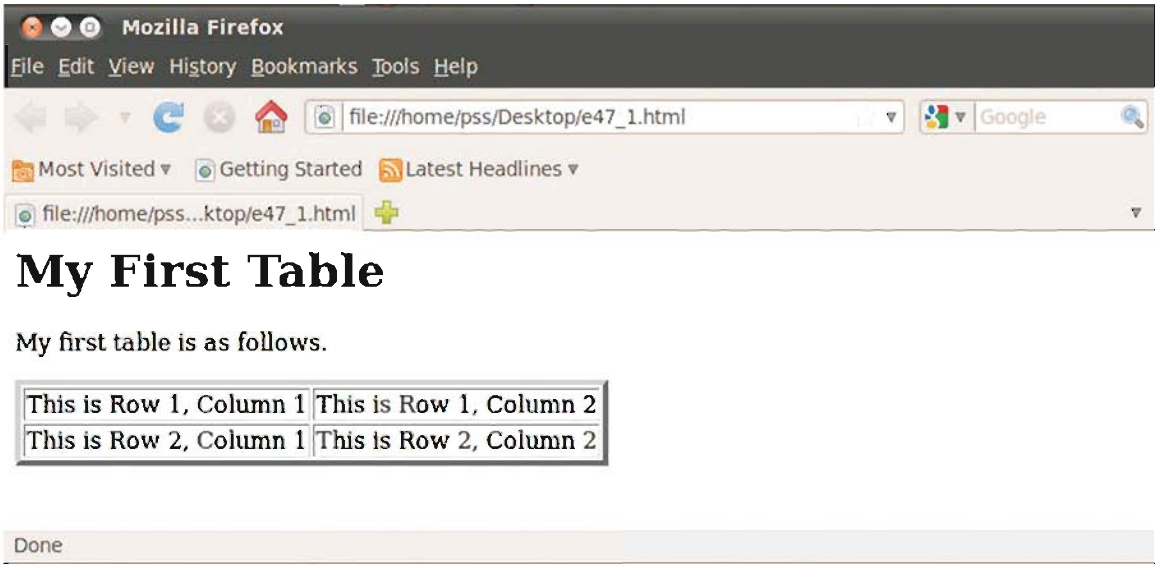
Border ગુણધર્મ

કોડ-લિસ્ટિંગ 4.6માં ચર્ચેલ ઉદાહરણમાં border ગુણધર્મનો ઉપયોગ કરીને 2 પિક્સેલ માપની કિનારી બનાવવામાં આવે છે. આ ગુણધર્મ કોષ્ટકની ફરતે ચારે બાજુ તેમજ દરેક ખાનાની આસપાસ કિનારી દોરે છે. કિનારીની પહોળાઈ પિક્સેલ સ્વરૂપે અપાય છે. Border ગુણધર્મનો ઉપયોગ કરવો મરજિયાત છે. જો તમે ગુણધર્મ તરીકે 0 કિંમત આપશો તો કોઈ કિનારી દેખાશે નહીં. અગાઉ કોડ-લિસ્ટિંગ 4.6નો ઉપયોગ કરીને તૈયાર કરાયેલ HTML કોડ યાદ કરો. ચાલો, નીચે દર્શાવ્યા પ્રમાણે આપણે Border ગુણધર્મની કિંમત સુધારીએ :

```
<table border="4" align="left">
```

હવે, ફરી એકવાર કોષ્ટકનો દેખાવ ચકાસી લો. તે આકૃતિ 4.10માં દર્શાવ્યા મુજબ દેખાવી જોઈએ.

આગળના ફકરામાં આપેલ ઉદાહરણમાં આપણે ઉપયોગમાં લીધેલ એક વધુ ગુણધર્મ કદાચ તમારા ધ્યાને આવ્યો હશે. આ ગુણધર્મ છે align. શું તમે કોઈ બાબતની નોંધ લીધી ? તમે કદાચ કોઈ ફેરફાર નહીં નોંધ્યો હોય. કોષ્ટક જમણી બાજુ ગોઠવવા ગોઠવણી (alignment) બદલો. હવે તમે કોષ્ટક ગોઠવણી(table alignment)માં ફેરફાર જોઈ શકશો. અન્ય ગોઠવણી (alignment) ની કિંમતો પછીથી જોઈશું.



આકૃતિ 4.10 : કિનારીની બદલાયેલ કિંમત સાથેનાં કોષ્ટક

Bgcolor ગુણધર્મ

ચાલો, કોષ્ટકમાં બૅકગ્રાઉન્ડ રંગ મૂકીને આપણે કોષ્ટકને વધુ રંગીન અને આકર્ષક બનાવીએ. bgcolor ગુણધર્મ કોષ્ટકનો બૅકગ્રાઉન્ડ રંગ ગોઠવવાનું કાર્ય કરે છે. આ ગુણધર્મની કિંમત કોઈ રંગનું નામ અથવા છ આંકડાનો સોળઅંકી સંખ્યા લેખન પદ્ધતિનો કોડ હોઈ શકે. તાજેતરમાં તમે જેનું પરીક્ષણ કરતા હતા તે HTML કોડ (કોડ-લિસ્ટિંગ 4.6માં આપ્યા મુજબ)ની પ્રથમ લીટી સુધારો.

```
<table border="4"   align="right"   bgcolor="yellow">
```

બ્રાઉઝરમાં કોડ નિહાળીને પરિણામ ચકાસો.

Cellpadding ગુણધર્મ

જો કોષ્ટકના બે ખાનાં એકબીજાની બાજુમાં મૂકવામાં આવે અને બંનેમાં શાબ્દિક લખાણ હોય તો થોડી સમસ્યા ઊભી થઈ શકે. જો બંને ખાનાંની ધાર અને લખાણ વચ્ચે પૂરતી જગ્યા નહીં હોય તો બંનેના શબ્દો ભેગા થઈ

ગયેલા લાગશે અને તેના કારણે વાંચવામાં તકલીફ પડશે. એ જ રીતે, જો દરેક ખાનાની ફરતે કિનારી હશે અને લખાણ આ કિનારીને અડતું હશે, તો તે વાંચવામાં તકલીફ પડશે. આવા કિસ્સામાં જો ખાનાંઓમાં થોડી વધુ જગ્યા (space as padding) રાખવામાં આવે તો તે ખાનાઓનાં લખાણ વાંચવામાં સરળતા રહે.

કોષ્ટકના બે ખાનાંઓની કિનારી અને તેના લખાણની વચ્ચે થોડી જગ્યા મૂકવા cellpadding ગુણધર્મનો ઉપયોગ કરવામાં આવે છે. cellpadding ગુણધર્મ કોષ્ટકના ખાનાઓની દરેક દીવાલ(wall of cell)ની અંદરની બાજુ થોડી જગ્યાની ગાદી (pad) બનાવે છે. આ ગુણધર્મની કિંમત ટકાવારીમાં અથવા પિક્સેલમાં આપી શકાય. નીચેનું ઉદાહરણ જુઓ.

```
<table border="4"   align="center"   bgcolor="pink"   cellpadding="25">
```

આના બદલે નીચે મુજબ તમે ટકાવારીનો ઉપયોગ કરી શકો :

```
<table border="4"   align="center"   bgcolor="pink"   cellpadding="20%">
```

તમે કદાચ એ નોંધ્યું હશે કે, આપણે align અને bgcolor ગુણધર્મની કિંમતો પણ બદલી છે. કોષ્ટકના બદલાયેલા દેખાવને બ્રાઉઝરમાં ચકાસો.

Cellspacing ગુણધર્મ

cellspacing ગુણધર્મ, કોષ્ટકના ખાનાઓની વચ્ચે જગ્યા કરવા માટે ઉપયોગમાં લેવાય છે. કેટલી જગ્યા રાખવી તે માટેની કિંમત ટકાવારીમાં અથવા પિક્સેલમાં દર્શાવી શકાય છે. અહીં, ટકાવારી એટલે કોષ્ટકની દરેક ખાનાની પહોળાઈના ટકા. નીચેનું ઉદાહરણ જુઓ :

```
<table border="4"   align="center"   bgcolor="yellow"   cellspacing="20%">
```

અથવા

```
<table border="4"   align="center"   bgcolor="yellow"   cellspacing="25">
```

Width ગુણધર્મ

width ગુણધર્મ કોષ્ટકની પહોળાઈ નક્કી કરવા માટે ઉપયોગમાં લેવાય છે. આ પહોળાઈ કેટલી રાખવી તેની કિંમત હરહંમેશની જેમ પિક્સેલમાં અથવા ઉપલબ્ધ જગ્યાની ટકાવારીમાં આપી શકાય છે.

```
<table border="4"  align="center"  bgcolor="pink"  cellspacing="25" frame="box"
rules="cols"   width="50%">
```

અથવા

```
<table border="4"  align="center"  bgcolor="pink"  cellspacing="25" frame="box"
rules="cols"   width="50">
```

Align ગુણધર્મ

align ગુણધર્મ આડી હારમાં આવેલા તમામ ખાનાઓના લખાણનું સ્થાન નક્કી કરે છે. સાદા લખાણની જેમ જ કોષ્ટકના ખાનાની અંદર લખાણ પણ ગોઠવી શકાય છે. કોષ્ટક 4.3 align ગુણધર્મ માટેની શક્ય કિંમતોની યાદી રજૂ કરે છે.

કિંમત	વર્ણન
left	લખાણને ખાનાની ડાબી તરફ ગોઠવવામાં આવે છે. સામાન્ય લખાણ માટે મોટાભાગે આ જ ગોઠવણી થાય છે.
right	લખાણને જમણી તરફ ગોઠવવામાં આવે છે.
center	લખાણ ખાનાની અંદર મધ્યમાં ગોઠવે છે.
justify	ખાનામાં રહેલા લખાણને એ રીતે ગોઠવવામાં આવે છે કે જેથી તે આખા ખાનામાં સમાઈ જાય.
char	ખાનાનું લખાણ કોઈ ચોક્કસ અક્ષરની પ્રથમ ઉપસ્થિતિ(instance)ની આસપાસ આડું ગોઠવાય છે. (ઉદાહરણ તરીકે, સંખ્યાઓ દશાંશ સ્થાનની પ્રથમ ઉપસ્થિતિને આધારે ગોઠવાય છે.)

કોષ્ટક 4.3 : align ગુણધર્મ માટે શક્ય કિંમતો

જો align ગુણધર્મની કિંમત char હોય તો આડી હરોળની અંદર આવેલા કોષ્ટકના દરેક ખાનાનું લખાણ કોઈ એક ચોક્કસ અક્ષરની પ્રથમ ઉપસ્થિતિની આસપાસ ગોઠવાશે. આવા નક્કી કરેલા ચોક્કસ અક્ષરને ''ધરી અક્ષર'' (axis character) તરીકે ઓળખવામાં આવે છે. આ ગુણધર્મ માટેનો આવો ''ધરી અક્ષર'' સામાન્ય સંજોગોમાં દશાંશ ચિહ્ન (.) હોય છે. લખાણની ગોઠવણી માટે char તરીકે દશાંશ ચિહ્ન (char=".")નો ઉપયોગ કરવાથી ખાનામાં રહેલી દશાંશ સંખ્યાઓ દર્શાવ્યા મુજબ દશાંશ સ્થાનના આધારે ગોઠવાશે.

	1	2	3	.	5	6	
		4	6	.	1	2	7
3	8	1	6	.	4	5	3

table ટૅગ અને તેના ગુણધર્મો વિશે આટલી વિસ્તૃત ચર્ચા કર્યા પછી ચાલો હવે પછીના વિભાગમાં કોષ્ટકની આડી હરોળ માટેની <tr> ટૅગ જોઈએ.

કોષ્ટકની આડી હરોળ માટેની ટૅગ

કોષ્ટકમાં આડી હરોળ દર્શાવવા માટે <tr> ટૅગનો ઉપયોગ કરવામાં આવે છે. <tr> ટૅગની અંદર આવતી દરેક બાબત એ જ લીટી પર દર્શાવવી જોઈએ. તે કોષ્ટક 4.4માં દર્શાવ્યા મુજબ તેમાં ત્રણ ગુણધર્મો દર્શાવી શકાય.

ગુણધર્મ	વર્ણન
align	આડી હરોળનું લખાણ નક્કી કર્યા મુજબ ગોઠવાય છે.
bgcolor	પસંદ કરેલ આડી હરોળનો બૅકગ્રાઉન્ડ રંગ નક્કી કરે છે.
valign	આડી હરોળના દરેક ખાનાના લખાણની ઊભી સમાંતર ગોઠવણી કરવાનું નક્કી કરે છે.

કોષ્ટક 4.4 : <tr> ટૅગના ગુણધર્મો

તો ચાલો, હવે જોઈએ કે આ ગુણધર્મોનો HTML કોડમાં ઉપયોગ કેવી રીતે કરી શકાય.

Align ગુણધર્મ

align ગુણધર્મ આડી હરોળના બધા ખાનાંમાંના લખાણનું સ્થાન દર્શાવે છે. Align ગુણધર્મનું સામાન્ય સ્વરૂપ નીચે મુજબ છે :

align="alignment"

align ગુણધર્મની કિંમત તરીકે કોષ્ટક 4.3માં દર્શાવ્યા મુજબ left, right, center, justify અને char પૈકીની એક હોઈ શકે. કોષ્ટક રજૂ કરતા કોઈ પણ HTML કોડમાં આ ગુણધર્મોનો ઉપયોગ કરો.

Bgcolor ગુણધર્મ

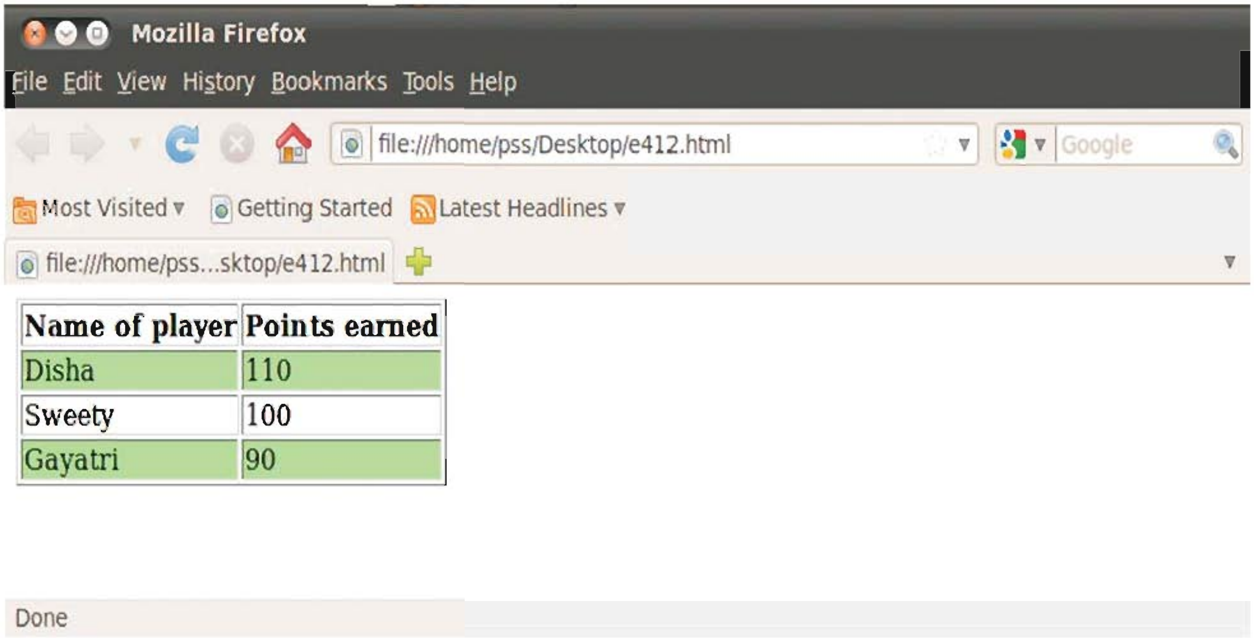
અગાઉ કોષ્ટકના બૅકગ્રાઉન્ડના રંગ પૂરવા આપણે bgcolor ગુણધર્મ જોયો, જેમાં આપણે રંગનું નામ લખીને અથવા રંગના સોળઅંકી કોડ આપીને રંગ દર્શાવતા હતા. હવે જો કોષ્ટકની કોઈ એક આડી હારને ચોક્કસ રંગથી રંગવી હોય તો શું? ફરીથી આ માટેનો જવાબ છે ચોક્કસ રંગનું નામ અથવા નિયત સોળઅંકી કોડ ધરાવતો bgcolor ગુણધર્મ. bgcolor ગુણધર્મ આડી હરોળ માટેનો રંગ ગોઠવે છે.

કોષ્ટકની એકાંતરે આપેલી આડી હરોળને જુદા જુદા રંગ વડે રંગવા <tr> ટૅગ સાથે bgcolor ગુણધર્મ બહુ સામાન્યપણે ઉપયોગમાં લેવાય છે. આમ કરવાથી દરેક આડી હરોળને વાંચવામાં સરળતા રહે છે. કોડ-લિસ્ટિંગ 4.7માં આપેલ HTML કોડના ઉદાહરણનો પ્રયત્ન કરો.

```
<html>
<body>
<table border="1">
<tr >
        <th>Name of player</th>
        <th>Points earned</th>
</tr>
<tr bgcolor="lightGreen">
        <td>Disha</td>
        <td>110</td>
</tr>
<tr >
        <td>Sweety</td>
        <td>100</td>
</tr>
<tr bgcolor="lightGreen">
        <td>Gayatri </td>
        <td>90</td>
</tr>
</table>
</body>
</html>
```

કોડ-લિસ્ટિંગ 4.7 : એક પછી એક આડી હરોળને આછા લીલા રંગે રંગવા માટેના HTML કોડ

કોડ-લિસ્ટિંગ 4.7નું પરિણામ આકૃતિ 4.11માં દર્શાવ્યા મુજબ દેખાશે.



આકૃતિ 4.11 : રંગબેરંગી બૅકગ્રાઉન્ડ ધરાવતી આડી હરોળવાળું કોષ્ટક

Valign ગુણધર્મ

Valign ગુણધર્મ અહીં હરોળના દરેક ખાનાના લખાણની ઊભી સમાંતર ગોઠવણી કરવાનું નક્કી કરે છે. આવું નીચે મુજબની સામાન્ય વાક્યરચના(syntax)નો ઉપયોગ કરીને કરી શકાય.

valign="position"

કોષ્ટક 4.5, valign ગુણધર્મની શક્ય કિંમતો દર્શાવે છે.

કિંમત	વર્ણન
top	લખાણને ખાનાની ટોચના ભાગ તરફ ગોઠવે છે.
middle	લખાણને ખાનાની મધ્યમાં ગોઠવે છે.
bottom	લખાણને ખાનાની નીચે તરફ ગોઠવે છે.
baseline	લખાણને એ રીતે ગોઠવે છે કે, જેથી દરેક ખાનામાંના લખાણની પ્રથમ લીટી એ જ આડી સમાંતર લીટીથી શરૂ થાય.

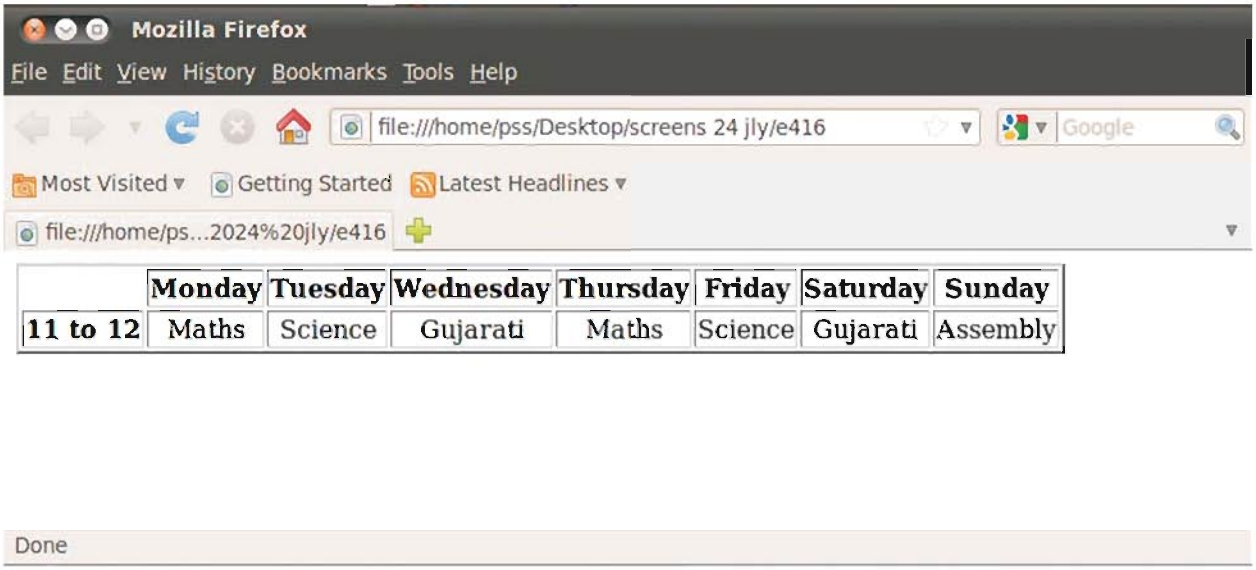
કોષ્ટક 4.5 : valign ગુણધર્મ માટેની શક્ય કિંમતો

<tr> ટેગનું ઉદાહરણ : <tr width="150". valign="bottom">. તમે પ્રયોગ કરતા હો તે HTML કોડમાં આ ટેગ લખીને બ્રાઉઝરમાં તેને ચકાસી જુઓ. કોડ-લિસ્ટિંગ 4.8માં આપેલ HTML કોડનો પ્રયત્ન કરી જુઓ.

```
<html>
<body>
<!-- -------------------------------------------------------------------------------- -->
        <table border="2">
                <tr>
                        <th> </th>
                        <th> Monday              </th>
                        <th> Tuesday             </th>
                        <th> Wednesday      </th>
                        <th> Thursday          </th>
                        <th> Friday                </th>
                        <th> Saturday           </th>
                        <th> Sunday              </th>
                </tr>
<!-- -------------------------------------------------------------------------------- -->
                <tr align="middle">
                        <th> 11 to 12             </th>
                        <td> Maths                 </td>
                <td> Science                 </td>
                <td> Gujarati               </td>
                <td> Maths                    </td>
                <td> Science                </td>
                <td> Gujarati              </td>
                <td> Assembly           </td>
        </tr>
</table >
<!-- -------------------------------------------------------------------------------- -->
</body>
</html>
```

કોડ-લિસ્ટિંગ 4.8 : સમયપત્રક તૈયાર કરવા માટેના HTML કોડ

કોડ-લિસ્ટિંગ 4.8માં દર્શાવેલ કોડ કોઈ એક વર્ગ માટે સાદા સમયપત્રકની બે આડી હરોળ બનાવે છે. જો તમે આ કોડને બ્રાઉઝરમાં જોશો તો તે આકૃતિ 4.12માં દર્શાવ્યા મુજબ દેખાશે.



આકૃતિ 4.12 : HTML કોડના ઉપયોગ દ્વારા બનતું સમયપત્રક

th અને tdના ઉપયોગ દ્વારા ખાનાઓની રજૂઆત

કોષ્ટકમાં દરેક ખાનું <td> અથવા <th> ટૅગ દ્વારા રજૂ થાય છે. એ રીતે, આ બંને ટૅગ કોઈ એક આડી હરોળને અનેક ખાનાંઓમાં વિભાજિત કરવાની સવલત પૂરી પાડે છે. જો મથાળું નક્કી કરવાનું હોય તો <th> ટૅગનો ઉપયોગ કરવામાં આવે છે, જ્યારે કોષ્ટકની વિગત રજૂ કરવી હોય તો <td> ટૅગનો ઉપયોગ કરવામાં આવે છે.

સામાન્ય રીતે, <th> ટૅગના લખાણને મોટેભાગે બોલ્ડ અક્ષરોમાં, ખાનાને આડા સમાંતર મધ્યમાં ગોઠવીને દર્શાવાય છે. જ્યારે <td> ટૅગમાં સમાવિષ્ટ લખાણ સામાન્ય કદના અક્ષરોમાં ખાનાને સમાંતર ડાબી બાજુ ગોઠવીને દર્શાવાય છે. <th> અને <td> બંને ટૅગમાં એક જ સરખા ગુણધર્મોનો ઉપયોગ થઈ શકે છે. આ ગુણધર્મની અસર માત્ર એક જ ખાના પર થશે. આ ગુણધર્મો દ્વારા ઊભી થનારી અસર આખા કોષ્ટક માટે કરેલી ગોઠવણી અથવા આની ઉપરના સ્તર જેમકે આડી હરોળ માટે કરેલી ગોઠવણીની અસરને પણ દબાવી દેશે. <th> અને <td> ગુણધર્મો દ્વારા થતી અસર એ જ આખરી ગણાશે.

સર્વ સામાન્ય ગુણધર્મો અને મૂળભૂત ઘટનાને લગતા ગુણધર્મો ઉપરાંત <th> અને <td> ટૅગ સાથે કોષ્ટક 4.6માં દર્શાવ્યા મુજબના ગુણધર્મો પણ હોઈ શકે.

ગુણધર્મ	વર્ણન
abbr	ખાનાની માહિતીનું ટૂંકું સ્વરૂપ પૂરું પાડે છે.
align	ખાનાની માહિતીને ગોઠવે છે.
bgcolor	ખાનામાં બૅકગ્રાઉન્ડ ઉમેરે છે.
char	ખાનાનાં લખાણને કોઈ ચોક્કસ અક્ષરની પ્રથમ ઉપસ્થિતિની આસપાસ ગોઠવે છે.
colspan	ખાનું કેટલાં ખાનાં સુધી વિસ્તરી શકે તે સંખ્યા દર્શાવે છે.
headers	ખાનાને સંબંધિત મથાળું દર્શાવે છે.
height	ખાનાની ઊંચાઈ નક્કી કરે છે.
nowrap	લખાણને ખાનાની અંદર જ આપમેળે નવી લાઇન ઉપર જતાં અટકાવે છે.
rowspan	ખાનું કેટલી આડી હરોળ સુધી વિસ્તરી શકે તે સંખ્યા દર્શાવે છે.
valign	ખાનામાં લખાણ ઊભી સમાંતર ગોઠવણી કરવાનું નક્કી કરે છે.
width	ખાનાની પહોળાઈ નક્કી કરે છે.

કોષ્ટક 4.6 : <td> અને <th>ના ગુણધર્મો

જેમ જેમ આપણે પ્રકરણમાં આગળ વધતાં જઈશું તેમ તેમ આપણે આ ગુણધર્મો વિશે વિગતવાર શીખીશું.

કોષ્ટકને નામ આપવું

કોષ્ટકને નામ આપવા માટે <caption> ટૅગનો ઉપયોગ કરવામાં આવે છે. જ્યારે તમે કોષ્ટકના નામ સાથે કોઈ ચોક્કસ કોષ્ટક દર્શાવો ત્યારે આ ટૅગની જરૂર પડે છે. કોષ્ટકનું નામ એ દર્શાવે છે કે, કોષ્ટક શેના માટે છે; પરિણામના હેતુ માટે, સમયપત્રક માટે અથવા ખોરાકની માત્રાનો આલેખ મોટાભાગના બ્રાઉઝર <caption> ટૅગના લખાણને કોષ્ટકના ઉપરના ભાગે મધ્યમાં ગોઠવીને દર્શાવે છે.

ટેબલ બનાવતા કોઈ પણ માન્ય HTML કોડની પ્રથમ લીટીની ઉપર નીચે મુજબની લીટી ઉમેરવાથી કોષ્ટકની ઉપર "This is our timetable" એવું કોષ્ટકનું નામ દર્શાવશે.

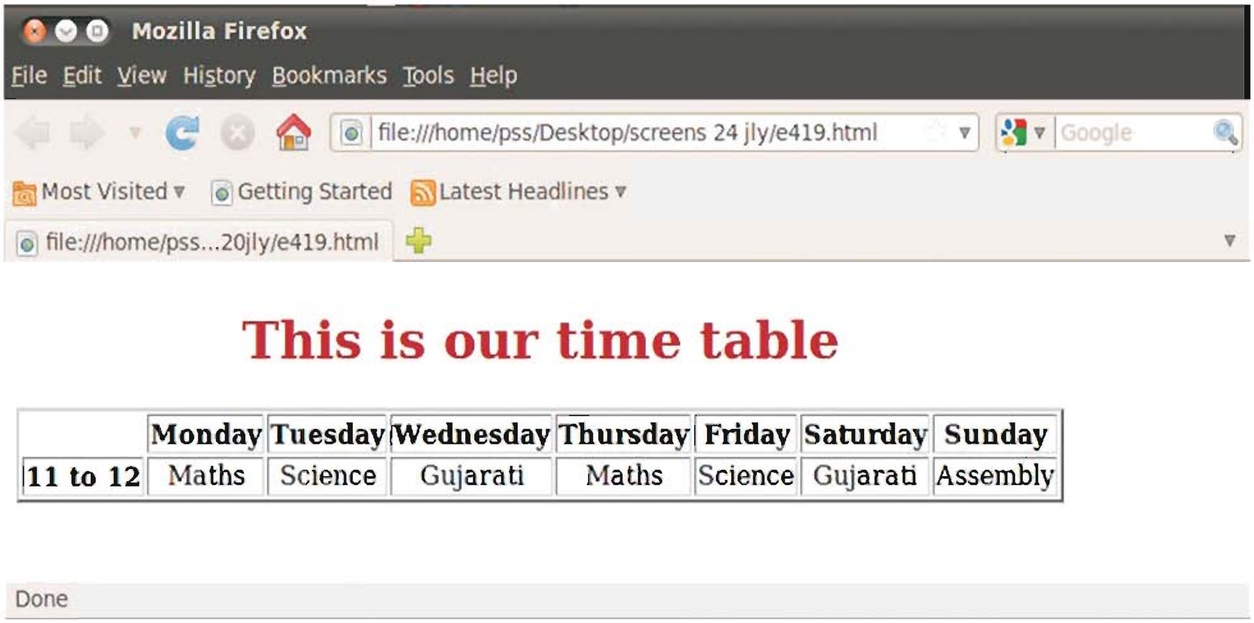
```
<caption> This is our timetable </caption>
```

કોડ-લિસ્ટિંગ 4.9 દર્શાવે છે કે, HTMLના ઉપયોગ દ્વારા કેવી રીતે સમયપત્રક બનાવી શકાય.

```
html>
<body>
      <table border="2">
<!-- -------------------------------------------------------------------------------- -->
      <caption> <h1> <font color="Brown" >
                        This is our time table
                    </font>   </h1> </caption>
<!-- -------------------------------------------------------------------------------- -->
      <tr>
            <th> </th>
            <th> Monday </th>
            <th> Tuesday </th>
            <th> Wednesday </th>
            <th> Thursday </th>
            <th> Friday </th>
            <th> Saturday </th>
            <th> Sunday </th>
      </tr>
<!-- -------------------------------------------------------------------------------- -->
      <tr align ="middle">
            <th> 11 to 12</th>
            <td> Maths</td>
            <td> Science </td>
            <td> Gujarati</td>
            <td> Maths</td>
            <td> Science </td>
            <td> Gujarati</td>
            <td> Assembly </td>
      </tr>
     </table>
</body>
</html>
```

કોડ-લિસ્ટિંગ 4.9 : કોષ્ટકનું નામ ઉમેરવું

કોઈ પણ કોષ્ટકને નામ આપવાથી કોઈ પણ ઉપયોગકર્તા માટે કોષ્ટક પહેલી નજરમાં સમજાઈ જાય છે. આકૃતિ 4.13 કોડ-લિસ્ટિંગ 4.9નું પરિણામ રજૂ કરે છે.



આકૃતિ 4.13 : નામ સાથેનું કોષ્ટક

બરાબર ધ્યાનથી અવલોકન કરો, કે આકૃતિ 4.13માં દર્શાવેલ સમયપત્રક અપૂર્ણ છે. વિદ્યાર્થીઓ તેને કદાચ પૂર્ણ કરીને તેનું પરિણામનું અવલોકન કરી શકે.

કોષ્ટકની અંદર કોષ્ટક

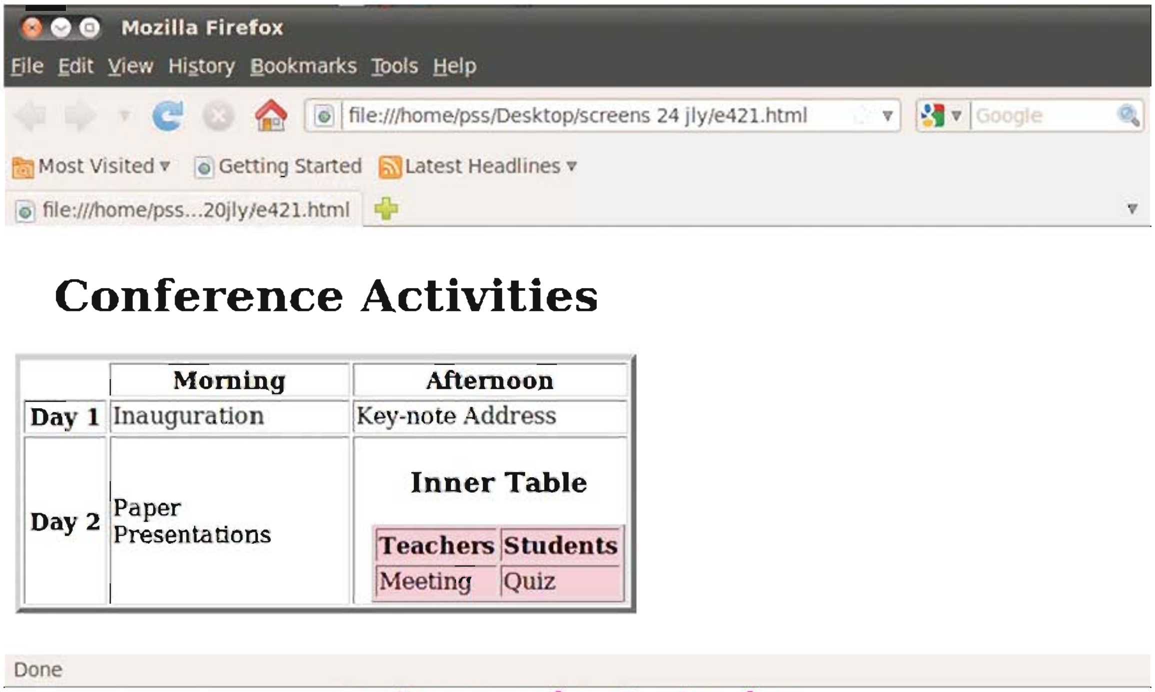
ઘણીવાર આપણને કોઈ એક કોષ્ટકની અંદર બીજું કોષ્ટક ઉમેરવાની જરૂર પડતી હોય છે. કોડ-લિસ્ટિંગ 4.10માં દર્શાવેલ ઉદાહરણ કોષ્ટકની અંદર કોષ્ટક દર્શાવે છે.

```
<html>
<body>
<!-- ---------------------------------------------------------------------------------- -->
<table border="4">
<caption> <h1>Conference Activities </h1></caption>
<tr>
        <th> </th>
        <th width ="40%"> Morning </th>
        <th> Afternoon </th>
</tr>
<tr>
        <th> Day 1</th>
<td> Inauguration </td>
<td> Key-note Address</td>
</tr>
<!-- ---------------------------------------------------------------------------------- -->
```

```
<tr>
        <th> Day 2 </th>
        <td > Paper Presentations</td>
        <td>
                <table border="1" bgcolor="pink" frame="box" align="right">
                        <caption> <h3> Inner Table </h3></caption>
                        <tr >
                                <th> Teachers</th>
                                <th> Students </th>
                        </tr>
                                <tr>
                                        <td> Meeting </td>
                                        <td> Quiz </td>
                                </tr>
                </table>
        </td>
</tr>
</table >
<!-- ------------------------------------------------------------------------------------ -->
</body>
</html>
```

કોડ-લિસ્ટિંગ 4.10 : કોષ્ટકની અંદર કોષ્ટક

આકૃતિ 4.14, કોડ-લિસ્ટિંગ 4.10નું પરિણામ દર્શાવે છે.



આકૃતિ 4.14 : કોષ્ટકની અંદર કોષ્ટક

તમે કદાચ એ નોંધ્યું જ હશે કે અંદરના કોષ્ટકને બેકગ્રાઉન્ડ રંગ અને કિનારી છે. અંદરના તેમજ બહારના એમ બંને કોષ્ટકને તેમનાં નામ સાથે દર્શાવાયાં છે.

કોષ્ટકનું ઉદાહરણ

આકૃતિ 4.15 કોષ્ટકનું ઉદાહરણ દર્શાવે છે, તમે ક્યારેક તો જોયું જ હશે અને શક્યતઃ વાપર્યું પણ હશે. ભાવસહિત તમારી પસંદગીની વાનગીઓ ધરાવતું કોઈ પંજાબી રેસ્ટોરન્ટના મેનૂ વિશે છે.

Star Restaurant

Starters

Item	Price
Paneer Tikka Dry	100.00
Panner kabab	100.00

Roti and Bread

Item	Price
Plain Roti	10.00
Tandoori Roti	20.00

Curries

Item	Price
Paneer Bhurji	100.00
Mixed Veg	100.00

Drinks

Item	Price
Tea	20.00
Coffee	20.00
Koko juicy	25.00
Mengo delight	25.00

Desserts

Item	Price
Gulab Jamun	30.00
Rasgulla	35.00
Pudding	40.00
Ice-cream	50.00

Water

Item	Price
Regular	---
Mineral	15
Sparkling	28

The prices are in Indian rupees.
Customers have to pay extra taxes.
Please wait for 20 minutes after giving order.

આકૃતિ 4.15 : રેસ્ટોરન્ટના મેનૂનું કાર્ડ

કોડ-લિસ્ટિંગ 4.11, આકૃતિ 4.15માં દર્શાવાયેલ રેસ્ટોરન્ટ મેનૂ કાર્ડ તૈયાર કરવા માટેના આંશિક HTML કોડ દર્શાવે છે.

```
<html>
      <head> <title> Star Restaurant Menu </title> </head>
      <body>
      <center>
<table width="75%"  border="1"  bordercolor="#000000" bordercolordark="#000000"
cellspacing="0">
<tr align="center">
      <th align="center" colspan="6"> <h1> Star Restaurant </h1></th>
</tr>
<tr>
      <td colspan="2" align="center"> <h2> Starters  </h2></td>
      <td colspan="2" align="center"> <h2>Roti and Bread </h2></td>
      <td colspan="2" align="center"> <h2>Curries </h2></td>
```

```
</tr>
<tr>
    <td align="center"> <strong> <font color = "Blue" > Item  </font>  </strong> </td>
    <td align="center"> <strong> <font color = "Blue" > Price </font>  </strong> </td>
    <td align="center"> <strong> <font color = "Blue" > Item  </font>  </strong> </td>
    <td align="center"> <strong> <font color = "Blue" > Price </font>  </strong> </td>
    <td align="center"> <strong> <font color = "Blue" > Item  </font>  </strong> </td>
    <td align="center"> <strong> <font color = "Blue" > Price </font>  </strong> </td>
</tr>
<tr bgcolor="#CCCCCC">
    <td align="center"> <strong> Paneer Tikka Dry </strong> </td>
    <td align="center"> 100.00 </td>
    <td align="center"> <strong> Plain Roti </strong> </td>
    <td align="center"> 10.00 </td>
    <td align="center"> <strong> Paneer Bhurji </strong> </td>
    <td align="center"> 100.00 </td>
</tr>
<tr>
    <td align="center"> <strong> Paneer Kabab </strong> </td>
    <td align="center"> 100.00 </td>
    <td align="center"> <strong> Tandoori Roti </strong> </td>
    <td align="center"> 20.00 </td>
    <td align="center"> <strong> Mixed Veg </strong> </td>
    <td align="center"> 100.00 </td>
</tr>
<!-- –Remaining part of the table may be completed using similar code.... -->
<tr align="center">
    <td align="center" colspan="6"> The prices are in Indian rupees. <br>
    Customers have to pay extra taxes. <br>
    Please wait for 20 minutes after giving the order.
</td>
</tr>
</table>
</center>
</body>
</html>
```

કોડ-લિસ્ટિંગ 4.11 : HTML કોષ્ટકના ઉપયોગ દ્વારા રેસ્ટોરન્ટ મેનૂ કાર્ડ દર્શાવવું

અહીં એ નોંધ લો કે, કોડ-લિસ્ટિંગ 4.11માં આપેલ HTML કોડ આંશિક કોષ્ટક દર્શાવે છે. આવા જ વધુ HTML કોડ ઉમેરીને વિદ્યાર્થીઓ કોષ્ટકનો બાકીનો ભાગ પૂરો કરી શકે.

HTMLમાં ફ્રેમ

HTML દસ્તાવેજમાં ફ્રેમનો ઉપયોગ અનેક વેબપેઇજને ભેગાં કરીને એક જ વેબપેઇજ તરીકે દર્શાવવા થાય છે. ફ્રેમ, બ્રાઉઝરની વિન્ડોને ઘણા ભાગોમાં અથવા પેટા-વિન્ડોમાં વિભાજિત કરે છે જે દરેક એક અલગ વેબપેઇજ ધરાવે છે. બ્રાઉઝર વિન્ડોને અનેક ફ્રેમમાં વિભાજિત કરી શકાય છે, જેથી તમે જુદા જુદા HTML કોડને સ્વતંત્ર રીતે સંભાળી શકો અને તેને લઈ આવવાનું (loading) અને ફરી લઈ આવવાનું(reloading) કાર્ય સંભાળી શકો. બ્રાઉઝર વિન્ડોમાં ફ્રેમના સમૂહને ''ફ્રેમસેટ'' (frameset) તરીકે ઓળખવામાં આવે છે.

અહીં એ નોંધ લેવી જોઈએ કે, કેટલાક બ્રાઉઝર ફ્રેમસેટને સમર્થન આપતાં નથી. ફ્રેમસેટ દસ્તાવેજ તૈયાર કરવા, સૌપ્રથમ આપણે <frameset> એલિમેન્ટ બનાવવો પડે, જેને વિવિધ ફ્રેમના સંગ્રાહક તરીકે ગણવામાં આવશે. ફ્રેમસેટ બ્રાઉઝર વિન્ડોનો વિભાગ વ્યાખ્યાયિત કરે છે. ફ્રેમસેટની અંદર દરેક ફ્રેમ <frame> અને </frame> ટૅગ વડે દર્શાવવામાં આવે છે.

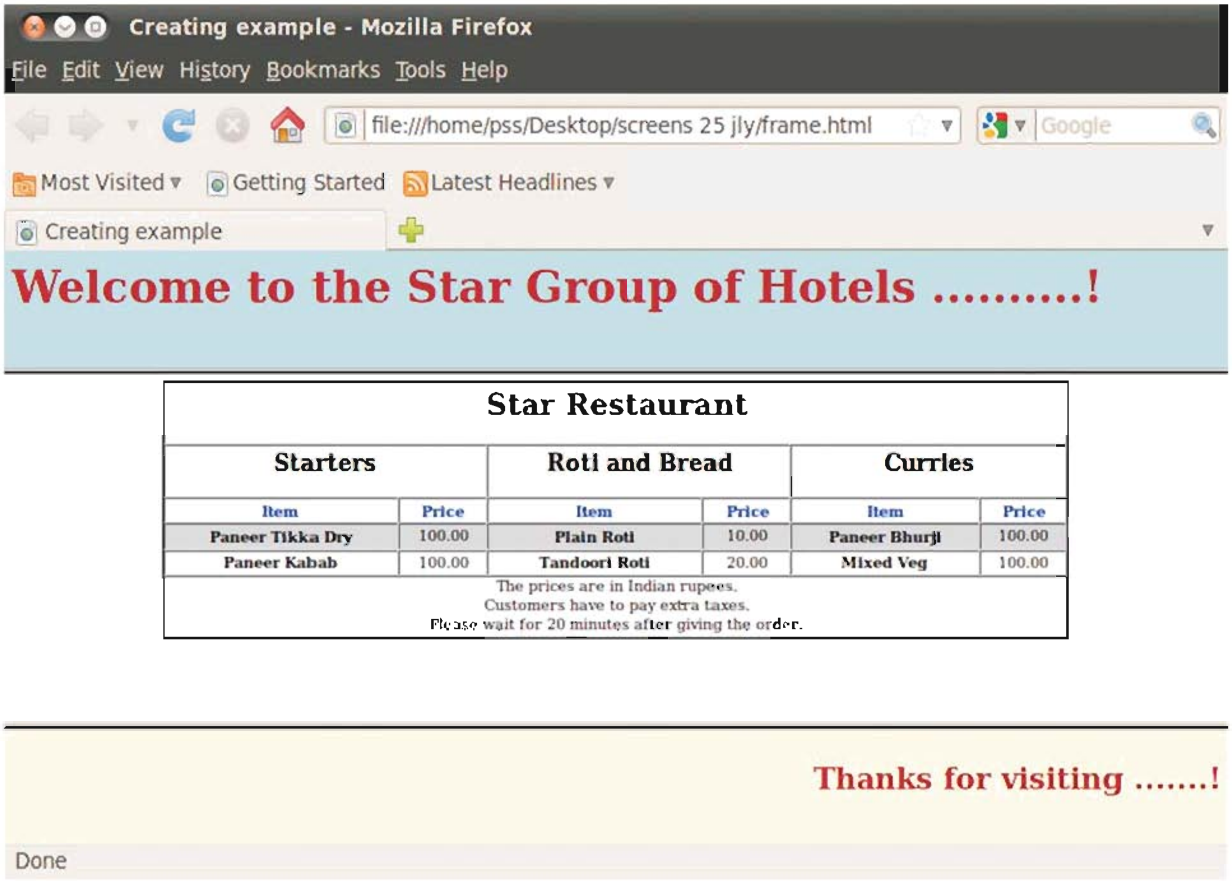
ફ્રેમસેટની અંદર, તમે <noframes> એલિમેન્ટ ઉમેરી શકો, જેથી જ્યારે ઉપયોગમાં લેવાતું બ્રાઉઝર ફ્રેમ સવલતને સમર્થન ન આપતું હોય ત્યારે તે ઉપયોગકર્તા માટે વૈકલ્પિક સંદેશ પૂરો પાડે છે. કોડ-લિસ્ટિંગ 4.12માં આપેલ ઉદાહરણ જુઓ.

```
<html>
<head>
<title>Creating example</title>
</head>
     <frameset rows="20%,60%,20%">
     <frame src="top.html" />
     <frame src="main.html" />
     <frame src="bottom.html" />
     <noframes>
          <body>
               Your browser does not support frames.
          </body>
     </noframes>
     </frameset>
</html>
```

કોડ લિસ્ટિંગ 4.12 : ફ્રેમ બનાવવા માટેના HTML કોડ

કોડ-લિસ્ટિંગ 4.12માં દર્શાવેલ HTML કોડ લખો અને તેને frame.html તરીકે સંગ્રહ કરો. જરૂરી ફાઈલો બનાવો જેવી કે, top.html, main.html અને bottom.html. આ ફાઈલોમાં તમારું મનપસંદ લખાણ ઉમેરો. એકવાર તમે આ ફાઈલો બનાવી દો એ પછી આપણે તેને બ્રાઉઝરમાં ચકાસી શકીએ. આપણું પરિણામ આકૃતિ 4.16માં દર્શાવ્યા મુજબનું દેખાશે. અહીં એ ખાસ નોંધ લો કે, તમને જે પરિણામ જોવા મળશે તે તમે ફાઈલમાં શું લખાણ ઉમેર્યું છે તેના આધારે જુદું પડી શકે.

આકૃતિ 4.16માંથી એ અવલોકન કરવામાં આવ્યું કે, બ્રાઉઝર વિન્ડો ત્રણ ભાગમાં વિભાજિત કરવામાં આવી છે. વિન્ડોના ઉપરનો ભાગ વાદળી રંગે રંગવામાં આવે છે અને તે top.html નામની HTML ફાઈલનું લખાણ દર્શાવે છે. મધ્ય ભાગમાં રેસ્ટોરન્ટ મેનૂ માટે સાદું કોષ્ટક છાપે છે. મધ્ય ભાગ main.html નામની HTML ફાઈલ મારફત દર્શાવાય છે. આછા પીળા રંગે રંગાયેલો છેક નીચેનો ભાગ bottom.html નામની HTML ફાઈલ મારફતે દર્શાવાય છે.



આકૃતિ 4.16 : HTMLમાં ફ્રેમ

અહીં એ નોંધો કે, બ્રાઉઝર વિન્ડોને ઊભા ભાગમાં વહેંચવાનું પણ શક્ય છે. બ્રાઉઝર વિન્ડોને ઊભા ભાગમાં વહેંચવા માટે આપણે નીચે મુજબ ઊભી હરોળ (cols) સાથે frameset એલિમેન્ટનો ઉપયોગ કરી શકીએ.

<frameset cols="25%,*,25%">

અહીં, આપણે બ્રાઉઝર વિન્ડોને ત્રણ ઊભા ભાગમાં વહેંચવાનો પ્રયત્ન કરેલ છે. કોડ-લિસ્ટિંગ 4.12માં <frameset rows="20%,60%,20%"> કોડને બદલીને <frameset cols="25%,*,25%"> કરો અને ફરીથી તેનું પરિણામ નિહાળવાનો પ્રયત્ન કરો.

સારાંશ

આ પ્રકરણમાં, આપણે HTML માં યાદી અને કોષ્ટકો તૈયાર કરવાની વિવિધ રીતો જોઈ. આપણે જુદા જુદા ગુણધર્મો સહિત ક્રમબદ્ધ યાદી ઉપરાંત અક્રમિક યાદી પણ જોઈ. યાદીની અંદર બીજી યાદી સમાવવાની વ્યવસ્થા પણ આ પ્રકરણમાં સામેલ કરવામાં આવી. સાદાં કોષ્ટકો ઉપરાંત કોષ્ટકની અંદર કોષ્ટક બનાવવાનું પણ આપણે શીખ્યા. આ પ્રકરણમાં, આખા કોષ્ટક પર અસર કરતા ગુણધર્મો ઉપરાંત અલગ કોષ્ટકની આડી હરોળ અને અલગ ખાનાંઓ માટેના ગુણધર્મો પણ જોયા.

સ્વાધ્યાય

1. યોગ્ય ઉદાહરણ આપીને સમજાવો કે HTML દસ્તાવેજમાં યાદી કેવી રીતે વ્યાખ્યાયિત કરવામાં આવે છે.
2. HTMLમાં વિવિધ પ્રકારની યાદી વર્ણવો.
3. HTMLમાં યાદીની અંદર યાદી ઉપર ટૂંક નોંધ લખો.
4. HTMLમાં કોષ્ટકના વ્યવસ્થાપન બાબતે ટૂંક નોંધ લખો.

5. નીચેના પ્રશ્નો માટે આપેલા વિકલ્પોમાંથી યોગ્ય વિકલ્પ પસંદ કરી જવાબ આપો :

(1) HTMLમાં દ્વારા શું નક્કી થાય છે ?

(a) સાદી યાદી
(b) ક્રમબદ્ધ યાદી
(c) અક્રમિક યાદી
(d) સાદું કોષ્ટક

(2) HTMLમાં દ્વારા નીચેનામાંથી કઈ બાબત નક્કી થાય છે ?

(a) સાદી યાદી
(b) ક્રમબદ્ધ યાદી
(c) અક્રમિક યાદી
(d) સાદું કોષ્ટક

(3) યાદીમાંની વિગતોને નીચેનામાંથી કઈ ટૅગ ઓળખી કાઢે છે ?

(a) અને
(b) <items> અને </items>
(c) <object> અને </object>
(d) <table> અને </table>

(4) બુલેટ્સ વગર યાદી રજૂ કરવા માટે નીચેનામાંથી કઈ ટૅગની જોડી ઉપયોગમાં લેવાય છે ?

(a) અને
(b) અને
(c) <dt> અને </dt>
(d) <pt> અને </pt>

(5) કોષ્ટકની આડી હરોળ વ્યાખ્યાયિત કરવા માટે નીચેનામાંથી કઈ ટૅગની જોડી ઉપયોગમાં લઈ શકાય ?

(a) <tr> અને </tr>
(b) <td> અને </td>
(c) <col> અને </col>
(d) <row> અને </row>

(6) જ્યારે કોઈ ખાનું એક કરતાં વધુ આડી હરોળમાં વિસ્તરણ પામતું હોય ત્યારે નીચેનામાંથી કયો ગુણધર્મ ઉપયોગમાં લેવાય છે ?

(a) colspan
(b) rowspan
(c) span
(d) scope

(7) બ્રાઉઝર વિન્ડોને અનેક ભાગોમાં વિભાજિત કરવા નીચેનામાંથી શાનો ઉપયોગ કરી શકાય ?

(a) Frameset
(b) Elements
(c) Layouts
(d) Design

(8) જો બ્રાઉઝર frameને સમર્થન ન આપતું હોય તો, તેના બદલે વૈકલ્પિક લખાણ દર્શાવવા નીચેનામાંથી કયું ઉપયોગમાં લઈ શકાય ?

(a) Noframes
(b) Yesframe
(c) Falseframe
(d) Trueframe

(9) HTMLની ક્રમબદ્ધ યાદીમાં નીચેનામાંથી કયું બદલી શકાય છે ?

(a) મુદ્દાઓનો ક્રમ
(b) શરૂઆતનો ક્રમ
(c) ક્રમ માટે સંખ્યાચિહ્નનો પ્રકાર
(d) આપેલા તમામ

(10) HTML કોષ્ટકમાં નીચેનામાંથી કઈ વૈકલ્પિક બાબત (entity) છે ?

(a) કોષ્ટકનું નામ (Caption)
(b) કોષ્ટકનું મથાળું (Heading)
(c) કોષ્ટકની ઊભી હરોળનું ગઠબંધન
(d) આપેલા તમામ

પ્રાયોગિક સ્વાધ્યાય

1. તમારા મિત્રોનાં નામ અને શાળાની પરીક્ષામાં તેમણે મેળવેલ કુલ ગુણ (ટકાવારીમાં) દર્શાવતું એક કોષ્ટક તૈયાર કરતા HTML કોડ લખો.
2. તમે તમારા માતા-પિતાને આવક અને ઘરેલુ ખર્ચાની નોંધ રાખવા ડાયરીમાં હિસાબો લખતા જોયા હશે. ઘરેલુ કાર્ય માટે થતા ખર્ચા અને આવકો દર્શાવતું કોષ્ટક તૈયાર કરવા HTML કોડ લખો.
3. આ પ્રકરણના પ્રથમ વિભાગમાં દર્શાવાયા મુજબ ગુણપત્રક (marksheet) તૈયાર કરવા HTML કોડ લખો.
4. તમારા વર્ગનું સમયપત્રક તૈયાર કરવા માટેના HTML કોડ લખો. આ પ્રકરણમાં દર્શાવેલ સમયપત્રક નિહાળી યોજના ઘડો.
5. કોઈ એક મલ્ટિપ્લેક્સ સિનેમા માટે તેમાં ઉપલબ્ધ ચલચિત્રોની જાહેરાત તેમજ પ્રદર્શન થાય તે માટેનું કોષ્ટક તૈયાર કરવા માટેના HTML કોડ લખો. ધારી લો કે, તમારા શહેરના મલ્ટિપ્લેક્સ થિયેટરમાં ત્રણ સ્ક્રીન છે, અને સવારે 9 વાગે, બપોરે 1 વાગે, સાંજે 5 વાગે અને રાત્રે 9 વાગે એમ ચાર શો છે. ત્રીજા સ્ક્રીન પર મોડી રાત્રિનો (late night) શો ત્રણ જુદા જુદા પ્રાદેશિક ભાષાના ચલચિત્રો જેમાં એક ગુજરાતી, બીજું મરાઠી અને ત્રીજું તેલુગુ ઉપલબ્ધ બને છે. કોષ્ટકનો એક નમૂનારૂપ દેખાવ નીચે મુજબ છે. દરેક ખાનામાં તમારી પસંદગીના ચલચિત્રોના નામ ઉમેરો. અંદરના કોષ્ટકને "Regional Movies" એવું નામ પણ આપો.

Timings→	9:00 am	1:00 pm	5:00 pm	9:00 pm
Screen 1				
Screen 2				
Screen 3				Regional Movies

6. તમારે જે ચીજવસ્તુ ખરીદવી હોય તે વસ્તુઓની યાદી તૈયાર કરવા માટેના HTML કોડ લખો. ચીજવસ્તુઓને જુદા જુદા જૂથમાં વિભાજિત કરો જેમ કે, પુસ્તકો, શાકભાજી અને ખોરાકની ચીજો. નીચેનું નમૂનારૂપ ફ્રેમવર્ક જુઓ, જે દર્શાવે છે કે યાદી કેવી હોવી જોઈએ.

List of items to be purchased

I. Books
   a. Textbook of science
   b. Practice book of maths
   c. Notebook

II. Vegetables
   a. Tomatoes
   b. Spinach
   c. Peas

III. Food items
   a. Chocolates
   b. Butter
   c. Bread

કેલ્સીનો પરિચય

કેલ્સી (calc) એ ઓપનઑફિસ સ્યૂટનું એક વિજાણુ વિસ્તારપત્રક (ઇલેક્ટ્રૉનિક સ્પ્રેડશીટ) પૅકેજ છે. સ્પ્રેડશીટ (Spreadsheet) એ પ્રકારનું સૉફ્ટવેર પૅકેજ છે જે દસ્તાવેજમાં ઝડપી અને સચોટ ગણતરીઓ કરવા માટે તેમજ દસ્તાવેજની માહિતીને આકર્ષક રીતે ગોઠવવા માટે (ફૉર્મેટ કરવા) માટે વપરાય છે. (માહિતીને યોગ્ય તેમજ આકર્ષક રીતે ગોઠવવાની ક્રિયાને ફૉર્મેટિંગ કહેવામાં આવે છે). આ સ્પ્રેડશીટ પૅકેજનો ઉપયોગ નાણાંકીય અને હિસાબી દસ્તાવેજનાં વ્યવસ્થાપનમાં, માહિતીપત્રકો તૈયાર કરવામાં, મોકલેલા માલની કિંમત સાથેની યાદી (ઇન્વૉઇસ) બનાવવામાં, વૈજ્ઞાનિક અને આંકડાકીય સંશોધનો પરથી માહિતી વિશ્લેષણ કરવામાં તેમજ માહિતી ઉપર વિવિધ પ્રકારની ગણતરીઓ કરવામાં થાય છે. કોઈ પણ સ્પ્રેડશીટ પ્રોગ્રામ ડેટાનો સંગ્રહ કરવા, તેના ઉપર ગણતરીઓ કરવા તેમજ માહિતીને રેખાત્મક રીતે (ગ્રાફિકલ) રજૂઆત કરવા સમક્ષ હોય છે.

સ્પ્રેડશીટ પૅકેજ ડેટાને આડી હરોળ (row) અને ઊભી હરોળ(column)નાં સ્વરૂપમાં દાખલ કરવાની પરવાનગી આપે છે. તમે યાદ કરો : ગણિતનો અભ્યાસ કરવા માટે કદાચ તમે તમારા બાળપણમાં ખાસ પ્રકારની નોટબુક કે જેમાં આડી હરોળ અને ઊભી હરોળ વડે ચિહ્નિત સેલ હોય તેનો ઉપયોગ કર્યો હશે. સ્પ્રેડશીટ એ ડેટાનું વિશ્લેષણ તેમજ તેના ઉપર ગણતરીઓ કરવા માટેની કમ્પ્યૂટર સ્ક્રીન ઉપરની આડી હરોળ અને ઊભી હરોળ વડે બનેલી એક લાંબી શીટ (વિસ્તાર) છે. બીજી રીતે કહીએ તો સ્પ્રેડશીટ એ એક ગ્રીડ (જાળી જેવું બંધારણ) છે કે જે ક્રિયા-પ્રતિક્રિયાની રીતે (interactively) ડેટાનું રો અને કૉલમની રીતે સંચાલન તેમજ આયોજન કરે છે. આ કામગીરીની સરળતા માટે સ્પ્રેડશીટ પૅકેજ ડેટા અને ફૉર્મેટિંગની સગવડ ઉપરાંત સૂત્રો દાખલ કરવાની પરવાનગી પણ આપે છે. કોઈ નિર્ણય લેવા માટે ઉપયોગકર્તા સ્પ્રેડશીટનાં પાનાંમાં ક્રિયા-પ્રતિક્રિયાની રીતે ડેટા દાખલ કરી શકે, ડેટાને ફૉર્મેટ કરી શકે તેમજ ડેટા ઉપર ગણતરી / વિશ્લેષણ કરી શકે છે. ઉપયોગકર્તાની માહિતી અને સૂત્રો ઉપરાંત સર્વસામાન્ય ગાણિતિક, નાણાંકીય, આંકડાકીય અને તાર્કિક કાર્યો ખૂબ જ અત્યાધુનિક રીતે કરવા માટેનાં સૂત્રો / વિધેયો સ્પ્રેડશીટ પૅકેજમાં જ સમાવિષ્ટ કરેલાં હોય છે. સ્પ્રેડશીટ પૅકેજની આ પ્રકારની અનેક ક્ષમતાઓને કારણે સંગઠિત (structured) ડેટા તૈયાર કરવા તેમજ તેના ઉપર પ્રક્રિયા કરવા માટે તે એક વિશ્વવ્યાપી પ્રોગ્રામ તરીકે વપરાય છે. Libre Office Calc [www.libreoffice.org] પ્રોગ્રામ એ calc જેવો લાગે છે, અને તે નિઃશુલ્ક અને ઓપનસોર્સ સ્પ્રેડશીટ પૅકેજ પણ છે.

સ્પ્રેડશીટની પેલી પાર (Beyond a spreadsheet)

જેમ જેમ સ્પ્રેડશીટનું કદ મોટું બનતું જાય છે તેમ તેમ તેના સંચાલનનું કાર્ય વધારે કઠિન બને છે. સ્પ્રેડશીટનાં વધતા કદનાં નિયંત્રણ માટે વર્કબુક(workbook)નો એક ખ્યાલ અમલમાં મૂકવામાં આવ્યો. વર્કબુકનો મુખ્ય હેતુ સંખ્યાબંધ સ્પ્રેડશીટનું સારી રીતે સંચાલન કરવાનો છે. વર્કબુક ઉપરાંત હાલમાં અનેક ઑનલાઇન સ્પ્રેડશીટ પણ લોકપ્રિય બની રહી છે.

કેટલાંક લાક્ષણિક વિનિયોગોમાં સ્પ્રેડશીટ પૅકેજનો ઉપયોગ (Typical Applications of Spreadsheet Packages)

માહિતી વિશ્લેષણ તેમજ એકાઉન્ટિંગનાં કાર્યો માટે સ્પ્રેડશીટ પૅકેજનો વ્યાપકપણે ઉપયોગ થાય છે. સામાન્ય રીતે સ્પ્રેડશીટ પૅકેજ દ્વારા કરવામાં આવતાં કેટલાંક વિશિષ્ટ કાર્યોની યાદી કોષ્ટક 5.1માં આપવામાં આવેલી છે.

કાર્ય	વર્ણન
સરવૈયું (બેલેન્સ શીટ)	નાણાંકીય સ્થિતિનાં પત્રકો અને લાક્ષણિક એકાઉન્ટ ફૉર્મેટમાં સારાંશનાં પત્રકો
પરિણામ વિશ્લેષણ અને મેરિટની યાદી બનાવવી	પરીક્ષાઓ જેવી કેટલીક પ્રવૃત્તિઓનાં પરિણામની ગણતરી, સૉર્ટિંગ (ક્રમબદ્ધ ગોઠવણી) તેમજ ફિલ્ટરીંગ (કોઈ ચોક્કસ ધોરણો આધારિત પરિણામ)
આંકડાકીય ડેટા વિશ્લેષણ	સૉર્ટિંગ (ક્રમમાં ગોઠવણી), ડેટાની વ્યવસ્થિત ગોઠવણી અને આંકડાકીય પદ્ધતિઓનો અમલ જેમ કે મધ્યક (mean), મધ્યસ્થ (median) અને સંભાવના (probability) વગેરેની ગણતરી કરવી
નાણાંકીય કાર્યો	લોનના હપ્તાની ગણતરી, વ્યાજની ગણતરી વગેરે જેવાં નાણાંકીય કાર્યો
અંગત કાર્યો	વજનનું વ્યક્તિગત રીતે નિયંત્રણ (દેખરેખ), વિવિધ પ્રવૃત્તિઓ, વસ્તુઓ અને ઘટનાની યાદીનું સંચાલન જેમ કે કોઈ મિજબાની અથવા આગામી ઘટના માટે મહેમાનોની યાદી બનાવવી.

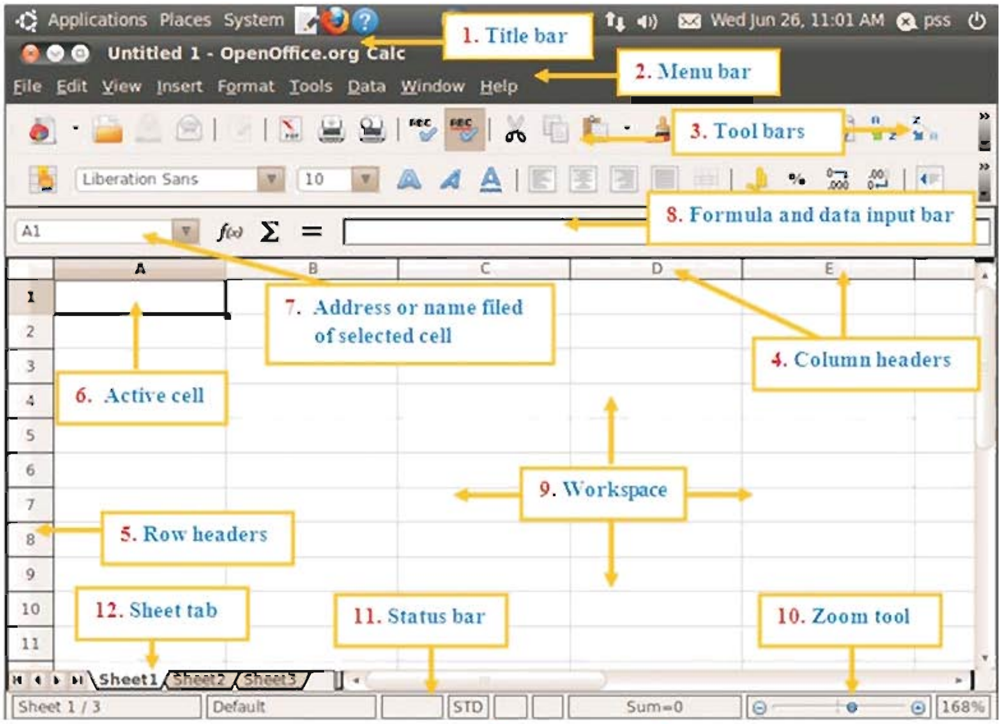
કોષ્ટક 5.1 : સ્પ્રેડશીટના કેટલાક પ્રચલિત વિનિયોગો

કેલ્સી સાથે પ્રારંભ (Getting started with Calc)

સૌપ્રથમ તમારે એ જાણવું જોઈએ કે કેલ્સીને કઈ રીતે ખોલી શકાય. આ માટે તમે આકૃતિ 5.1માં દર્શાવ્યા પ્રમાણેના આદેશોની શ્રેણીને અનુસરો : **Applications → Office → OpenOffice.org Spreadsheet**

આકૃતિ 5.1 : કેલ્સી સાથે શરૂઆત

આ આદેશ એક વિનિયોગ દસ્તાવેજ ખોલશે જે સ્પ્રેડશીટ કહેવાય છે. સ્પ્રેડશીટ અનેક શીટની બનેલી હોય છે; જે વર્કશીટ (worksheet) પણ કહેવાય છે. પૂર્વનિર્ધારિત રીતે નવી સ્પ્રેડશીટમાં ત્રણ વર્કશીટ હોય છે પણ આપણી જરૂરિયાત પ્રમાણે તેમાં સંખ્યાબંધ નવી વર્કશીટ ઉમેરી પણ શકાય તેમજ દૂર પણ કરી શકાય છે. કોઈ પણ કેલ્સી સ્પ્રેડશીટ દસ્તાવેજમાં મહત્તમ 256 વર્કશીટ હોઈ શકે.



આકૃતિ 5.2 : એક ખાલી વર્કશીટ

જ્યારે તમે કેલ્સીમાં એક નવો દસ્તાવેજ ખોલો છો ત્યારે આકૃતિ 5.2માં દર્શાવ્યા પ્રમાણે એક ખાલી સ્પ્રેડશીટ તમે સ્ક્રીન ઉપર જોશો. તેમાં નંબર ટેગ સાથે અલગ-અલગ ઘટક દર્શાવેલ છે. ચાલો, હવે આપણે દરેક ઘટકની વિગતવાર ચર્ચા કરીએ.

ટાઈટલ બાર (Title bar)

ટાઈટલ બાર સૌથી ઉપર સ્થિત થયેલ હોય છે. તે હાલમાં ખુલ્લી સ્પ્રેડશીટનું નામ (શીર્ષક) બતાવે છે. જો તમે હમણાં નવી જ ફાઈલ ખોલી છે અને સ્પ્રેડશીટને નામ આપેલું નથી તો તે એક નામ વગરની સ્પ્રેડશીટ છે. તે Untitled Xથી દર્શાવવામાં આવે છે, જ્યાં X એ એક સંખ્યા છે. જ્યારે કોઈ સ્પ્રેડશીટ તમે પ્રથમ વખત સંગ્રહ કરો છો ત્યારે તમારી પસંદગીનું નામ તમને પૂછવામાં આવે છે. તે તમને હાલમાં કઈ વર્કબુકમાં કામ કરી રહ્યા છો તે જણાવશે. આકૃતિ 5.3માં ટાઈટલ બાર દર્શાવેલ છે.

આકૃતિ 5.3 : ટાઈટલ બાર

મેનૂ બાર (Menu bar)

મેનૂ બારમાં વિવિધ કાર્યો કરવા માટેના મેનૂ સાથે આદેશો હોય છે. દરેક મેનૂ આઈટમ એક અલગ સબમેનૂ (submenu) રજૂ કરે છે. આ સબમેનૂને પુલડાઉન મેનૂ (pull down menu) પણ કહેવામાં આવે છે. મેનૂ જોવા માટે માઉસ કર્સરને ફક્ત તેના ઉપર મૂકો અને એક વાર ક્લિક કરો આથી સબમેનૂ આપણને સ્ક્રીન ઉપર દેખાય છે. આકૃતિ 5.4માં એક લાક્ષણિક મેનૂ બાર દર્શાવેલ છે.

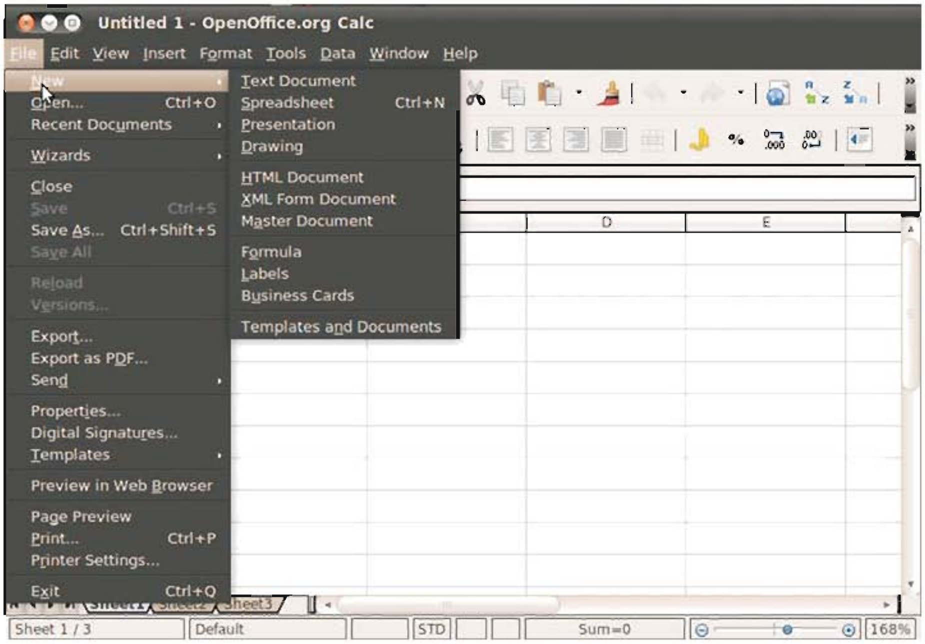
આકૃતિ 5.4 : મેનૂ બાર

મુખ્ય મેનૂના દરેક આદેશોની તેનાં ટૂંકા વર્ણન સાથેની યાદી કોષ્ટક 5.2માં આપેલી છે.

મેનૂ	વર્ણન
File	દસ્તાવેજમાં કામ કરવા માટેના આદેશો જેવા કે નવા દસ્તાવેજ બનાવવા, ફાઈલનો સંગ્રહ કરવો, પ્રિન્ટિંગ કરવું, પ્રિન્ટ કરતાં પહેલાં સ્ક્રીન ઉપર તેનો દેખાવ જોવો (પ્રિન્ટ પ્રિવ્યૂ) વગેરે.
Edit	દસ્તાવેજમાં સુધારા-વધારા કરવા માટેના આદેશો જેવા કે copy, paste, find & replace વગેરે.
View	વપરાશકર્તાના સેતુના ઘટકો ઉમેરવા અથવા દૂર કરવા, પાનું પૂરું થાય તેની નિશાની જોવી (પેઇજ બ્રેક પ્રિવ્યૂ) વગેરે.
Insert	આડી હાર, ઊભી હાર, વર્કશીટ, ઘટકો અને ઑબ્જેક્ટ ઉમેરવા વગેરે.
Format	સેલ ફૉર્મેટ કરવા, ઘટકોનો સમૂહ બનાવવો, સૉર્ટિંગ, શરતી ફૉર્મેટિંગ વગેરે.
Tools	વધારાનાં ટૂલ્સ જેવાં કે સ્પેલિંગ ચેક કરવા માટે, દસ્તાવેજનું રક્ષણ કરવા માટે, સૂત્રો તેમજ ભૂલો સુધારવા માટે વગેરે.
Data	ડેટા પર પ્રક્રિયા કરવી, ડેટા સૉર્ટ કરવો, શરત પ્રમાણે માત્ર જરૂરી ડેટા રજૂ કરવો (ફિલ્ટર) વગેરે.
Window	નવી વિન્ડો ખોલવી, સેલને સ્થાયી સ્વરૂપ આપવું તેમજ OpenOffice.orgના ખુલ્લા દસ્તાવેજોની યાદી વગેરે.
Help	વિષય તેમજ વિનિયોગ બાબતની માહિતી અને સૉફ્ટવેરની આવૃત્તિઓ વગેરે બાબતની મદદ

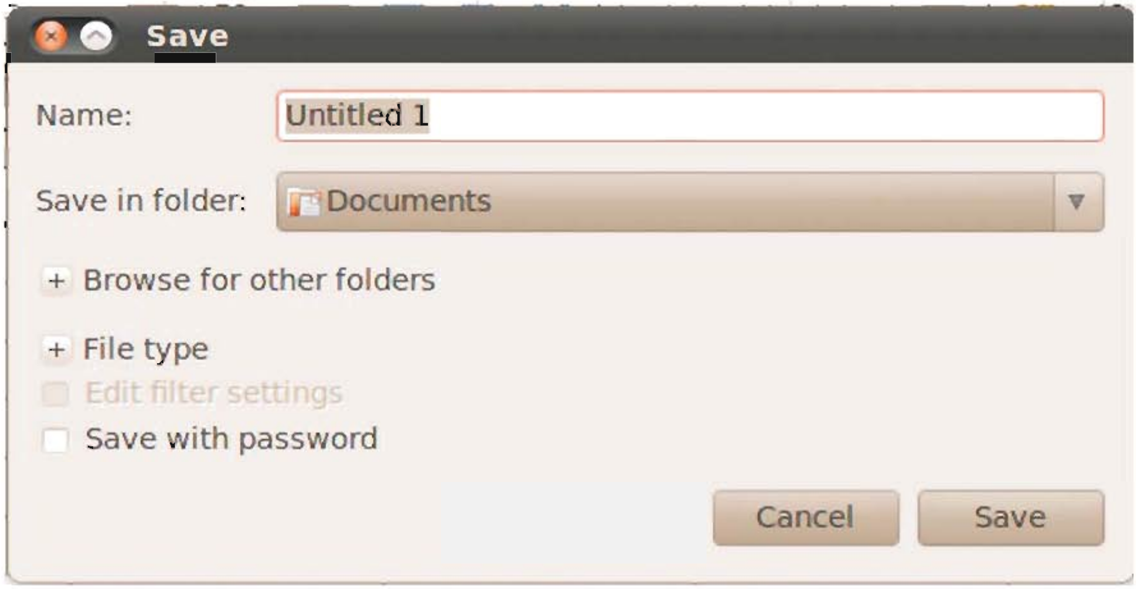
કોષ્ટક 5.2 : કેલ્સીના વિવિધ મેનૂ

જો તમે કોઈ એક મેનૂ પસંદ કરશો તો તેને અનુરૂપ સબમેનૂ સ્ક્રીન ઉપર પ્રદર્શિત થશે. સબમેનૂના વિકલ્પો સાથે એક ત્રિકોણ (▸) આકાર કદાચ તમારા ધ્યાન ઉપર આવ્યો હશે. જ્યારે આ પ્રકારની ત્રિકોણ (▸) સંજ્ઞા સ્ક્રીન ઉપર પ્રદર્શિત થાય ત્યારે તે વિકલ્પમાં વધારાના વિકલ્પો ઉપલબ્ધ હોય છે; જેમાંથી એક યોગ્ય વિકલ્પની પસંદગી તમે કરી શકો છો. આકૃતિ 5.5માં આ પ્રકારની સ્થિતિ દર્શાવેલ છે. આકૃતિ 5.5માં File મેનૂની અંદર 'New' વિકલ્પની પસંદગી સાથે ત્રિકોણ (▸) આકાર વધારે વિકલ્પોને સૂચવે છે. આ વિકલ્પો બીજી કાળી ઊભી હરોળનાં સ્વરૂપમાં પોપ ડાઉન મેનૂ તરીકે રજૂ કરવામાં આવે છે.



આકૃતિ 5.5 : મેનૂના ઉપલબ્ધ વિકલ્પો

જો ઉપયોગકર્તાની સૂચનાઓનું પાલન કરવા માટે વધારાની માહિતીની જરૂરિયાત હોય તો (...) સંજ્ઞા દર્શાવતું એક ડાયલૉગ બૉક્સ સ્ક્રીન ઉપર પ્રદર્શિત થશે. ધારો કે તમે **Save as** વિકલ્પ પસંદ કરો છો તો આકૃતિ 5.6માં દર્શાવ્યા પ્રમાણે કૅલ્સી તમને ફાઈલનું નામ અને કઈ જગ્યાએ તે ફાઈલનો સંગ્રહ કરવો છે તેની માહિતી તમને પૂછશે.



આકૃતિ 5.6 : Save ડાયલૉગ બૉક્સ

એ જ રીતે તમે **"Print"** વિકલ્પ નિહાળશો તો **"Print"** વિકલ્પનાં અંતમાં બિંદુઓની સંજ્ઞા (...) પણ તમે જોઈ શકશો; જેનો અર્થ એ છે કે અનેક પ્રિન્ટ વિકલ્પો દર્શાવતું એક ડાયલૉગ બૉક્સ સ્ક્રીન ઉપર પ્રદર્શિત થશે. આપણે આ બધા મેનૂ વિશે ઊંડાણમાં ચર્ચા હવે પછી કરીશું.

ટૂલબાર (Toolbar)

સૌથી સામાન્ય આદેશોને ટૂલબાર સ્ક્રીન ઉપર જુદા જુદા બટન સ્વરૂપમાં રજૂ કરે છે. માઉસ પૉઈન્ટરને ફક્ત તે બટન ઉપર ક્લિક કરીને તમે તે યુટિલિટીને પસંદ કરી શકો છો. કદાચ જો કોઈ બટન શું કાર્ય કરે છે તે તમે ન જાણતા હો તો તમારે ફક્ત તે બટન ઉપર માઉસ પૉઈન્ટરને આમતેમ ફેરવવું પડશે. આથી તરત જ તે બટનને અનુરૂપ કાર્યનું નામ સ્ક્રીન ઉપર પ્રદર્શિત થશે. **View → Toolbars** પસંદ કરીને તમે તમારી જરૂરિયાત પ્રમાણે ટૂલબારને પ્રદર્શિત કરી પણ શકો અને પ્રદર્શિત થતા બંધ પણ કરી શકો. પૂર્વનિર્ધારિત રીતે (સિસ્ટમ દ્વારા સેટ કરેલો) ટૂલબાર નીચે પ્રમાણે છે :

- **સ્ટાન્ડર્ડ ટૂલબાર (Standard toolbar) :** ટૂલબારનું નામ સૂચવે છે તે પ્રમાણે, આ ટૂલબાર ઉપર File અને Edit મેનૂમાંથી વારંવાર વપરાતા આદેશો રાખેલા હોય છે. તાજેતરમાં ઉપયોગમાં લેવાયેલા આઇકોન આ બાર ઉપર જોવા મળે છે. સામાન્ય રીતે નવો કેલ્સી દસ્તાવેજ બનાવવા, હયાત દસ્તાવેજ ખોલવા, દસ્તાવેજમાં લખાણની જોડણી ચકાસવા, માહિતી કટ અને પેસ્ટ કરવા અને દસ્તાવેજ પ્રિન્ટ કરવા વગેરે માટેના આઇકોન તમે આ સ્ટાન્ડર્ડ ટૂલબાર ઉપર જોશો, આકૃતિ 5.7માં એક સ્ટાન્ડર્ડ ટૂલબાર દર્શાવેલ છે.

આકૃતિ 5.7 : સ્ટાન્ડર્ડ ટૂલબાર

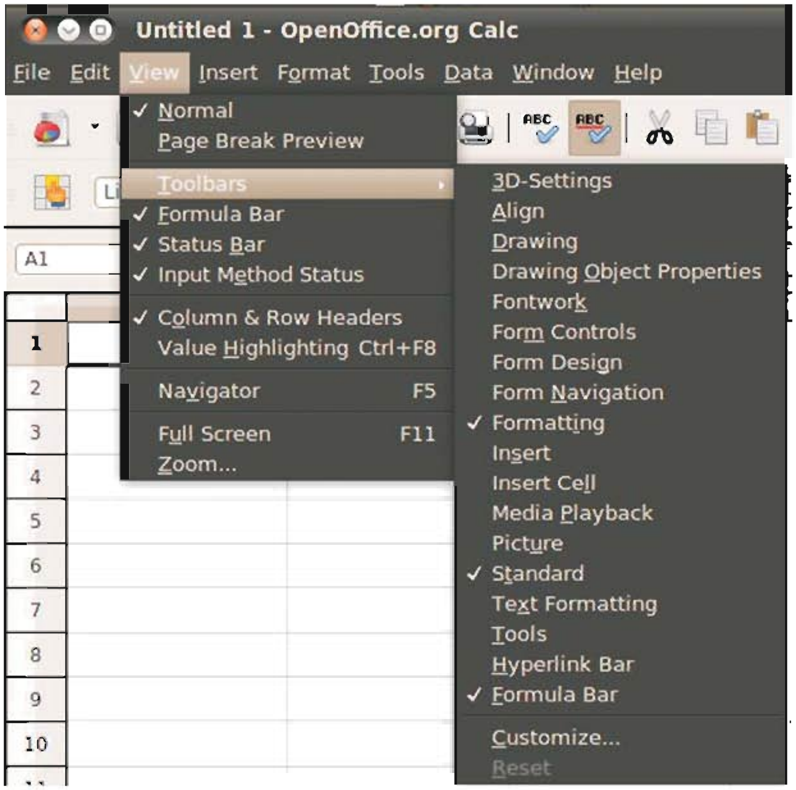
- **ફૉર્મેટિંગ ટૂલબાર (Formatting toolbar) :** સેલની માહિતીને ફૉર્મેટ કરવા માટેનાં સૌથી વધારે વપરાતા આદેશો ફૉર્મેટિંગ ટૂલબાર ઉપર હોય છે. આ ટૂલબાર ઝડપથી ફૉર્મેટિંગ કરવાની સગવડ પૂરી પાડે છે. જેમ જેમ તમે વિનિયોગનો ઉપયોગ કરતા જશો તેમ તેમ આ બાર ઉપરનાં આઇકોન બદલાતાં જશે. જો કોઈ આઇકોન આ બાર ઉપર ઉપલબ્ધ ન હોય તો ફૉર્મેટિંગ ટૂલનાં બિલકુલ અંતમાં જઈને તમે આઇકોન ઉમેરવાનો વિકલ્પ પસંદ કરી શકો. આકૃતિ 5.8માં એક લાક્ષણિક ટૂલબાર દર્શાવેલ છે.

આકૃતિ 5.8 : ફૉર્મેટિંગ ટૂલબાર

- **ફૉર્મ્યૂલા ટૂલબાર (Formula toolbar) :** સેલમાં કોઈ સૂત્ર લખવા અથવા તેમાંની માહિતી સુધારવાની સગવડ ફૉર્મ્યૂલા ટૂલબાર ઉપર હોય છે. આકૃતિ 5.9માં એક લાક્ષણિક ફૉર્મ્યૂલા ટૂલબાર દર્શાવેલ છે.

આકૃતિ 5.9 : ફૉર્મ્યૂલા ટૂલબાર

ટૂલબાર ઉપર રહેલાં બટન બદલી શકાય છે. ઉપયોગકર્તા પોતાની ઇચ્છા પ્રમાણે ટૂલબાર ઉપર બટન ઉમેરી શકે છે, બટન દૂર કરી શકે છે તેમજ બટનની શ્રેણી (ગોઠવણી) બદલી પણ શકે છે. આકૃતિ 5.10માં દર્શાવ્યા પ્રમાણે તમે ટૂલબાર ઉમેરી શકો તેમજ દૂર પણ કરી શકો છો.

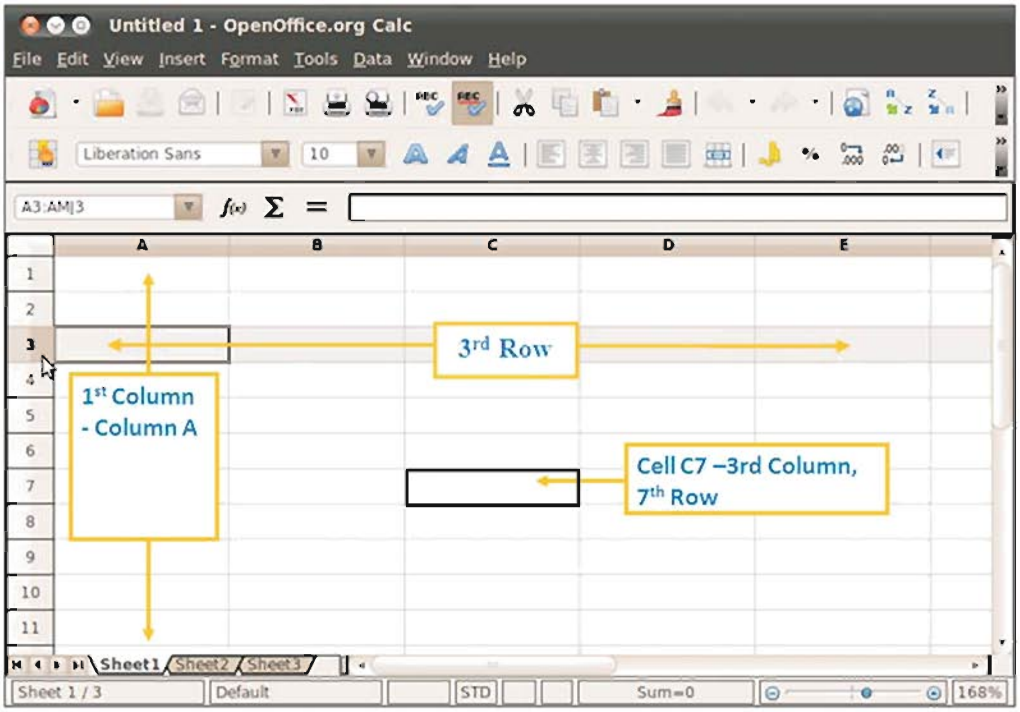


આકૃતિ 5.10 : ટૂલબાર ઉમેરવા / દૂર કરવા

આડી હરોળ, ઊભી હરોળ અને સેલ (Rows, columns and cells)

દરેક વર્કશીટને ઊભી હરોળ (કૉલમ) અને આડી હરોળ(રો)માં વિભાજિત કરીને સેલ બનાવવામાં આવે છે. દરેક રો અને કૉલમને ક્રમ આપવામાં આવે છે; કૉલમને મૂળાક્ષરો (અને તેનાં સંયોજનો) અને રોને સંખ્યાથી ઓળખવામાં આવે છે. વર્કશીટની ત્રીજી રો આકૃતિ 5.11માં હાઈલાઇટ થયેલી છે.

કેલ્સીની દરેક કૉલમ સેલની ઊભી શ્રેણી છે. સ્પ્રેડશીટના યોગ્ય અક્ષર ઉપર ક્લિક કરીને આખી કૉલમ પસંદ કરી શકાય છે. આકૃતિ 5.11માં વર્કશીટની પહેલી કૉલમ દર્શાવેલ છે.



આકૃતિ 5.11 : કેલ્સીની રો

સેલ એ રો (આડી હરોળ) અને કૉલમ(ઊભી હરોળ)નું છેદનબિંદુ છે. (રેખાઓ અરસપરસ છેદવાથી બનતો વિસ્તાર છે.) સેલને કૉલમનાં મૂળાક્ષર (મૂળાક્ષરો) અને રો નંબરનાં સંયોજનથી ઓળખવામાં આવે છે, દા.ત. સેલ C7. આકૃતિ 5.11માં સેલ C7 દર્શાવેલ છે, જે સેલ હાલમાં પસંદ કરેલ હોય તેને કાર્યરત (ઍક્ટિવ) સેલ કહેવામાં આવે છે. અહીં આકૃતિમાં સેલ C7 ને કાર્યરત સેલ બતાવ્યો છે. (અહીં એ નોંધ કરશો કે વાસ્તવિક સ્ક્રીનના કાર્યરત સેલની જમણી બાજુની નીચે વત્તાનું ચિહ્ન હોય છે.) અહીં જે ચોરસ બૉક્સ દેખાય છે તે કાર્યરત સેલનું સૂચક છે. કાર્યરત સેલના ખૂણામાં ભરેલો એક નાનો લંબચોરસ હાઈલાઇટ થયેલ છે જે ઑટોફિલ હૅન્ડલ (autofill handle) તરીકે ઓળખાય છે, જેની ચર્ચા આપણે હવે પછી આગામી પ્રકરણમાં કરીશું.

આપણી પાસે સરખામણીમાં અંગ્રેજી મૂળાક્ષરો (26) થોડા અને વધારે સંખ્યામાં રો હોવાથી આપણને મૂળાક્ષરોનાં સંયોજનનો ઉપયોગ કરવાની જરૂર પડે છે જેમ કે AA, AB, …, AZ, BA, ….., BZ, CA, …. વગેરે. કેલ્સીમાં 1024 કૉલમ અને 1,048,576 રો હોય છે.

સેલ એ સ્પ્રેડશીટનો એક પાયારૂપ મૂળભૂત ભાગ છે. સેલમાં ડેટા અને સૂત્રો ઉમેરવામાં આવે છે. એટલે કે, સેલ અલગ અલગ ઘટકો જેવા કે શાબ્દિક માહિતી, સંખ્યા અને સૂત્રો ધરાવે છે. તમે કોઈ એક સેલ ઉપર ક્લિક કર્યા પછી ડ્રૅગ કરવાથી એક કરતાં વધારે રો, કૉલમ અથવા સેલ પસંદ કરી શકો છો.

ફૉર્મ્યૂલા અને ડેટા ઇનપુટ બાર (Formula and data input bar)

ટૂલબારની તરત નીચે અને વર્કશીટના ટોચ વિસ્તાર ઉપર એક મોટી ખાલી લીટી ચકાસો. તમે આ પહેલાના વિભાગમાં આકૃતિ 5.9માં આ તમે જોયું હશે. આ ખાલી જગ્યા વપરાશકર્તાને ડેટા અને સૂત્ર દાખલ કરવા માટે પૂરી પાડવામાં આવે છે. એક વાર ડેટા દાખલ કરવામાં આવે એટલે તે કોઈ ચોક્કસ સેલમાં દેખાશે.

જે સેલ હાલમાં પસંદ કરેલો હોય તેનું સરનામું ફૉર્મ્યૂલા બારમાં ડાબી બાજુએ જોવા મળે છે. જમણી બાજુએ f(x) વડે સૂચવવામાં આવેલ બૉક્સ એક વિસ્તાર પૂરો પાડે છે જેના વડે તમે સેલમાં ડેટા અથવા સૂત્રો દાખલ કરી શકો છો.

કાર્યવિસ્તાર (વર્કસ્પેસ - Workspace)

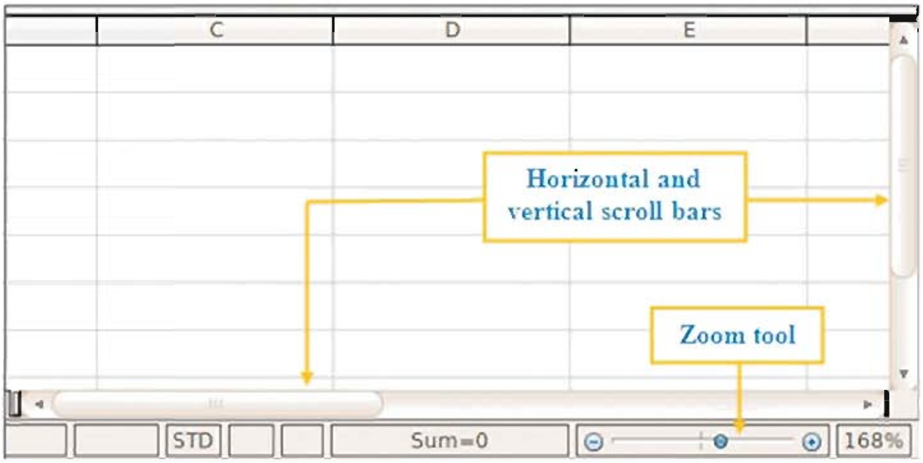
અનેક ખાલી સેલની જાળ (ગ્રીડ) એક વિસ્તાર બનાવે છે, જેમાં વપરાશકર્તાના ડેટા નજરે પડે છે. તમે કોઈ ચોક્કસ સેલ ઉપર ક્લિક કરી (તે સેલને કાર્યરત બનાવવા માટે) સીધા તેમાં ડેટા તેમજ સૂત્ર દાખલ કરી શકો છો. સેલમાં તમને સૂત્ર જોવા મળશે નહીં પણ ડેટા સ્વરૂપે તે સૂત્રનું પરિણામ જોવા મળશે.

ઝૂમ ટૂલ (Zoom tool)

વર્કશીટને ઝૂમ ઇન / આઉટ કરવા માટે ઝૂમ ટૂલ વપરાય છે. તમે વધારે સારા નિરીક્ષણ માટે વર્કશીટને ઝૂમ આઉટ કરી શકો. આકૃતિ 5.12માં ઝૂમ ટૂલને હાઇલાઇટ કર્યું છે જેના વડે જોઈ શકશો કે વર્કશીટ 168% ઝૂમ કરેલી છે.

સ્ક્રોલ બાર (Scroll bar)

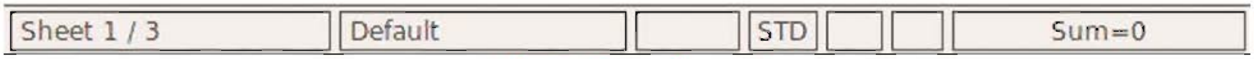
કમ્પ્યૂટર સ્ક્રીન(મોનિટર)ના કદ કરતાં વર્કશીટ મોટી હોવાથી આકૃતિ 5.12માં દર્શાવ્યા પ્રમાણે કેલ્સી આપોઆપ શીટની બંને બાજુએ તમને સ્ક્રોલબાર પૂરા પાડે છે. એટલે કે, સ્ક્રીન ઉપર તમે આડા અને ઊભા સ્ક્રોલબાર જોઈ શકશો. સ્ક્રોલબારના ઉપયોગથી તમને માહિતી નિહાળવાની વધારે અનુકૂળતા રહે છે, ખાસ કરીને જ્યારે ડેટા સ્ક્રીનમાં સમાતો ન હોય.



આકૃતિ 5.12 : ઝૂમ ટૂલ સાથે ઊભા અને આડા સ્ક્રોલબાર

સ્ટેટસ બાર (Status bar)

દસ્તાવેજની હાલની સ્થિતિ સાથે અન્ય માહિતી જેવી કે પસંદ કરેલા સેલનો સરવાળો, પાનાંની શૈલી, પસંદ કરવાની રીત અને સંગ્રહ ન કરેલાં પરિવર્તનો આ બાર દર્શાવે છે. આકૃતિ 5.13માં એક લાક્ષણિક સ્ટેટસ બાર દર્શાવેલ છે.



આકૃતિ 5.13 : સ્ટેટસ બાર

શીટ ટૅબ (Sheet tab)

શીટ ટૅબનો ઉપયોગ કરીને દસ્તાવેજની અલગ અલગ વર્કશીટ(ફક્ત શીટ પણ કહેવામાં આવે છે)માં તમે જઈ શકો છો. શીટ ટૅબ વર્કશીટ માટે નેવિગેશન ટૂલ તરીકે કામ આપે છે. વર્તમાન વર્કશીટ ટૅબનું નામ સફેદ રંગની પૃષ્ઠભૂમિ ધરાવે છે અને અન્ય બધી શીટ રાખોડી રંગની પૃષ્ઠભૂમિ ધરાવે છે. તમે વર્કશીટને પસંદ કરવા માટે જરૂરી વર્કશીટ ઉપર ફક્ત ક્લિક કરો. તમે "Sheet1" લખાણ ઉપર ફક્ત રાઇટ ક્લિક કરો, તે પછી "Rename sheet..." વિકલ્પ પસંદ કરો અને આપણને પસંદ નામ ટાઇપ કરીને પણ શીટનું નામ બદલી શકાય છે. આકૃતિ 5.14માં શીટ ટૅબ દર્શાવેલ છે. આના વિકલ્પરૂપે, વિશિષ્ટ પરિસ્થિતિઓમાં, આકૃતિ 5.14માં દર્શાવ્યા પ્રમાણે બીજી વર્કશીટમાં જવા માટે તીરનો ઉપયોગ પણ કરી શકાય છે. તમે તીરનો ઉપયોગ કરીને હાલ પછીની વર્કશીટ (જમણી બાજુ), સૌથી અંતમાં (જમણી બાજુ), હાલ પહેલાંની વર્કશીટ (ડાબી બાજુ) અને પ્રથમ વર્કશીટ (ડાબી બાજુએથી સૌપ્રથમ) ઉપર જઈ શકો છો.

આકૃતિ 5.14 : વર્કશીટ પસંદ કરવા માટેની શીટ ટેબ

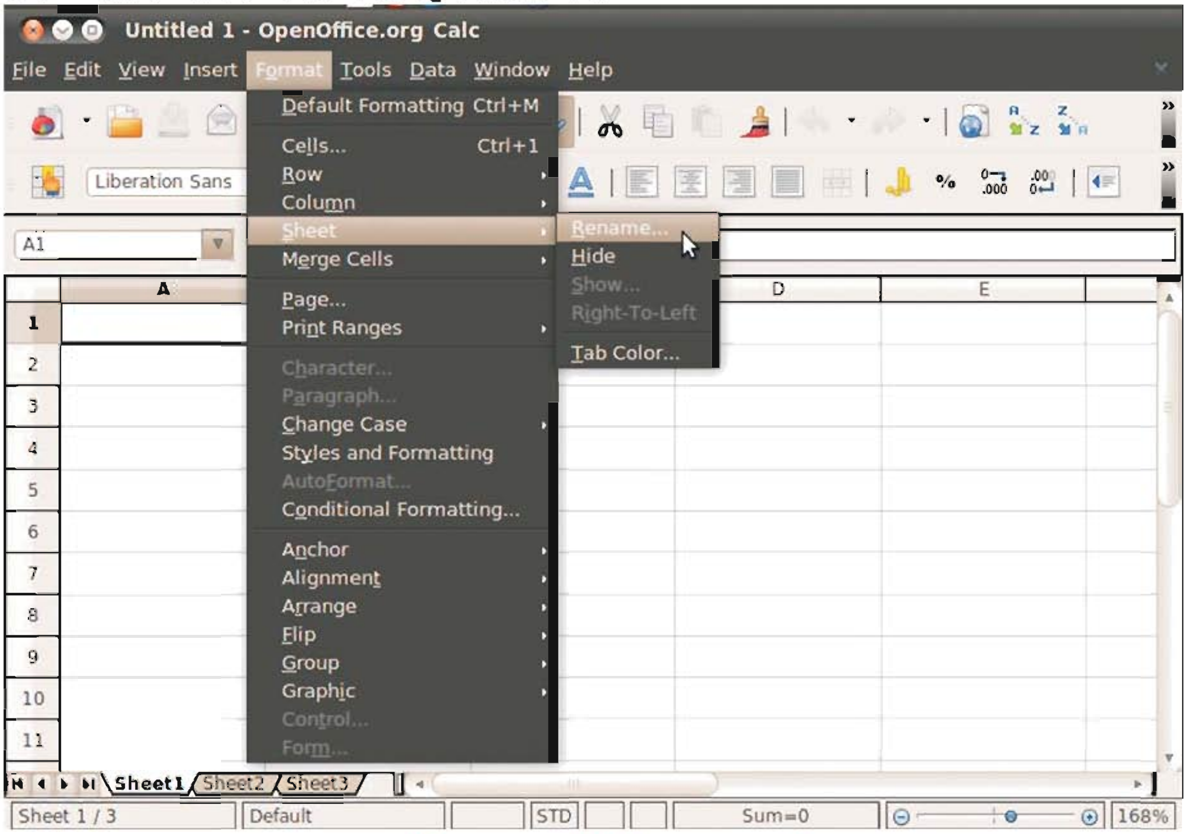
નવો કેલ્સી દસ્તાવેજ બનાવવો (Creating a Calc Document)

ચાલો, આપણે ખરીદીના બિલનો એક સાદો દસ્તાવેજ બનાવીએ. આ બિલમાં કંપનીનું નામ અને ઍડ્રેસ, ખરીદી કરેલી દરેક વસ્તુનાં નામ, કેટલા એકમ વસ્તુ ખરીદી, એકમ દીઠ ભાવ અને વસ્તુની કુલ રકમનો સમાવેશ કરવામાં આવે છે. આ બિલમાં દરેક વસ્તુની રકમનો કુલ સરવાળો, વેરાની રકમ, વળતર (જો કંઈ હોય તો) અને ગ્રાહકને આપવાની કુલ રકમનો પણ સમાવેશ કરવાની જરૂર છે. આ પ્રકારનો દસ્તાવેજ (ડૉક્યૂમેન્ટ) બનાવવા માટેનાં પગલાં નીચે જણાવેલાં છે :

પગલું 1 : સૌ પ્રથમ નીચે પ્રમાણે આદેશ આપો : Applications → Office → OpenOffice.org Spreadsheet આ આદેશ આપવાથી કેલ્સી ખૂલશે. તમે કેલ્સીનો સેતુ (ઇન્ટરફેઇસ) જોઈ શકશો.

પગલું 2 : તમે જોઈ શકશો કે કેલ્સી આપોઆપ ત્રણ વર્કશીટ દર્શાવે છે. જો કે અહીં અગાઉ જણાવ્યું તે પ્રમાણે ફક્ત રાઇટ ક્લિક કરી તમે જરૂરિયાત પ્રમાણે વર્કશીટ ડિલીટ કરી શકો છો તેમજ નવી વધારાની ઉમેરી પણ શકો છો. પૂર્વનિર્ધારિત આ ત્રણ વર્કશીટનાં નામ Sheet1, Sheet2, Sheet3 હોય છે. આમાંની કોઈ વર્કશીટને તમારી પસંદગીનું કોઈ ચોક્કસ નામ આપવા માટે નીચે જણાવ્યા પ્રમાણે કાર્ય કરો :

- સ્ટેટસ બારની તરત જ ઉપર સ્થિત વર્કશીટ ટેબ(આકૃતિ 5.14)માં જે વર્કશીટનું નામ બદલવાનું હોય તે વર્કશીટ ઉપર ક્લિક કરી તે વર્કશીટ પસંદ કરો.
- હવે આકૃતિ 5.15માં દર્શાવ્યા પ્રમાણે **Format** મેનૂ ઉપર ક્લિક કરો, તે પછી મેનૂનાં વિવિધ વિકલ્પોમાંથી **Sheet** વિકલ્પ પસંદ કરી અને તેનાં સબમેનૂના ઉપલબ્ધ વિકલ્પોમાંથી **Rename** પસંદ કરો.
- હવે યોગ્ય નામ (જેમ કે **"bill"**) ટાઇપ કરો અને કામગીરી પૂર્ણ કરવા માટે **OK** બટન ઉપર ક્લિક કરો. હવે વર્કશીટના નામમાં થયેલા ફેરફારનું નિરીક્ષણ કરો.



આકૃતિ 5.15 : શીટનું નામ બદલવું

પગલું 3 : હવે કોષ્ટક 5.3માં આપેલી માહિતી 'bill' નામની વર્કશીટમાં દાખલ કરો.

Sr. No.	Item	Quantity	Unit price
1	Pen	5	10
2	Pencil	6	2
3	Pencil box	1	50
4	Notebook	10	20
5	Notebook cover	10	2

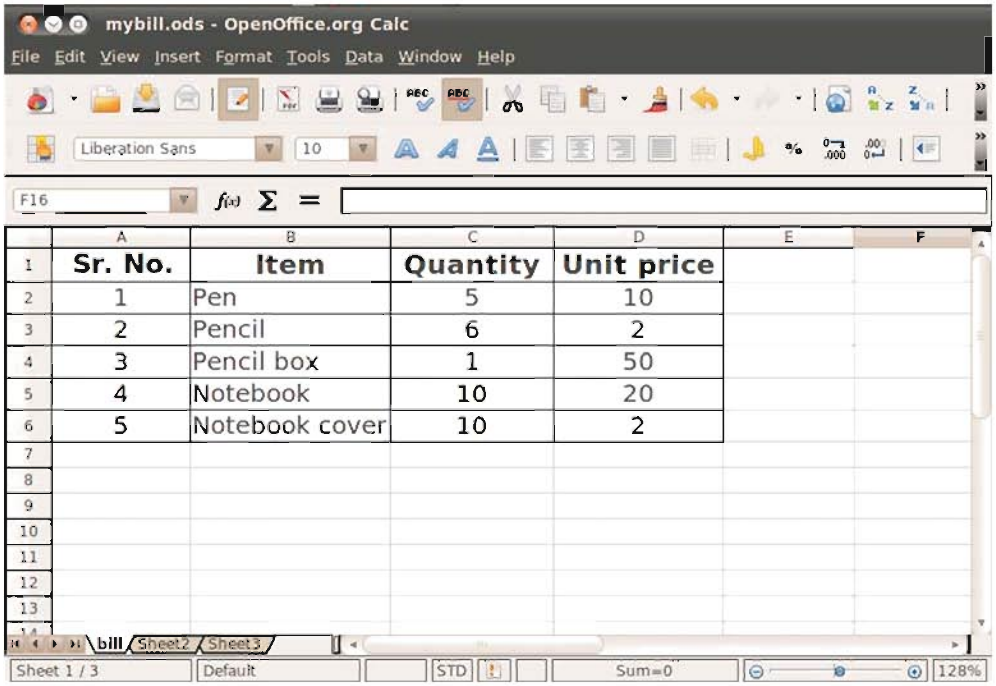
કોષ્ટક 5.3 : ખરીદીના બિલની માહિતી

'bill' નામની વર્કશીટમાં ડેટા દાખલ કરવા માટે નીચે મુજબ કાર્ય કરો :

- સેલ **A1** ઉપર લેફ્ટ ક્લિક કરો. હવે શબ્દ **"Sr. No."** દાખલ કરો અને એન્ટર કી દબાવો.
- સેલ **B1** ઉપર લેફ્ટ ક્લિક કરો. હવે શબ્દ **"Item"** દાખલ કરો અને એન્ટર કી દબાવો.
- સેલ **C1** ઉપર લેફ્ટ ક્લિક કરો. હવે શબ્દ **"Quantity"** દાખલ કરો અને એન્ટર કી દબાવો.
- સેલ **D1** ઉપર લેફ્ટ ક્લિક કરો. હવે શબ્દ **"Unit price"** દાખલ કરો અને એન્ટર કી દબાવો.

તમે ઊભી હરોળ(કૉલમ)નું કદ બદલવા માટે કૉલમને ડ્રેગ કરી શકો. તમે જ્યારે વસ્તુઓનાં નામ દાખલ કરી રહ્યા હોય ત્યારે તે ઊભી હરોળનું માપ બદલવાની જરૂર પડી શકે છે. આ કાર્ય કરવા માટે, જે ઊભી હરોળનું માપ તમે બદલવા ઇચ્છો છો તે ઊભી હરોળની ધાર પર માઉસ મૂકો અને જરૂરી લંબાઈ સુધી તેને ડ્રેગ કરો (ખેંચો). તમે જોઈ શકો છો કે પ્રથમ શીર્ષક લીટી તૈયાર થઈ ગઈ છે. તમે એક અથવા વધારે સેલ પસંદ કરી ફૉર્મેટિંગ ટૂલબાર વડે તેને ઘાટા (bold) કરી શકો છો.

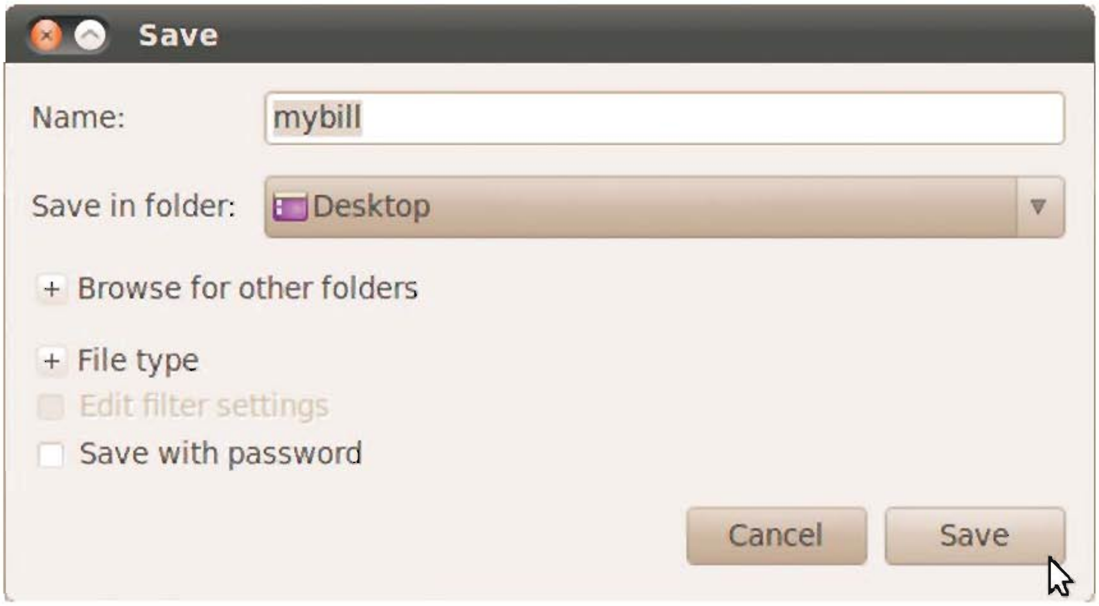
હવે સેલ A2 માં સંખ્યા "1", સેલ B2માં શબ્દ "Pen", સેલ C2માં સંખ્યા "5" અને સેલ D2માં સંખ્યા "10" દાખલ કરો. આ જ રીતે બાકીની લીટીઓમાં માહિતી દાખલ કરો. હવે વર્કશીટ આકૃતિ 5.16માં દર્શાવ્યા પ્રમાણે દેખાશે.



આકૃતિ 5.16 : bill વર્કશીટમાં દાખલ કરેલી માહિતી

પગલું 4 : હવે યોગ્ય નામ સાથે વર્કશીટનો સંગ્રહ કરો (સેવ કરો) જેથી દાખલ કરેલી માહિતી આકસ્મિક ગુમાવવી ન પડે. વર્કશીટનો સંગ્રહ કરવા માટે નીચે પ્રમાણે આદેશ આપો.

- **File → Save** ક્લિક કરો. આથી આકૃતિ 5.17માં દર્શાવ્યા પ્રમાણે એક ડાયલૉગ બૉક્સ સ્ક્રીન ઉપર દેખાશે; તેમાં ફાઈલનું નામ અને પાથ (જ્યાં તમે ફાઈલનો સંગ્રહ કરવા ઇચ્છો છો) જણાવો.
- હવે Save બટન ઉપર ક્લિક કરો.



આકૃતિ 5.17 : કેલ્સી દસ્તાવેજનો સંગ્રહ કરવો

સ્પ્રેડશીટનો **OpenOffice.org Calc** ફાઈલ સ્વરૂપે સંગ્રહ કરે છે અને તેનું અનુલંબન (એક્સટેન્શન) **.ods** હોય છે. તમે ફાઈલનો જે જગ્યાએ સંગ્રહ કરો તે સ્થાન તમારે યાદ રાખવું પડે જેથી ભવિષ્યમાં તે ફાઈલ મેળવી શકો.

હજુ સુધી આપણે ફક્ત જરૂરી ડેટા જ દાખલ કર્યો છે અને તે ફાઈલનો સંગ્રહ કર્યો છે. પણ આપણે દરેક વસ્તુની કિંમત, બધી વસ્તુઓની કુલ કિંમત, કર અને ચોખ્ખી ચૂકવવાપાત્ર રકમની ગણતરી કરી નથી. કોઈ ચોક્કસ વસ્તુ-(જેમ કે પેન)ની કિંમતની ગણતરી માટે આપણે વસ્તુની કિંમત(એક પેનની કિંમત)ના ખરીદેલ જથ્થા (પેનની સંખ્યા) સાથે ગુણાકાર કરવો પડે. પ્રથમ આડી હરોળ પ્રમાણે નંગદીઠ ₹ 10ના ભાવે 5 પેનની ખરીદી કરી હતી. આથી પહેલી વસ્તુની કુલ કિંમત ₹ 50 થાય. આ જ રીતે અન્ય વસ્તુઓ માટે ચૂકવવાપાત્ર રકમ નીચે પ્રમાણે છે :

પ્રથમ આડી હરોળ → 5 પેન, દરેકના ₹ 10	= 5*10 ₹	= ₹ 50
બીજી આડી હરોળ → 6 પેન્સિલ, દરેકના ₹ 2	= 6*2 ₹	= ₹ 12
ત્રીજી આડી હરોળ → 1 પેન્સિલ બૉક્સ, દરેકના ₹ 50	= 1*50 ₹	= ₹ 50
ચોથી આડી હરોળ → 10 નોટબુક, દરેકના ₹ 20	= 10*20 ₹	= ₹ 200
પાંચમી આડી હરોળ → 10 નોટબુક કવર, દરેકના ₹ 2	= 10*2 ₹	= ₹ 20

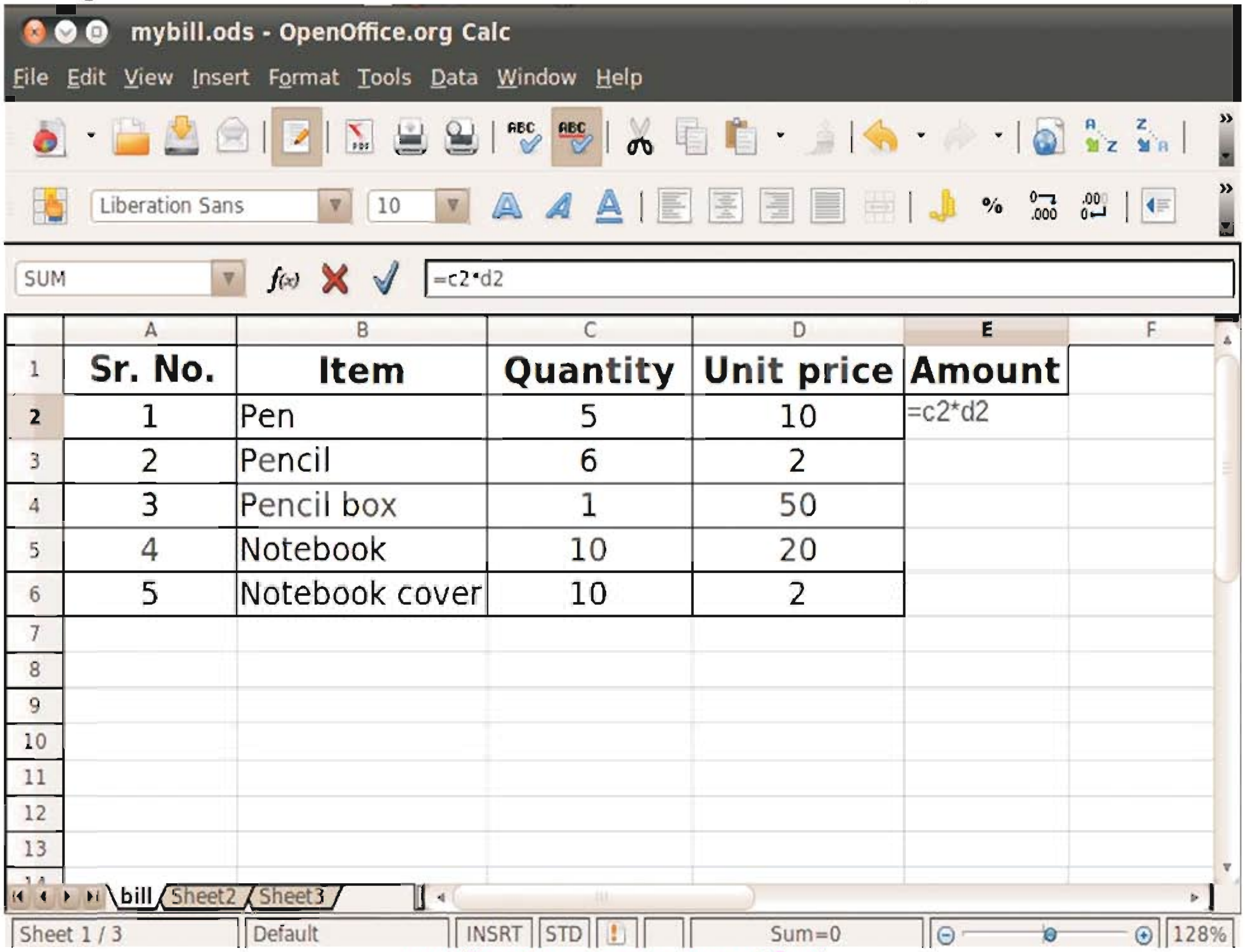
ઉપરની ગણતરી પ્રમાણે, આપણે એક સામાન્ય નિયમ તારવી શકીએ કે જો આપણે વસ્તુનાં x એકમ ખરીદીએ, દરેક એકમની y કિંમત હોય તો ચૂકવવાપાત્ર રકમ x*y થાય. આ રકમ (પરિણામ) **"Amount"** નામની છેલ્લી ઊભી હરોળમાં દાખલ કરવા માટે આપણે સંખ્યાનો ગુણાકાર જાતે કરીને સેલમાં લખી શકીએ. જો કે તેનાથી સારો વિકલ્પ આપણે જાતિગત સૂત્ર (generic formula) લખીએ તે છે.

સેલમાં કોઈ સૂત્ર દાખલ કરવા માટે આપણે સીધો ડેટા (જેમ કે =5*10) દાખલ કરી શકીએ અથવા ડેટા જે સેલમાં ઉપલબ્ધ છે તે સેલનો નિર્દેશ પણ આપણે કરી શકીએ. એટલે કે, સેલ C2 માં જો 5 હોય અને સેલ D2માં 10 હોય તો આ દાખલામાં (=5*10)ની કિંમત (=C2*D2) બરાબર થાય. કોઈ પણ સૂત્રની શરૂઆતમાં બરાબરનું ચિહ્ન '=' લખવું જરૂરી છે અને તેમ કરવામાં નિષ્ફળ જતાં સેલમાં નિયમ પ્રમાણે શાબ્દિક માહિતી તરીકે દાખલ થાય છે, (સંખ્યા દાખલ કરી હોય તો પણ) અને તેના ઉપર ગણતરીઓ કરવામાં આવતી નથી.

સીધી કિંમત કોઈ સેલમાં દાખલ કરવા કરતાં સેલ નિર્દેશ(cell reference)ના ઉપયોગના કેટલાક લાભ છે. પહેલો લાભ એ છે કે આપણે જાતે અંકગણિતની ક્રિયા કરવાની જરૂર પડતી નથી; કે જે અમુક સમયે ભૂલમાં પણ પરિણમી શકે. બીજો લાભ એ છે કે જ્યારે આપણે સેલમાં કિંમતને બદલીએ ત્યારે સૂત્રનું પરિણામ પણ આપોઆપ બદલાઈ જાય છે. ત્રીજો લાભ એ છે કે જ્યારે ડેટાનો કોઈ જથ્થો (બ્લૉક) એક જગ્યાએથી બીજી જગ્યાએ ખસેડવામાં આવે અથવા કેટલીક ઊભી કે આડી હરોળ ઉમેરવામાં / કાઢી નાખવામાં આવે તે સમયે સૂત્રમાં રહેલાં સેલ નિર્દેશ આપોઆપ બદલાઈ જશે.

ચાલો, આપણે અગાઉની 'bill' નામની વર્કશીટમાં દરેક વસ્તુદીઠ ચૂકવવાપાત્ર રકમની ગણતરી માટે એક ઊભી હરોળ વર્કશીટમાં ઉમેરીએ. હવે નીચે પ્રમાણે કાર્ય કરો :

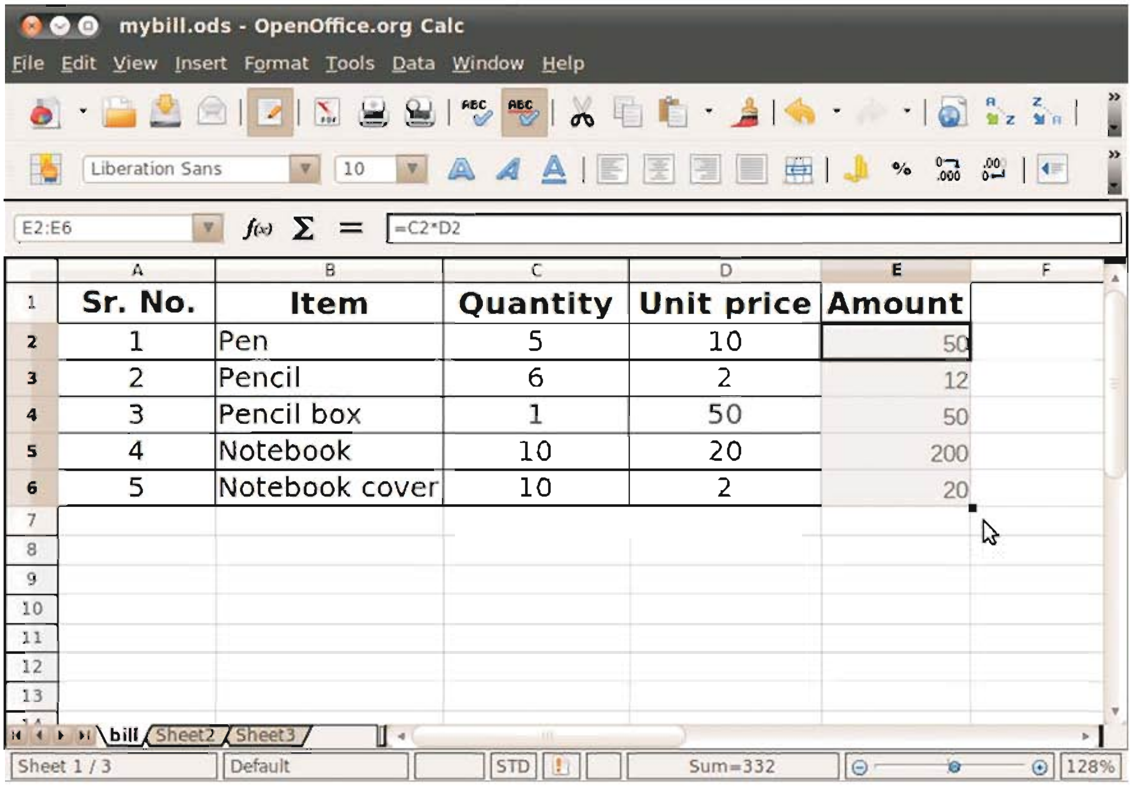
- સેલ E1 પસંદ કરો અને તેમાં **"Amount"** શબ્દ દાખલ કરો. તે પછી એન્ટર કી દબાવો. તમે આ શબ્દને ઘાટા (બોલ્ડ) કરી શકો છો.
- આકૃતિ 5.18માં દર્શાવ્યા પ્રમાણે સેલ E2 માં =C2*D2 ટાઇપ કરી સેલમાં સૂત્ર ઉમેરી શકો છો.



	A	B	C	D	E	F
1	Sr. No.	Item	Quantity	Unit price	Amount	
2	1	Pen	5	10	=c2*d2	
3	2	Pencil	6	2		
4	3	Pencil box	1	50		
5	4	Notebook	10	20		
6	5	Notebook cover	10	2		

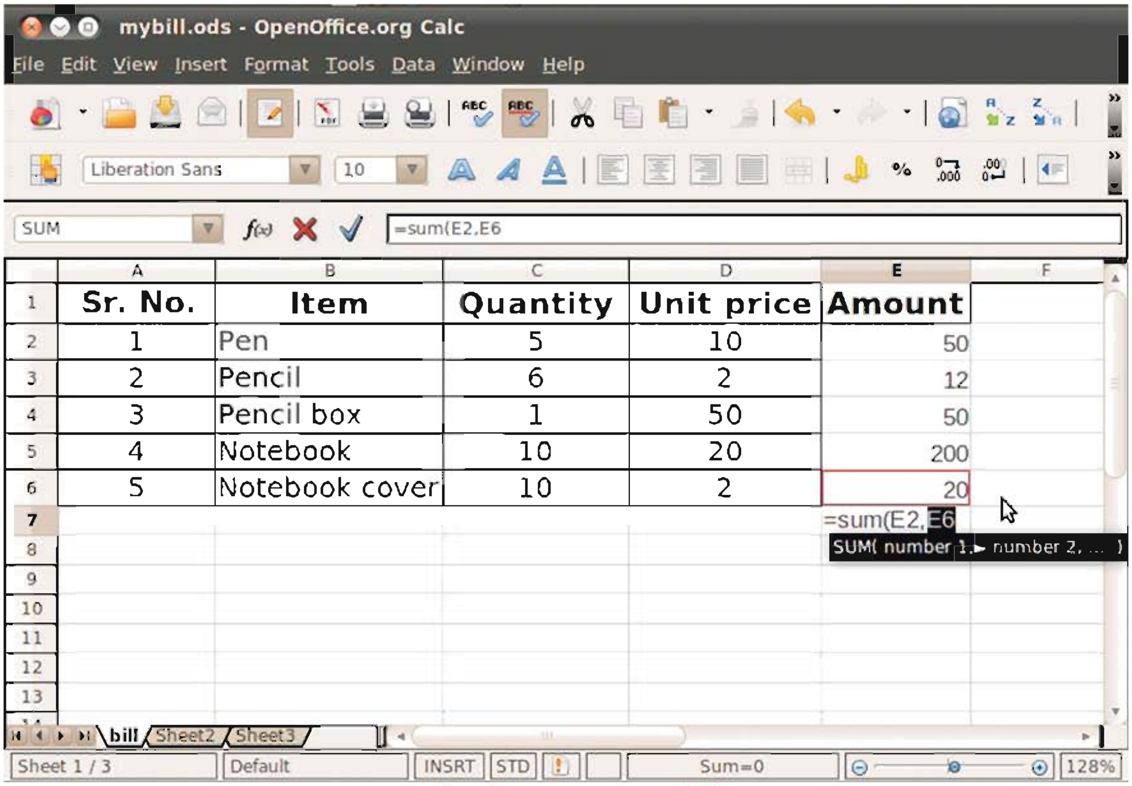
આકૃતિ 5.18 : સરળ સૂત્ર ઉમેરવું

- સેલ E2માં સૂત્ર દાખલ કર્યા પછી તમે જ્યારે એન્ટર કી દબાવો છો ત્યારે તમે સેલમાં ગુણાકારનું પરિણામ જોઈ શકશો.
- તમે સેલ E3માં (=C3*D3), સેલ E4માં (=C4*D4), સેલ E5માં (=C5*D5) અને સેલ E6માં (=C6*D6) સૂત્ર લખી શકો છો.
- આનો વિકલ્પ સેલ E2ની માહિતી તમે ઊભી હરોળના બાકીના સેલમાં સેલ E6 સુધી ફક્ત ડ્રેગ કરવાનો છે. આથી સૂત્રની નકલ આપોઆપ થઈ જશે. સેલની માહિતી ડ્રેગ કરવા માટે, સેલ ઉપર ક્લિક કરો. તે પછી તે સેલના જમણી બાજુના નીચેના ખૂણાને જરૂરી સંખ્યાના સેલ સુધી ડ્રેગ કરો.
- આકૃતિ 5.19માં દર્શાવ્યા પ્રમાણેનો આઉટપુટ તમે જોઈ શકશો.



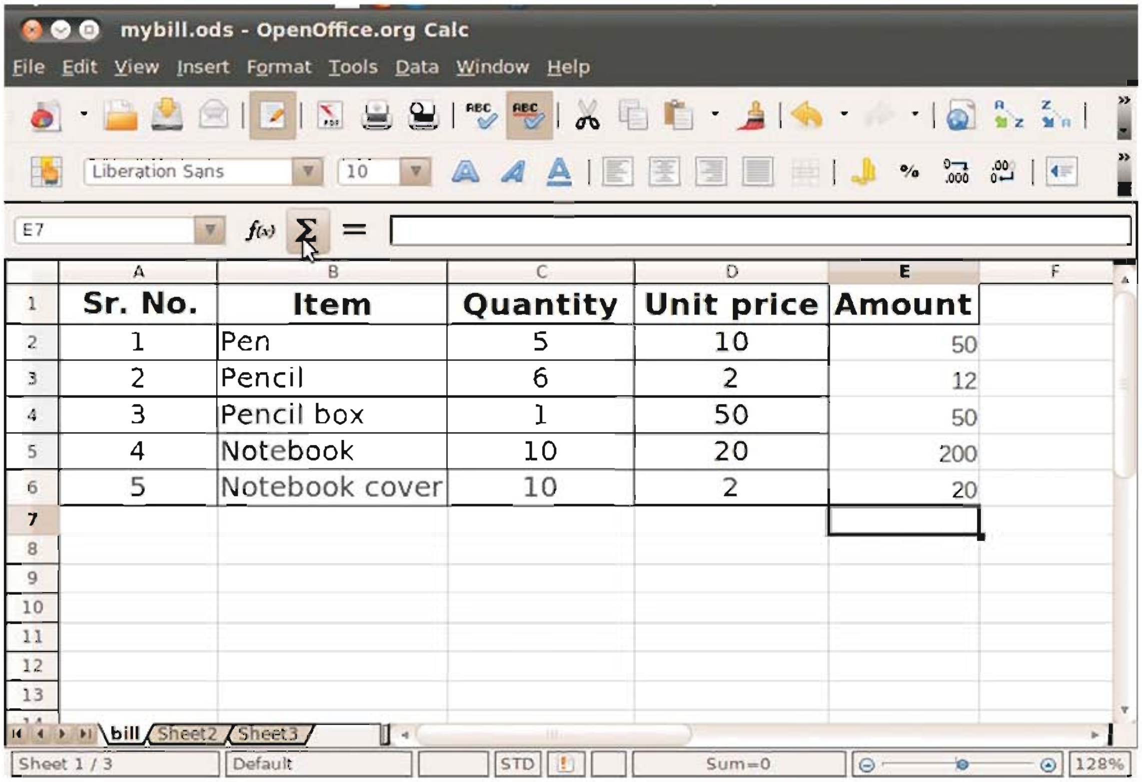
આકૃતિ 5.19 : એક સેલની કિંમતની ગણતરી કર્યા પછી અન્ય સેલમાં ડ્રેગ કરવું

આપણે જુદી જુદી વસ્તુઓની ચુકવવાપાત્ર રકમનો કુલ સરવાળો કરવાની પણ જરૂર છે. 'Amount' હરોળની બધી સંખ્યાઓનો સરવાળો કરવા માટે SUM વિધેયનો ઉપયોગ કર્યો છે. SUM વિધેયનું બટન સ્ક્રીનના ઉપરના ભાગમાં ફંક્શન બાર(function bar)માં આવેલું હોય છે. SUM વિધેય બટનનો ઉપયોગ કરી તમે પસંદ કરેલી સેલ રેન્જમાં રહેલી સંખ્યાનો સરવાળો આપોઆપ કરી શકો છો. આકૃતિ 5.20માં SUM વિધેય દર્શાવેલું છે.



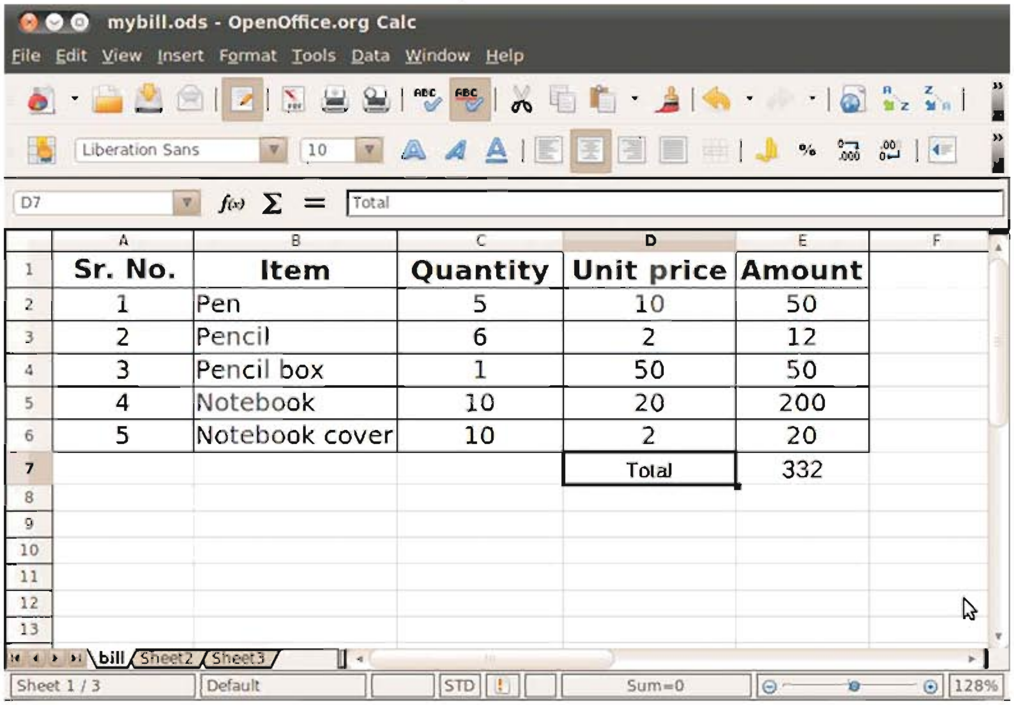
આકૃતિ 5.20 : SUM વિધેય

આકૃતિ 5.21માં દર્શાવ્યા પ્રમાણે સ્ક્રીનમાં સૌથી ઉપર ફોર્મ્યુલા બારમાં ઉપલબ્ધ ચિહ્ન ('Σ' તરીકે સૂચિત) પણ વાપરી શકાય છે.



આકૃતિ 5.21 : ઑટોસમ (autosum) ચિહ્નનો ઉપયોગ

સ્પ્રેડશીટમાં સેલની રેન્જ (range) દર્શાવવા માટે શરૂઆતના સેલનું ઍડ્રેસ અને છેલ્લા સેલનું ઍડ્રેસને વિસર્ગ(:)થી છૂટા પાડીને દર્શાવવામાં આવે છે. દા.ત., E2:E6 એ E2થી E6 સેલનો નિર્દેશ કરે છે. આના વિકલ્પરૂપે, તમે સેલ E7 પસંદ કરી તેમાં **=E2+E3+E4+E5+E6** સૂત્ર ટાઇપ પણ કરી શકો છો. સેલ D7 પસંદ કરી તેમાં શીર્ષક "Total" લખીને એન્ટર કી દબાવો. આથી તમે આકૃતિ 5.22માં દર્શાવ્યા પ્રમાણે સ્ક્રીન જોઈ શકશો.



આકૃતિ 5.22 : કુલ કિંમતની ગણતરી

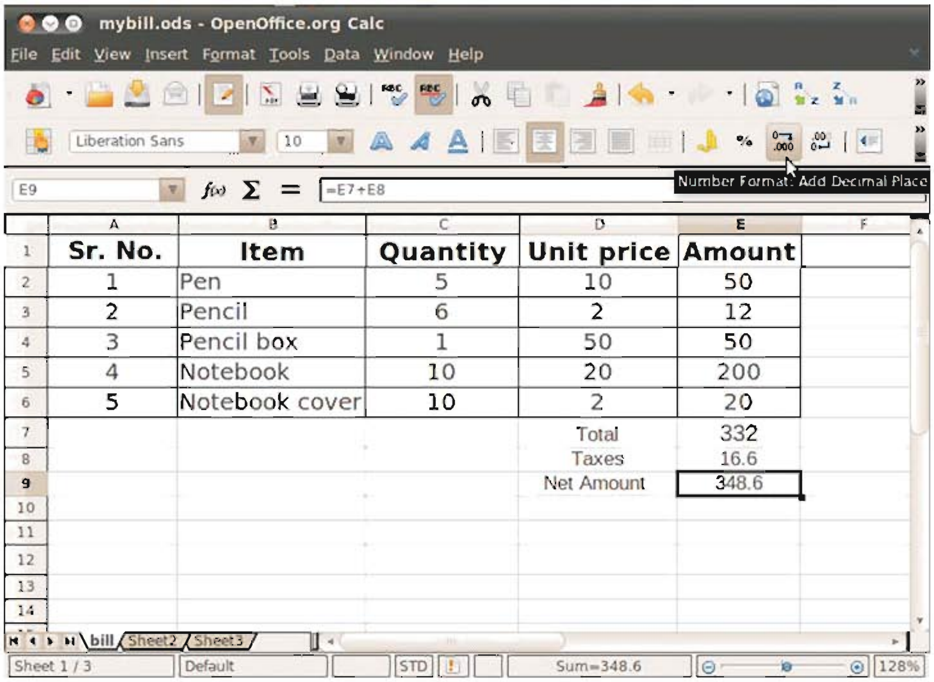
ચાલો, હવે આપણે ઉપર ગણતરી કરેલી કુલ કિંમત ઉપર કરની ગણતરી કરીએ. જો આપણે કુલ બિલની કિંમત ઉપર 5% કર ગણીએ તો કર રકમનું સૂત્ર મેળવવા માટે નીચે પ્રમાણે કરો :

કર = કુલ રકમ*5/100

- સેલ D8 પસંદ કરો અને તેમાં **"Taxes"** શીર્ષક લખો.
- સેલ E8 પસંદ કરો અને તેમાં =E7*5/100 સૂત્ર લખો.

ગ્રાહકને કુલ ચૂકવવાપાત્ર રકમની ગણતરી માટે કુલ રકમમાં કર ઉમેરવો પડે. સેલ E7 અને સેલ E8ની સંખ્યાનો સરવાળો કરી સેલ E9માં પ્રદર્શિત કરો. આપણે ઉપર ચર્ચા કરી તે પ્રમાણે sum વિધેયનો તમે ઉપયોગ કરી શકો અથવા સેલ E9 પસંદ કરી તેમાં (=E7+E8) સૂત્ર સીધું લખી શકો.

સેલ D9 પસંદ કરો અને તેમાં શીર્ષક **"Net Amount"** લખી એન્ટર કી દબાવો. તમે આકૃતિ 5.23માં દર્શાવ્યા પ્રમાણે પરિણામ જોઈ શકશો.



Sr. No.	Item	Quantity	Unit price	Amount
1	Pen	5	10	50
2	Pencil	6	2	12
3	Pencil box	1	50	50
4	Notebook	10	20	200
5	Notebook cover	10	2	20
			Total	332
			Taxes	16.6
			Net Amount	348.6

આકૃતિ 5.23 : ચૂકવવાપાત્ર રકમની ગણતરી

તમે અહીં જોઈ શકશો કે કુલ ચૂકવવાપાત્ર રકમ 348.6 થાય છે. તેને વધારે પરિચિત બનાવવા માટે આકૃતિ 5.23માં દર્શાવ્યા પ્રમાણે ટૂલબાર ઉપર ક્લિક કરીને આપણે દશાંશ ચિહ્ન ઉમેરી શકીએ છીએ. એક દશાંશ ચિહ્ન ઉમેરવાથી કુલ રકમની કિંમત 348.60 બનશે.

ધારો કે તમે વર્કશીટની સઘળી માહિતી એવી રીતે ખસેડવા ઇચ્છો છો કે જેથી તમે કંપનીનું નામ અને તારીખનો સમાવેશ કરી શકો. આ માટે નીચે જણાવેલાં પગલાંને અનુસરો :

- માઉસનો ઉપયોગ કરીને બધી માહિતીનો સમાવેશ થાય તે પ્રમાણે બધા સેલને પસંદ કરો. ઉપરના ઉદાહરણમાં, સંપૂર્ણ માહિતીનો સમાવેશ કરતા સેલ A1થી E9 છે.
- એડિટ (edit) મેનૂ પસંદ કરી તેમાંથી **"cut"** આદેશ પસંદ કરી આપણે પસંદ કરેલો ડેટા કટ કરો.
- હવે સેલ B3 ઉપર જાઓ અને ત્યાં ડેટા પેસ્ટ કરો.
- નિશ્ચિત કરો કે તમારા ડેટામાં કોઈ ફેરફાર થયેલો નથી.

તમે સૂત્રોમાં પરિવર્તન ઉપર ધ્યાન આપ્યું હશે. કુલ રકમ, કર અને ચૂકવવાપાત્ર રકમ હવે E હરોળને બદલે F હરોળનો નિર્દેશ કરે છે. સેલ જો એક જગ્યાએથી બીજી જગ્યાએ ખસેડવામાં આવે તો કેલ્સી સ્વયં રીતે સેલ નિર્દેશો બદલી નાખે છે, સિવાય કે સ્પષ્ટપણે તેમ ન કરવા સૂચના આપી હોય. હવે પછીના પ્રકરણમાં આપણે સાપેક્ષ સ્થાનાંક (relative address) અને નિરપેક્ષ સ્થાનાંક (absolute address) જેવી કાર્યરીતિ ઉપર વધારે શીખીશું.

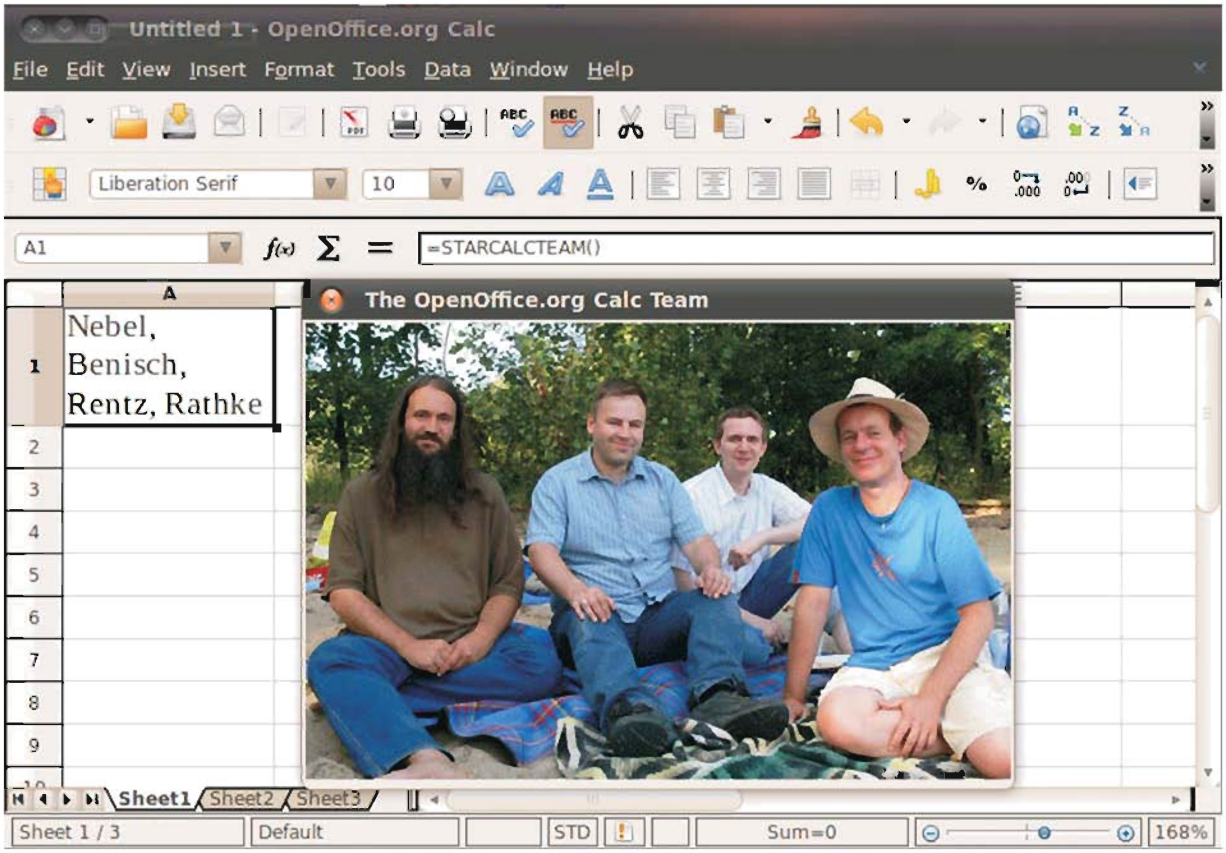
ફાઈલનો સંગ્રહ કરવો અને ફરી ખોલવી (Saving and Re-opening File)

સ્પ્રેડશીટમાં (અથવા કોઈ પણ કમ્પ્યૂટર વિનિયોગમાં) કામ કરતી વખતે તમારા કામનો વારંવાર સંગ્રહ કરવાની જરૂરિયાત હોય છે. વીજળીમાં વધઘટ થવી કે આવી અન્ય કોઈ સમસ્યાને કારણે તમે દાખલ કરેલો ડેટા ગુમાવવાનો સમય પણ આવે. ફાઈલનો વારંવાર સંગ્રહ કરવાની આદત સારી છે અને આકસ્મિક રીતે ડેટા ગુમાવવાને અટકાવે છે.

ફાઈલ ઉપરનું કામ એક વખત પૂર્ણ થઈ જાય પછી તમારે ફાઈલ બંધ કરવી પડે. તમે વિન્ડોમાં સૌથી ઉપર રહેલા ક્લોઝ (close) બટન ઉપર ક્લિક કરી શકો અથવા File મેનૂમાંથી close આદેશ આપો. તે ફાઈલ ફરી ખોલવા માટે તમારે File મેનૂમાંથી Open આદેશ આપવો પડે. આ કાર્ય માટે તમે સ્ટાન્ડર્ડ ટૂલનો ઉપયોગ પણ કરી શકો છો.

કેલ્સી વડે ઓપન ઑફિસ સ્યૂટ બનાવનારી ટુકડીને મળો (Meet the Developers of the OpenOffice Suit via Calc)

હવે તમે ઓપનઑફિસમાં સ્પ્રેડશીટ કઈ રીતે કાર્ય કરે છે તે જાણો છો. નીચે જણાવ્યા પ્રમાણે કાર્ય કરો. કેલ્સીમાં એક નવી વર્કબુક ખોલો. કોઈ પણ સેલ પસંદ કરી તેમાં =starcalcteam() ટાઈપ કરો, આથી તે ઓપનઑફિસ સ્યૂટ બનાવનારી ટુકડીના સભ્યોનો ફોટોગ્રાફ પ્રદર્શિત કરશે, આકૃતિ 5.24માં કેલ્સી બનાવનારી ટુકડી પ્રદર્શિત કરે છે.



આકૃતિ 5.24 : કેલ્સી બનાવનારી ટુકડી

સારાંશ

આપણે આ પ્રકરણમાં સ્પ્રેડશીટ શું છે તે બાબત શીખ્યા અને આવા પૅકેજનો ઉપયોગ કરીને બનાવવામાં આવતા કેટલાક લાક્ષણિક વિનિયોગો વિશે જાણ્યું. આ પ્રકરણમાં ઓપન ઑફિસ સ્યૂટના સ્પ્રેડશીટ પૅકેજના પરિચય અને મૂળભૂત કામગીરી ઉપર ધ્યાન કેન્દ્રિત કર્યું છે. તેની લાક્ષણિકતાઓ અને કામગીરી બાબતની ચર્ચા વાસ્તવિક ઉદાહરણને ક્રમબદ્ધ રીતે સમજાવીને કરેલી છે. તમે આ પાઠમાં જે મૂળભૂત બાબતો શીખ્યા છો, તેના વડે તમે નવી વર્કશીટ બનાવી શકો, તેમાંના ડેટામાં સુધારા-વધારા કરી શકો, કેટલાંક મૂળભૂત સૂત્રો વડે ડેટા ઉપર પ્રક્રિયા તેમજ વિશ્લેષણ કરી શકો, સ્પ્રેડશીટનો સંગ્રહ કરી શકો અને જરૂર હોય ત્યારે ફરી ખોલી શકો છો.

સ્વાધ્યાય

1. સ્પ્રેડશીટ પૅકેજનો ઉપયોગ કરી તૈયાર થતા યોગ્ય વિનિયોગની યાદી બનાવો.
2. કૅલ્સીમાં ફાઈલનો સંગ્રહ કરવાની અને ફરી ખોલવાની ક્રિયા સમજાવો.
3. કૅલ્સી વર્કશીટમાં સૂત્રો કઈ રીતે કાર્ય કરે છે તે સમજાવો.
4. કૅલ્સીની હયાત શીટનું નામ આપણે બદલી શકીએ? કઈ રીતે?
5. જો કૅલ્સી દસ્તાવેજમાં સ્ટાન્ડર્ડ ટૂલબાર દૃશ્યમાન ન હોય તો તમે શું કરશો?
6. નીચેનામાંથી સાચો વિકલ્પ પસંદ કરો :

(1) કૅલ્સી નીચેનામાંથી કયા પ્રકારના પૅકેજનો નિર્દેશ કરે છે?
   - (a) સ્પ્રેડશીટ
   - (b) મલ્ટીશીટ
   - (c) ડબલ શીટ
   - (d) નક્કી કરી શકાતું નથી.

(2) નીચેનામાંથી ક્યો વિનિયોગ કૅલ્સી માટે યોગ્ય નથી?
   - (a) સરવૈયું તૈયાર કરવું
   - (b) પરિણામનું વિશ્લેષણ
   - (c) કોઈ ઉત્પાદન વિશે માહિતી રજૂ કરવી
   - (d) આપેલા તમામ

(3) કૅલ્સીમાં બનાવેલી વર્કશીટનું અનુલંબન નીચેનામાંથી શું હોય છે?
   - (a) .ods
   - (b) .odd
   - (c) .xls
   - (d) .obj

(4) કૅલ્સીનાં કોઈ સેલમાં જો =starcalcteam() દાખલ કરવામાં આવે તો નીચેનામાંથી શું વર્કશીટમાં જોવા મળશે?
   - (a) તારાઓ
   - (b) કૅલ્સી બનાવનાર ટુકડીની તસવીર
   - (c) કૅલ્સી પરવાના (લાયસન્સ) માહિતી
   - (d) કૅલ્સી આવૃત્તિ માહિતી

(5) કૅલ્સી દસ્તાવેજની વર્કશીટમાં કોઈ વ્યક્તિએ દાખલ કરેલી કિંમતોના સરવાળાની ગણતરી કેવી રીતે કરી શકે?
   - (a) જાતે ગણતરી કરીને
   - (b) autosum વડે
   - (c) સૂત્ર વડે
   - (d) ઉપરોક્ત તમામ

(6) જો આપણે વર્કશીટના અન્ય કોઈ સેલનો નિર્દેશ ધરાવતા સૂત્રવાળા સેલને અન્ય જગ્યાએ ખસેડીએ તો સૂત્રમાંના સેલનું શું થશે?
   - (a) ગંતવ્ય પર સેલની રો (આડી હરોળ) અને કૉલમ (ઊભી હરોળ) બદલાઈ જશે.
   - (b) ગંતવ્ય પર સેલના રો નંબર બદલાઈ જશે.
   - (c) ગંતવ્ય પર સેલના કૉલમ નંબર બદલાઈ જશે.
   - (d) કશું બદલાશે નહીં.

પ્રાયોગિક સ્વાધ્યાય

1. આ પ્રકરણમાં ખરીદીના બિલના ઉદાહરણની ચર્ચા કરી હતી તેની રચના કરો.
2. તમે કોઈ સુપર સ્ટોરમાંથી મેળવેલ બિલનો અભ્યાસ કરો અને કૅલ્સીમાં તેની રચના કરો.
3. તમારા છ જુદા જુદા વિષયોના ગુણનો એક કૅલ્સી દસ્તાવેજ બનાવો. તે પછી ડેટાના આધારે કુલ ગુણ, સરેરાશ અને ટકા શોધો.
4. તમારા મિત્રોના ગુણ ઉમેરી આ સ્વાધ્યાયના ત્રીજા ઉદાહરણને વિસ્તારો.

કેલ્સીમાં ડેટાનું એડિટિંગ અને ફૉર્મેટિંગ

અગાઉના પ્રકરણમાં ચર્ચા કરી હતી તે પ્રમાણે સ્પ્રેડશીટમાં માહિતીનો સંગ્રહ કરવા માટેનો મૂળભૂત એકમ સેલ છે. ઊભી હરોળ (કૉલમ) અને આડી હરોળ (રો) એકબીજાને છેદવાથી બનતા ઘણા બધા સેલનો સમૂહ એટલે જ સ્પ્રેડશીટ. બધો જ ડેટા, સૂત્રો અને વિધેયો આ સેલની અંદર લખવામાં આવે છે. આ પ્રકરણ તમને ડેટા એડિટિંગ (ડેટામાં સુધારા-વધારા કરવાની ક્રિયા) અને ફૉર્મેટિંગ (ડેટાને યોગ્ય અને આકર્ષક સ્વરૂપમાં ગોઠવવા) સંબંધિત મૂળભૂત કામગીરી વિશે સંક્ષિપ્ત રૂપરેખા આપશે.

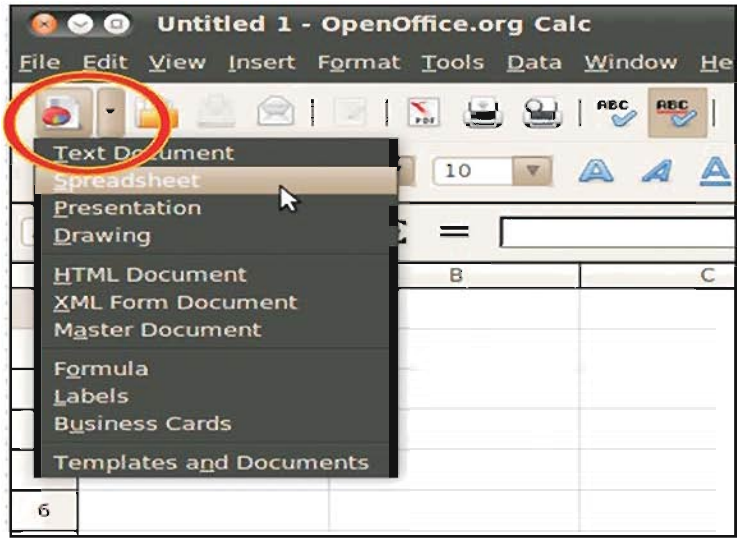
કેલ્સીમાં મૂળભૂત વર્કશીટ કાર્યો (Basic worksheet operations in Calc)

કેટલાંક કાર્યો જેવાં કે નવી અથવા હયાત સ્પ્રેડશીટ ખોલવી, સ્પ્રેડશીટનો સંગ્રહ કરવો, save as આદેશ વડે આખી સ્પ્રેડશીટનું નામ બદલવું, વર્કશીટનું નામ બદલવું, અનેક વર્કશીટ ઉમેરવી / દૂર કરવી વગેરે સ્પ્રેડશીટ અથવા વર્કશીટ સ્તર ઉપર વારંવાર થતી પ્રક્રિયાઓ છે. આ વિભાગમાં આ માટેના આદેશોની ચર્ચા કરી છે.

નવો દસ્તાવેજ બનાવવો (Creating a new document)

જ્યારે તમે OpenOffice.org Calc વિનિયોગ ખોલો છો ત્યારે તે આપોઆપ એક નવો (ખાલી) દસ્તાવેજ ખોલશે. જો તમે પહેલાંથી જ આ દસ્તાવેજનો ઉપયોગ કરી લીધો હોય અને તેમાં તમે કેટલીક માહિતી લખી હોય તો તમે બીજો નવો સ્પ્રેડશીટ દસ્તાવેજ ખોલી શકો છો. નવો કેલ્સી દસ્તાવેજ ખોલવા માટે, એટલે કે કેલ્સી સ્પ્રેડશીટ ખોલવા માટે નીચે મુજબની ક્રિયાઓ કરો :

- મેનૂબાર ઉપરથી **File → New → Spreadsheet** પસંદ કરો
- CTRL + N દબાવો, અથવા
- સ્ક્રીનના ઉપરના ભાગમાં ટૂલબાર ઉપરથી **New Document** આઈકોન ઉપર ક્લિક કરો અને સ્પ્રેડશીટ પસંદ કરો. આઇકોન (લંબગોળમાં દર્શાવેલ) આકૃતિ 6.1માં બતાવ્યા પ્રમાણે દેખાશે.



આકૃતિ 6.1 : નવી સ્પ્રેડશીટ ખોલવા માટેનો આઈકોન (icon)

જ્યારે તમે કોઈ નવી સ્પ્રેડશીટ ખોલો છો ત્યારે તે તમને ત્રણ અલગ અલગ વર્કશીટ આપે છે; જે રીતે તમે ફક્ત ત્રણ પાનાંની ગણિત માટે નોટબુક ખરીદી હોય જેમાં આડી હરોળો અને ઊભી હરોળોનું જાળું (ગ્રીડ) હોય છે. તમે પ્રકરણ 5માં ખરીદીના બિલની માહિતી દાખલ કરવા માટે કઈ રીતે ત્રણ વર્કશીટ સાથે સ્પ્રેડશીટ ખોલી શકાય તે શીખ્યા છો.

કોઈ હયાત કેલ્સી દસ્તાવેજને ઘણી બધી રીતે ખોલી શકાય. આમાંની પહેલી રીતમાં સૌપ્રથમ કેલ્સી ખોલી ફાઈલ મેનૂમાંથી open આદેશ આપો. આ માટેના આદેશોની શ્રેણી નીચે આપેલી છે :

- પસંદ કરોઃ **File → Open**
- **Open** ડાયલૉગ બૉક્સમાં પૂર્વનિર્ધારિત ફોલ્ડરમાંથી કે અન્ય ચોક્કસ ફોલ્ડરમાંથી તમારી જરૂરી ફાઈલ પસંદ કરો.
- આ ક્રિયા **Open** બટન ઉપર ક્લિક કરી પૂર્ણ કરો.

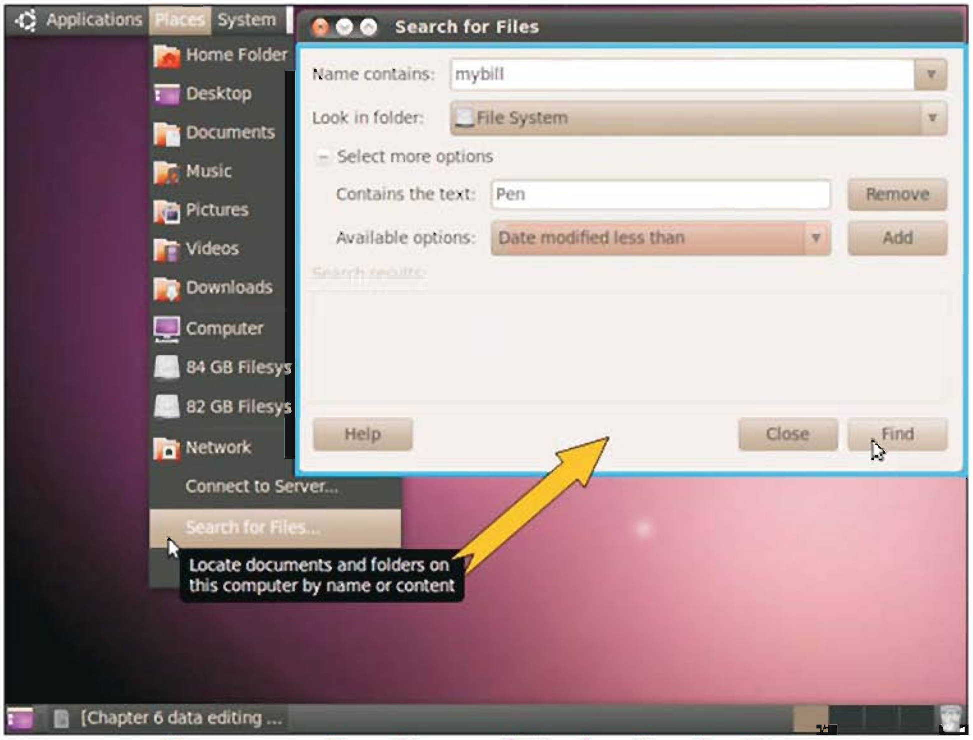
અન્ય વિકલ્પમાં, પહેલાં તમે કમ્પ્યૂટરમાં જરૂરી ફાઈલ શોધી કાઢો પછી નીચે જણાવ્યા પ્રમાણે કેલ્સી સૉફ્ટવેરમાં તેને ખોલો.

- **Places → Documents** આદેશ આપીને **Documents** ફોલ્ડર ખોલો.
- તમારી જરૂરી સ્પ્રેડશીટ ફાઈલ પસંદ કરી તેને ખોલો.

જો ઉપર જણાવેલાં પગલાંને અનુસરીને પણ તમે તમારી ફાઈલ શોધી ન શકો તો ઉબુન્ટુની સર્ચ (search) યુટિલિટી વડે ફાઈલને શોધી શકો છો. ખાસ કરીને જ્યારે તમારી જરૂરી ફાઈલની જગ્યા તમને યાદ ન હોય ત્યારે આ યુટિલિટી ખૂબ ઉપયોગી છે.

- આકૃતિ 6.2ના ઊભા મેનૂમાં દર્શાવ્યા પ્રમાણે **Places → Search for Files** આદેશ આપો.
- જ્યારે તમે **Search for Files** (જુઓ આકૃતિ 6.2માં માઉસ તીરની નિશાની) ઉપર ક્લિક કરો છો ત્યારે ફાઈલની માહિતી પૂછતું એક ડાયલૉગ બૉક્સ સ્ક્રીન ઉપર પ્રદર્શિત થશે. આ માહિતીમાં ફાઈલનું નામ, જે ફોલ્ડરમાં ફાઈલ શોધવાની હોય તેનું નામ અને અપેક્ષિત ફાઈલની અંદરની કેટલીક માહિતીનો સમાવેશ થાય છે. આપણે છેલ્લે ફાઈલમાં સુધારા-વધારા કર્યાની તારીખ જેવી કેટલીક તારીખ વડે પણ ફાઈલને શોધી શકીએ છીએ. આકૃતિ 6.2માં ડાયલૉગ બૉક્સ નિર્દેશ કરતું તીર સાથેનું એક બૉક્સ જુઓ.
- **Search for Files** ડાયલૉગ બૉક્સના **Name contains:** માં ફાઈલનું પૂર્ણ નામ અથવા નામનો થોડો ભાગ દાખલ કરો.
- હવે Find બટન ઉપર ક્લિક કરો.

આકૃતિ 6.2માં આ પગલાંઓ સચિત્ર દર્શાવ્યા છે. શોધવાની આ પ્રક્રિયાના પરિણામે ફાઈલની એક યાદી તે પ્રદર્શિત કરશે. તમે ફાઈલ ખોલવા માટે જરૂરી ફાઈલ ઉપર માઉસનો ઉપયોગ કરીને ડબલ ક્લિક કરો.



આકૃતિ 6.2 : ફાઈલ શોધવા માટેની શોધપ્રક્રિયાનું પરિણામ

સ્પ્રેડશીટનો સંગ્રહ કરવો અને બંધ કરવી (Saving and closing spreadsheet)

અગાઉના વિભાગમાં તમે કોઈ નવો દસ્તાવેજ અથવા હયાત દસ્તાવેજને જોવા માટે કે તેમાં સુધારા-વધારા માટે સ્પ્રેડશીટ દસ્તાવેજને તમે અત્યાર સુધીમાં ખોલેલો છે. ખુલ્લા દસ્તાવેજમાં એક વખત કામ પૂર્ણ થઈ જાય એટલે તેમાં કરેલાં પરિવર્તનોનો સંગ્રહ કરવાની તમને જરૂર પડે છે. કોઈ પણ ડેટા ગુમાવવાને અટકાવવા માટે કરેલાં પરિવર્તનોનો વારંવાર સંગ્રહ કરવો એ સલાહભર્યું છે.

આપણે પ્રકરણ 5માં વર્કશીટનો સંગ્રહ કઈ રીતે કરવો તે બાબત ચર્ચા કરી હતી. વર્કશીટનો સંગ્રહ કરવા માટે આદેશ આપો : **File → Save**. જો તમે દસ્તાવેજને અન્ય નવા નામથી કે અલગ પ્રકારની ફાઈલ તરીકે સંગ્રહ કરવા ઇચ્છતા હોય તો તમે Save as વિકલ્પને પણ પસંદ કરી શકો છો. અહીં અગાઉ જણાવ્યા પ્રમાણે, જો તમે સ્પ્રેડશીટનો પહેલી વખત સંગ્રહ કરો છો તો ફાઈલનું નામ અને તેને સંગ્રહ કરવાનો પાથ તમને પૂછશે.

જે ફાઈલ ખુલેલી છે (વર્તમાન) તેને બંધ કરવા માટે **File → Close** આદેશ આપો. જો તમે છેલ્લાં ફેરફારોનો સંગ્રહ ન કર્યો હોય તો કેલ્સી તમને ફાઈલ બંધ કરતા પહેલાં ફાઈલનો સંગ્રહ કરવાની એક તક આપશે.

વર્કશીટનો વિવિધ બંધારણ(ફૉર્મેટ)માં સંગ્રહ (Saving worksheet in different format)

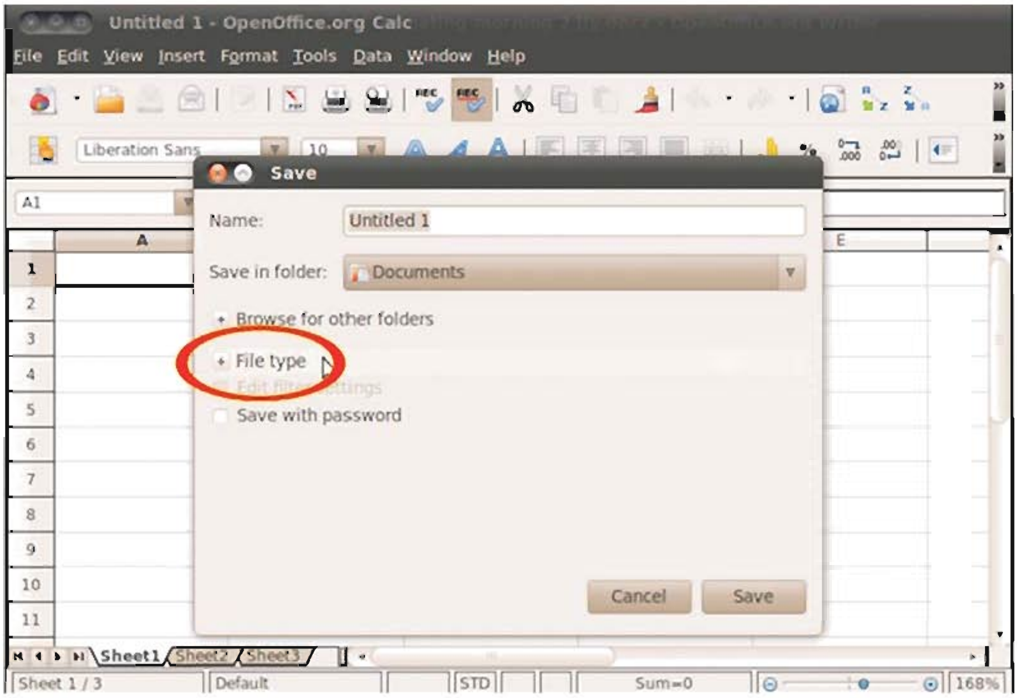
કેલ્સી પૂર્વનિર્ધારિત રીતે, સ્પ્રેડશીટને .ods અનુલંબન સાથે ફાઈલ સ્વરૂપે સંગ્રહ કરે છે. અન્ય કેટલાંક ફાઈલસ્વરૂપ (બંધારણ)ની યાદી કોષ્ટક 6.1માં આપી છે.

ફૉર્મેટ (બંધારણ)	અનુલંબન	વર્ણન
ODF સ્પ્રેડશીટ	ods	OpenOffice.org calc ફૉર્મેટ
ODF સ્પ્રેડશીટ ટેમ્પ્લેટ	ots	કેલ્સી સ્પ્રેડશીટ ટેમ્પ્લેટ ફૉર્મેટ
dBASE	dbf	ડેટા બેઝ ફાઈલ ફૉર્મેટ
શાબ્દિક CSV	csv	અલ્પવિરામથી કિંમતો અલગ પાડેલી શાબ્દિક ફાઈલ; ખાસ કરીને આવી ફાઈલ વિવિધ પ્રોગ્રામમાં માહિતી વિનિમય માટે વપરાય છે.
HTML દસ્તાવેજ (OpenOffice.org Calc)	html	વેબ પેઈજ ફૉર્મેટ
સુવાહ્ય (પોર્ટેબલ) દસ્તાવેજ બંધારણ	pdf	સૌથી વધારે વપરાતું ફૉર્મેટ; આ એક સુવાહ્ય દસ્તાવેજનું વિશ્વવ્યાપી એડોબ(Adobe)નું ફૉર્મેટ છે.
માઇક્રોસૉફ્ટ એક્સેલ 2007 XML	xlsx	એમએસઑફિસ 2007/2010ની સ્પ્રેડશીટ
માઇક્રોસૉફ્ટ એક્સેલ 2003	xls	એમએસઑફિસ 2003ની સ્પ્રેડશીટ

કોષ્ટક 6.1 : વિવિધ ફાઈલ ફૉર્મેટ કે જેમાં સ્પ્રેડશીટનો સંગ્રહ કરી શકાય

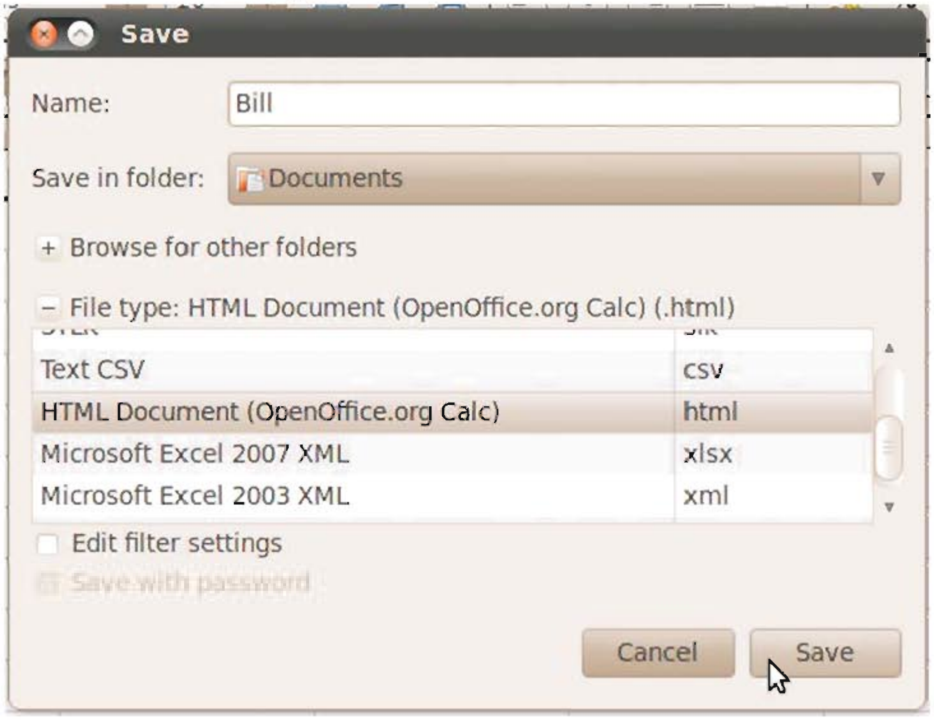
સ્પ્રેડશીટનો અન્ય ફાઈલ ફૉર્મેટમાં સંગ્રહ કરવા માટે :

- સૌપ્રથમ File અને પછી Save (કે જે દસ્તાવેજનો અગાઉ સંગ્રહ કર્યો નથી) અથવા Save As વિકલ્પ (ખુલ્લા અને અગાઉ સંગ્રહ કરેલાં દસ્તાવેજ માટે) પસંદ કરો.
- આકૃતિ 6.3માં દર્શાવ્યા પ્રમાણે file type મેનૂ (નિર્દેશિત) ખોલો.



આકૃતિ 6.3 : file type મેનૂ ખોલવું

- આકૃતિ 6.4માં દર્શાવ્યા પ્રમાણે જરૂરી ફાઈલ ફૉર્મેટની પસંદગી કરો.



આકૃતિ 6.4 : ફાઈલના પ્રકારની પસંદગી કરવી

- **Save** બટન ક્લિક કરી ફાઈલનો સંગ્રહ કરો.

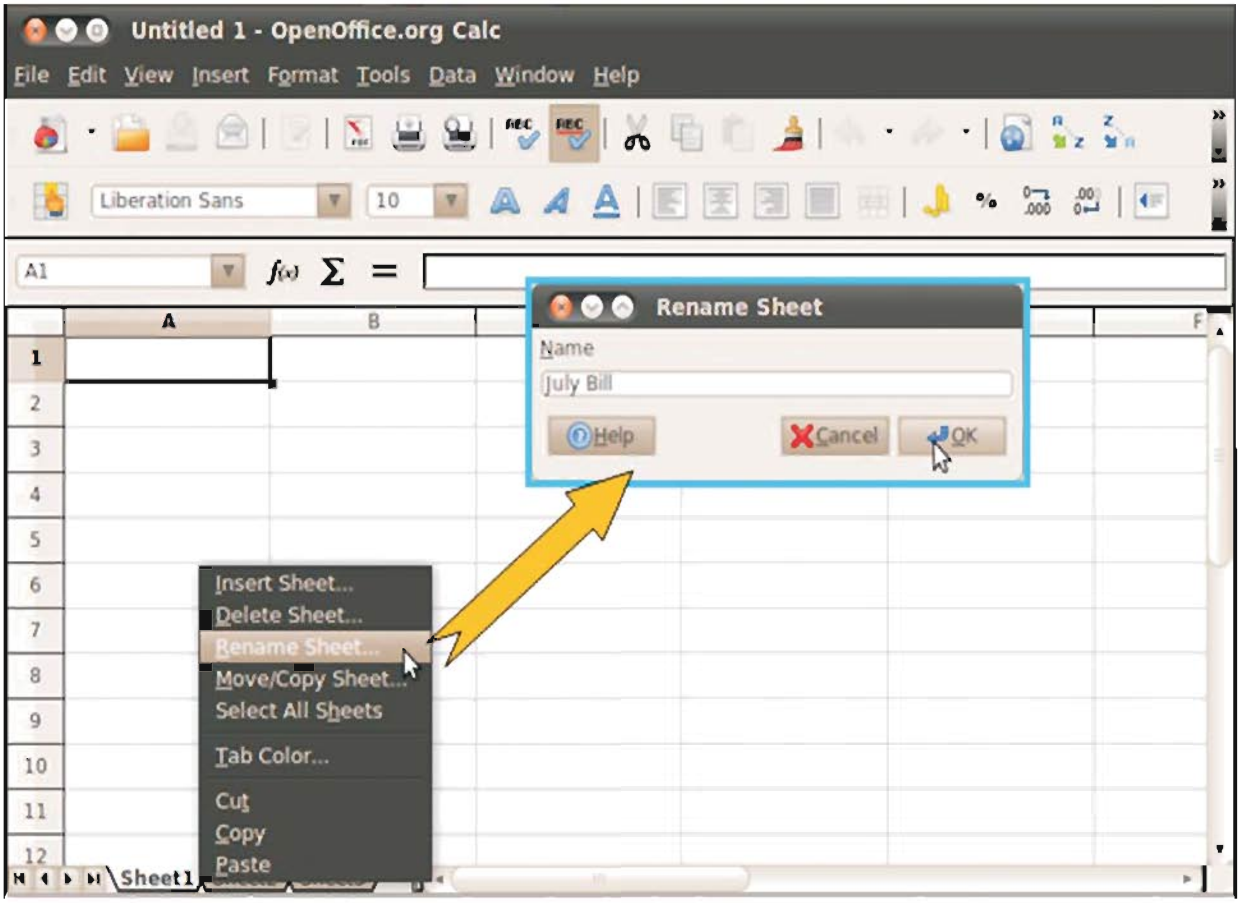
વર્કશીટનું નામ અને રંગ બદલવા (Renaming and re-colouring worksheet)

આપણે અગાઉના પ્રકરણમાં સ્ટેટસ બાર ઉપર રાઈટ ક્લિક કરી વર્કશીટ પસંદ કરીને વર્કશીટનું નામ બદલ્યું હતું. જો સ્પ્રેડશીટ દસ્તાવેજમાં અનેક વર્કશીટ તમારી પાસે હોય તો તે દસ્તાવેજના સરળ સંચાલન માટે તેને યોગ્ય નામ (અને રંગ) આપવા વધુ સારા છે. વર્કશીટનું નામ બદલવા માટે નીચે પ્રમાણે ક્રિયાઓ કરો :

* ખુલ્લી વર્કશીટમાંથી કોઈ પણ એક સેલની પસંદગી કરો.
* હવે આદેશ આપો : **Format → Sheet → Rename**
* પ્રદર્શિત **Rename Sheet** ડાયલૉગ બૉક્સમાં નવું નામ દાખલ કરો.
* **OK** બટન દબાવીને પરિવર્તનોનો સંગ્રહ કરો.

આના અન્ય વિકલ્પરૂપે શીટ ટૅબ ઉપર દર્શાવેલાં વર્કશીટનાં નામ ઉપર રાઇટ ક્લિક કરીને નીચે દર્શાવ્યા પ્રમાણે Rename sheet વિકલ્પ પસંદ કરો. આ વિકલ્પ વિશે તમે અગાઉ શીખ્યા છો. આ ક્રિયા નીચે જણાવેલ છે :

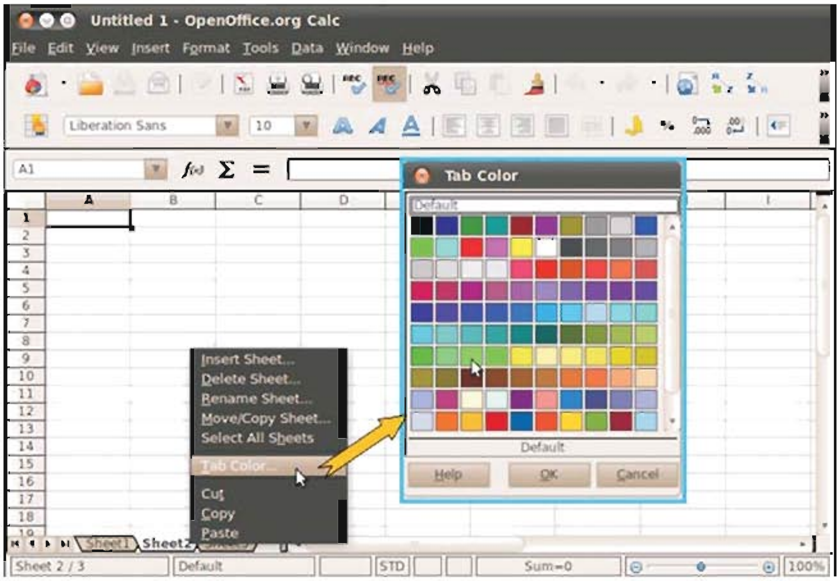
* તમારે જે વર્કશીટનું નામ બદલવું હોય તે શીટ પસંદ કરો. પસંદ કરેલી શીટ હાઇલાઇટ થયેલી દેખાશે (સફેદ પશ્ચાદ્ભૂમિ સાથે)
* તે શીટ ઉપર રાઇટ ક્લિક કરો. આકૃતિ 6.5માં દર્શાવ્યા પ્રમાણે એક ઊભું મેનૂ પ્રદર્શિત થશે તેમાંથી Rename sheet વિકલ્પ પસંદ કરો, આથી આકૃતિ 6.5ની મધ્યમાં લંબચોરસ આકારનું એક નવું ડાયલૉગ બૉક્સ દેખાય છે.
* હવે શીટને નવું નામ આપો.



આકૃતિ 6.5 : શીટ ટૅબ વડે વર્કશીટનું નામ બદલવું

વર્કશીટ ટૅબનો રંગ બદલવો (Changing colour of the worksheet tab)

વર્કશીટનાં ટૅબનો રંગ બદલવા માટે આકૃતિ 6.6માં દર્શાવ્યા પ્રમાણે આપણે ફક્ત વર્કશીટના નામ ઉપર રાઇટ ક્લિક કરીને સ્ક્રીન ઉપર પ્રદર્શિત ઊભા મેનૂમાંથી **Tab color** આદેશ પસંદ કરવો પડે છે. આપણી સામે રજૂ કરેલા વિવિધ રંગોમાંથી આપણે ટૅબ માટે યોગ્ય રંગ પસંદ કરવો પડે છે.



આકૃતિ 6.6 : વર્કશીટ ટૅબનો રંગ બદલવો

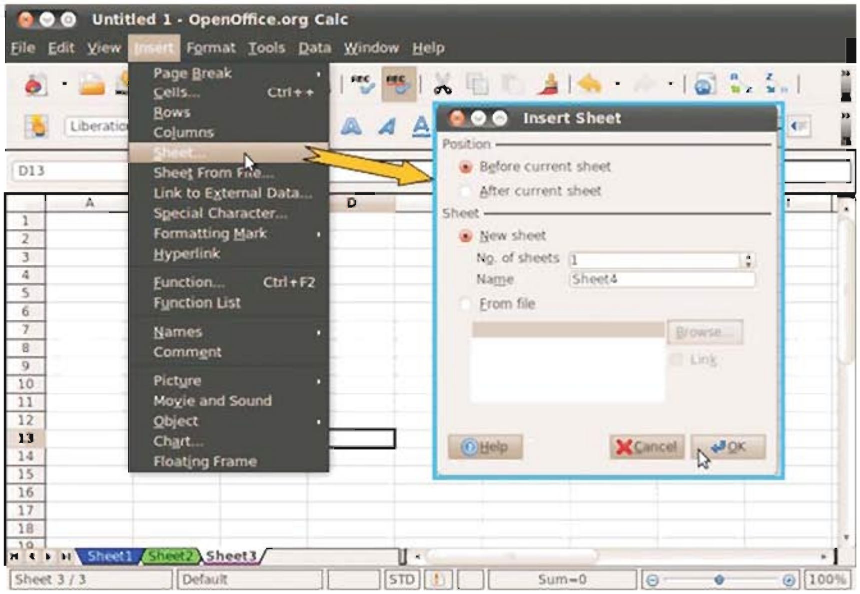
વર્કશીટ ઉમેરવી અને દૂર કરવી (Inserting and deleting worksheets)

તમે જાણતા હશો કે ઘણીવાર અમુક વિનિયોગ સંબંધિત બધો ડેટા દાખલ કરવા માટે કેલ્સી દ્વારા પૂરી પાડવામાં આવતી ત્રણ વર્કશીટ પૂરતી નથી. આવા સંજોગોમાં, આપણે વધારાની વર્કશીટ ઉમેરવી પડે છે. દા.ત. કોઈ પ્રવૃત્તિ, જેમ કે રમતગમતમાં વિદ્યાર્થીઓનાં પાંચ જૂથ છે. દરેક જૂથ માટે એક હાજરીપત્રક બનાવવું છે. આ ઉદાહરણમાં પાંચ વર્કશીટ હોય તે વધુ ઇચ્છનીય છે; દરેક વર્કશીટ જૂથદીઠ અલગ અલગ હાજરીપત્રક હોય. ફૉર્મેટ કરેલી સ્પ્રેડશીટમાં રો અને કૉલમ ઉમેરી શકાય છે, પહોળાઈ બદલી શકાય છે અને સેલને ભેગા કરી શકાય છે.

નવી વર્કશીટ ઉમેરવા માટે નીચે પ્રમાણે ક્રિયાઓ કરો :

- એ વર્કશીટનો કોઈ પણ સેલ પસંદ કરો કે જે વર્કશીટની આગળ કે પાછળ નવી વર્કશીટ ઉમેરવાની હોય.
- હવે આદેશ આપો: **Insert → Sheet**. આથી **Insert Sheet** ડાયલૉગ બૉક્સ સ્ક્રીન ઉપર પ્રદર્શિત થશે.
- આ ઇન્સર્ટ શીટ (Insert Sheet) ડાયલૉગ બૉક્સમાં વધારાના વિકલ્પો જણાવો.
- અંતમાં, આ ક્રિયાને મંજૂર કરાવવા માટે **OK** બટન ઉપર ક્લિક કરો.

વધારાની નવી વર્કશીટ ઉમેરવા માટેની ઉપર જણાવેલી ક્રિયાઓ આકૃતિ 6.7માં સચિત્ર દર્શાવેલી છે.



આકૃતિ 6.7 : દસ્તાવેજમાં નવી વર્કશીટ ઉમેરવી

તમે કદાચ વર્કશીટ ટૅબ મારફતે નવી વર્કશીટ ઉમેરવાની શક્યતા બાબતે પણ વિચાર્યું હશે. શીટ ટૅબનો ઉપયોગ કરીને વર્કશીટ ઉમેરવા માટે નીચે જણાવેલાં પગલાંને અનુસરો :

- વર્કશીટ પસંદ કરીને તેના ઉપર રાઇટ ક્લિક કરો.
- એક ઊભું મેનૂ ખૂલશે.
- હવે **Insert Sheet** પસંદ કરો.
- તમે Insert Sheet ડાયલૉગ બૉક્સ જોઈ શકશો.
- જરૂરી વિકલ્પો આપીને OK બટન ઉપર ક્લિક કરો.

વર્કશીટ દૂર કરવી (Deleting worksheet)

કૅલ્સીમાં તમે એક અથવા વધારે વર્કશીટ દૂર કરી શકો છો. કોઈ એક વર્કશીટ દૂર કરવા માટે તમે જે દૂર કરવા ઇચ્છતા હોય તે વર્કશીટના શીટ ટૅબ ઉપર રાઇટ ક્લિક કરો. તે તમને શક્ય ક્રિયાઓની યાદી જણાવશે. આ વિકલ્પોમાંથી **Delete Sheet** પસંદ કરો. આના વિકલ્પરૂપે, તમે આ પ્રમાણે આદેશ આપો : **Edit → Sheet → Delete**

રો અને કૉલમ કક્ષાની ક્રિયાઓ (Row and Column level operations)

અમુક સમયે આપણે વર્કશીટમાં કેટલીક કૉલમ કે રો ઉમેરવાની જરૂર પડે છે. જો કે વર્કશીટ ઘણી રો અને કૉલમ પૂરી પાડે છે પણ એ શક્ય છે કે આપણે દાખલ કરેલા ડેટાની વચ્ચે કેટલીક ખાલી રો કે કૉલમ ઉમેરવા ઈચ્છીએ. આપણે આ વિભાગમાં રો અને કૉલમ કક્ષાની અન્ય ક્રિયાઓ સાથે વર્કશીટમાં રો અને કૉલમ કઈ રીતે ઉમેરી શકીએ તેની ચર્ચા કરીશું. વર્કશીટમાં રો (આડી હરોળ) ઉમેરવા માટે નીચે પ્રમાણે કાર્ય કરો :

- જે રો ની ઉપર નવી રો ઉમેરવાની હોય તે રો માંથી કોઈ એક સેલ પસંદ કરો.
- હવે આદેશ આપો : **Insert → Rows**

વર્કશીટમાં કૉલમ (ઊભી હરોળ) ઉમેરવા માટે નીચે પ્રમાણે કાર્ય કરો :

- જે કૉલમની ડાબી બાજુમાં નવી કૉલમ ઉમેરવાની હોય ને કૉલમમાંથી એક સેલ પસંદ કરો.
- હવે આદેશ આપો : **Insert → Column**

કૉલમની પહોળાઈ અને રો ની ઊંચાઈ બદલવી (Width of a column and height of a row)

કૅલ્સીની બધી જ રો (આડી હરોળ) અને કૉલમ (ઊભી હરોળ) એકસરખી ઊંચાઈ અને પહોળાઈ ધરાવે છે. જ્યારે તમે સેલમાં શાબ્દિક માહિતી લખો છો ત્યારે ઘણી વખત બાજુના સેલની માહિતીને કારણે સેલની માહિતી અદશ્ય થઈ જાય છે. અમુક સમયે કેટલાંક સૂત્રોનું પરિણામ સેલમાં સમાતું પણ નથી. કૉલમની પહોળાઈ અથવા રોની ઊંચાઈ બદલવા માટે નીચે પ્રમાણે કાર્ય કરો :

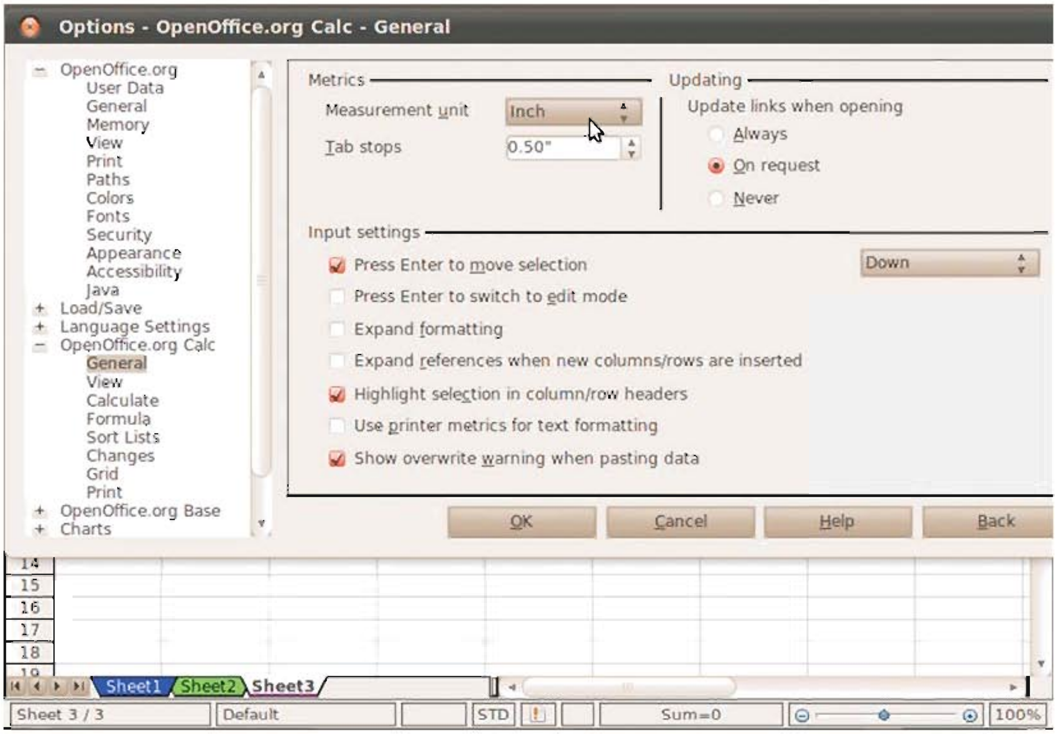
- કૉલમ(રો)નાં શીર્ષકોને જુદી પાડતી રેખા ઉપર ક્લિક કરો.
- માઉસનું ડાબું બટન છોડી દીધા વિના તમે જરૂરી દિશામાં ડ્રેગ કરો (ખેંચો).

કૉલમ અથવા રોને કોઈ ચોક્કસ માપ આપવા માટે નીચે પ્રમાણે કાર્ય કરો :

- કૉલમ(રો)માંથી કોઈ પણ એક સેલ પસંદ કરો.
- હવે આદેશ આપો : **Format → Column → Width** અથવા **Format → Row → Height**
- સ્ક્રીન ઉપર જે ડાયલૉગ બૉક્સ પ્રદર્શિત થાય તેમાં કૉલમની પહોળાઈ અથવા રો ની ઊંચાઈ ઈંચમાં જણાવો.
- હવે **OK** બટન દબાવો.

જો તમને ઈંચમાં માપ આપવાનું અનુકૂળ ન હોય તો નીચે પ્રમાણે આદેશ આપો :

Tools → Options → General આકૃતિ 6.8માં દર્શાવ્યા પ્રમાણે તમારી જરૂરિયાત મુજબ તે વર્ગ(General વર્ગ)માં પૂરા પાડવામાં આવેલાં વિકલ્પો બદલો.



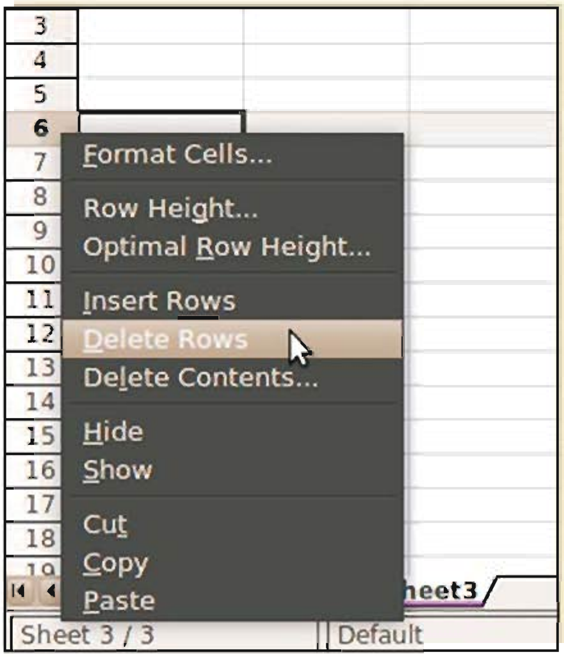
આકૃતિ 6.8 : કેલ્સીમાં General ના વિકલ્પો

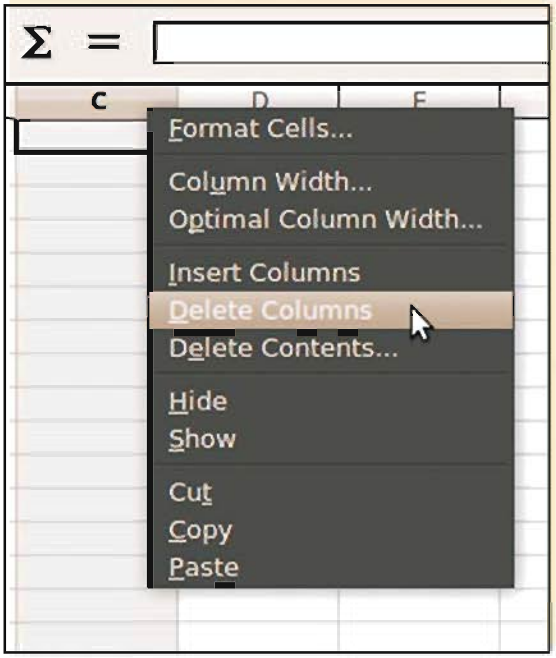
રો અને કૉલમ દૂર કરવી (Deleting rows and columns)

રો દૂર કરવા માટે નીચે જણાવેલાં પગલાં ભરો :

- તમે જે રો દૂર કરવા ઇચ્છતા હોય તે રો ઉપર તમે જાઓ.
- તે રો ઉપર રાઇટ ક્લિક કરો.
- હવે **Delete rows** પસંદ કરો.

આ જ રીતે તમે કૉલમ પણ દૂર કરી શકો છો. આકૃતિ 6.9માં રો અને કૉલમને દૂર કરવાની પ્રક્રિયાનું પ્રદર્શન કરેલું છે.





આકૃતિ 6.9 : રો અને કૉલમને દૂર કરવી

અન્ય વિકલ્પરૂપે, રો કે કૉલમ દૂર કરવા માટે નીચે જણાવેલાં પગલાંઓને અનુસરો :

- સૌપ્રથમ કૉલમ અથવા રો પસંદ કરો.
- **Edit → Delete Cells** પસંદ કરો.

આ ક્રિયા આકૃતિ 6.10માં સચિત્ર દર્શાવેલી છે. આ વિકલ્પ વડે આપણી જરૂરિયાત પ્રમાણે સેલને ખસેડીને આખી કૉલમ અથવા રો ને દૂર કરવી શક્ય છે.

રો અને કૉલમને પસંદ કરવી (Selecting rows and columns)

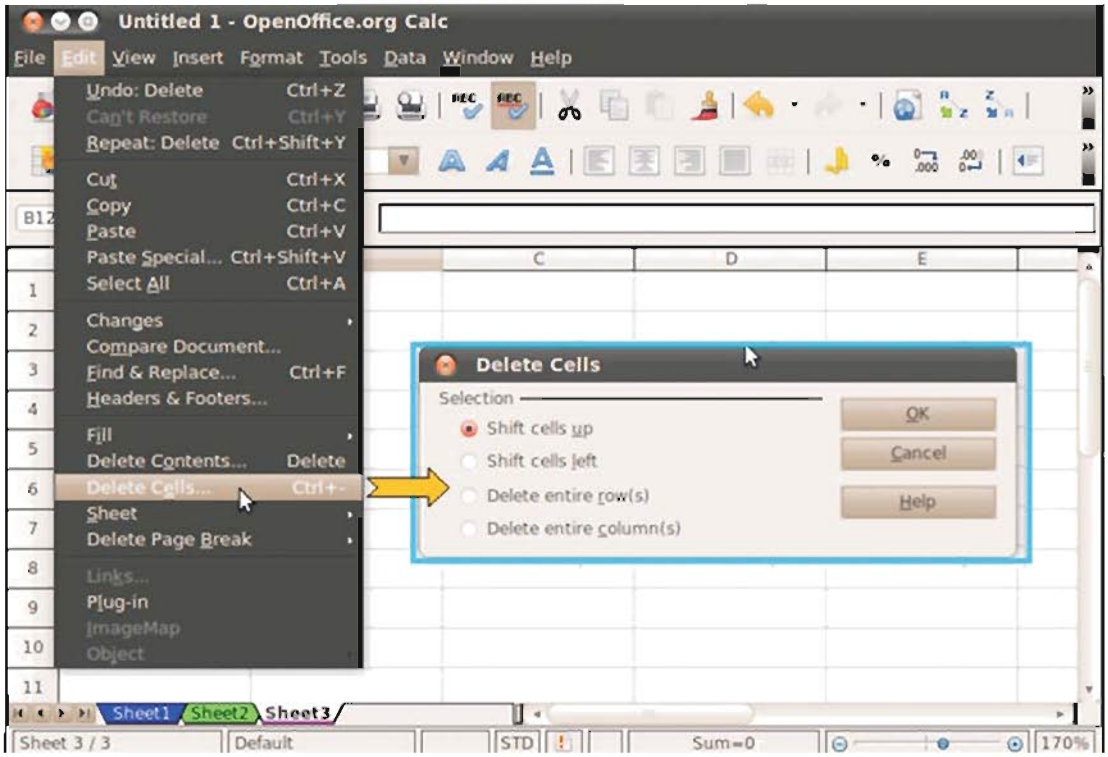
આખી ઊભી હરોળને પસંદ કરવા માટે કૉલમ હેડર (column header) ઉપર ક્લિક કરો, તમે જોઈ શકશો કે આખી ઊભી હરોળ હાઇલાઇટ થયેલી છે. આ જ પ્રમાણે, આખી આડી હરોળને પસંદ કરવા માટે રો હેડર (row header) ઉપર ક્લિક કરો અને આથી આખી આડી હરોળ હાઇલાઇટ થયેલી જોવા મળશે.

વિભાગ સ્થગિત બનાવવો (Freezing a pane)

જ્યારે એક જ વર્કશીટમાં સંખ્યાબંધ રો અને કૉલમનો સમાવેશ કરવામાં આવે છે ત્યારે કૉલમ હેડર અથવા રો હેડર સ્ક્રીન ઉપર દેખાતું રહેતું નથી કે જેથી ડેટાનું સંચાલન મુશ્કેલ બની જાય છે. કૉલમ હેડર અને રો હેડર હંમેશાં દૃશ્યમાન રહે તે માટે નીચે મુજબ કાર્ય કરો :

- બધા કૉલમ હેડર ધરાવતી આડી હરોળની નીચેની રોમાંથી કોઈ એક સેલ પસંદ કરો. કૉલમના કિસ્સામાં જે કૉલમ હંમેશાં દૃશ્યમાન રાખવી હોય તેની જમણી બાજુની કૉલમ પસંદ કરો.
- હવે પસંદ કરો : **Window → Freeze**

સેલને ચલિત (unfreeze) બનાવવા માટે **Window → Freeze** આદેશ ફરી આપીને Freeze વિકલ્પ નાપસંદ કરો.



આકૃતિ 6.10 : સેલને દૂર કરવા

સેલ કક્ષાની ક્રિયાઓ (Cell Level Operations)

આ વિભાગ તમને સેલમાં કઈ રીતે માહિતી લખી શકાય તેની સમજ આપશે. આ માહિતી શાબ્દિક લખાણ, સંખ્યા, વિધેયો, સૂત્રો કે અન્ય સેલનો સંદર્ભ હોઈ શકે. આ ઉપરાંત તમે એક સેલ કે અનેક સેલની માહિતીની નકલ કેવી રીતે કરવી અને કેવી રીતે ખસેડવી તે બાબત પણ શીખશો.

સેલ પસંદ કરવો (Selecting cells)

ઉપર જણાવ્યા પ્રમાણે સેલમાં શાબ્દિક લખાણ, સંખ્યા, વિધેયો, સૂત્રો કે અન્ય સેલનો સંદર્ભ દાખલ કરી શકાય. પણ સેલમાં કંઈ પણ દાખલ કરતાં પહેલાં આપણે સેલને પસંદ કરવો પડે. નહિતર ઈચ્છિત સેલમાં લખાણ ઉમેરાશે નહીં. સેલને પસંદ કરવા માટે ફક્ત તમારે તેના ઉપર લેફ્ટ ક્લિક કરવું પડશે. તે પછી સેલમાં તમે લખાણ કરી શકશો.

સેલનો વિસ્તાર પસંદ કરવો (Selecting a range of Cells)

કોઈ સેલ વિસ્તાર પસંદ કરવા માટે તે વિસ્તારના પહેલા સેલ ઉપર જઈ લેફ્ટ ક્લિક કરો. માઉસને લેફ્ટ ક્લિક કરીને પહેલાં સેલ પસંદ કર્યા પછી માઉસનું બટન છોડી દીધા વિના માઉસ પૉઈન્ટરને ઈચ્છિત વિસ્તારના છેલ્લા સેલ સુધીના બધા સેલનો સમાવેશ થાય ત્યાં સુધી ડ્રેગ કરો. એક સમયે વિસ્તારના બધા સેલ હાઈલાઈટ થઈ જાય પછી માઉસ બટનને છોડી દો.

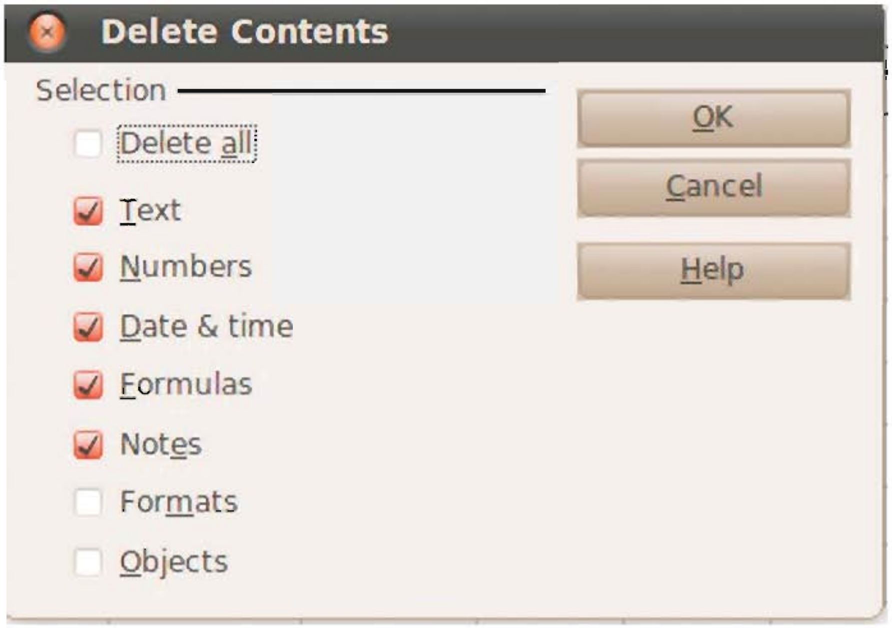
ઉપર જણાવેલી પ્રક્રિયા ફક્ત એકબીજાની બાજુમાં રહેલા સેલને લાગુ પડે છે. જો સેલ ભૌતિક રીતે એકબીજાને અડીને ન હોય તો તમારે **CTRL** કી દબાવીને રાખવી પડે છે. જો તમારે આખી વર્કશીટ (વર્કશીટના બધા સેલ) પસંદ કરવી હોય તો તમારે રો હેડર અને કૉલમ હેડરની શરૂઆતનું બટન જ ફક્ત ક્લિક કરવું પડે છે. રો અને કૉલમના નિર્દેશો આપીને પણ તમે વિસ્તાર પસંદ કરી શકો છો જેમ કે ઍડ્રેસ બૉક્સમાં A1:B12 જણાવીને.

કોઈ એક સેલ અથવા સેલના વિસ્તારની પસંદગી રદ કરવા માટે તમારે ફક્ત માઉસ વડે કોઈ પણ સેલ ઉપર લેફ્ટ ક્લિક કરવું પડે છે.

સેલની માહિતી રદ કરવી (To delete the content of a cell)

કોઈ એક સેલ કે સેલના વિસ્તારમાં રહેલી માહિતી દૂર કરવા માટે નીચે પ્રમાણે કાર્ય કરો :

- એક સેલ અથવા સેલનો વિસ્તાર પસંદ કરો.
- કી બોર્ડ ઉપરની ડિલીટ કી **(Delete)** દબાવો આથી સ્ક્રીન ઉપર એક ડાયલૉગ બૉક્સ પ્રદર્શિત થશે.
- ક્લિક કરીને તમારી અલગ અલગ પસંદ જણાવો. જો તમે શાબ્દિક લખાણ રદ કરવા ઈચ્છતા હોય તો **Text** ઉપર ક્લિક કરો. તમે એક કરતાં વધારે પસંદગી આપી શકો છો.
- આકૃતિ 6.11માં દર્શાવ્યા પ્રમાણે **Delete Contents** ડાયલૉગ બૉક્સનાં **OK** બટન ઉપર ક્લિક કરીને તમારી પસંદગીની પુષ્ટિ કરો.



આકૃતિ 6.11 : સેલની માહિતી રદ કરવાના વિવિધ વિકલ્પો

સ્ક્રીન ઉપર પ્રદર્શિત ડાયલૉગ બૉક્સ રો અને કૉલમ રદ કરવાના વિવિધ વિકલ્પો રજૂ કરે છે. કોષ્ટક 6.2માં **Delete Contents** ડાયલૉગ બૉક્સમાં ઉપલબ્ધ વિકલ્પોની યાદી રજૂ કરેલી છે.

વિકલ્પ	વર્ણન
Delete all	સેલ / સેલના વિસ્તારની સંપૂર્ણ માહિતી રદ કરે છે.
Text	પસંદ કરેલા સેલ / સેલ વિસ્તારનું શાબ્દિક લખાણ રદ કરે છે.
Numbers	પસંદ કરેલા સેલ / સેલ વિસ્તારમાંથી સંખ્યા રદ કરે છે.
Date & time	પસંદ કરેલા સેલ / સેલ વિસ્તારમાંથી ફક્ત તારીખ અને સમય રદ કરે છે.
Formulas	પસંદ કરેલા સેલ / સેલ વિસ્તારમાંથી સૂત્રો અને તેનાં પરિણામો રદ કરે છે.
Notes	જો કોઈ સેલ માટે નોંધ ઉમેરી હોય તો રદ કરે છે.
Formats	સેલ ફૉર્મેટિંગ (માળખું) રદ કરે છે પણ સેલનો ડેટા રાખે છે.
Objects	સેલનાં વધારાનાં તત્ત્વો રદ કરે છે - જેમ કે ઇમેજ.

કોષ્ટક 6.2 : Delete Contents ડાયલૉગ બૉક્સના ઉપલબ્ધ વિકલ્પો

સેલ માહિતીમાં સુધારા-વધારા કરવા (Editing the cell content)

સેલમાં રહેલી માહિતીમાં ફેરફાર કરવા (એડિટ - edit) માટે ફક્ત સેલને પસંદ કરો અને માઉસ વડે ડબલ ક્લિક કરીને માહિતીમાં સુધારા કરો. જ્યારે સુધારા કરવાનું કાર્ય પૂર્ણ થાય તે પછી એન્ટર કી દબાવો.

સેલની નકલ કરવી અને તેને ખસેડવા (Copying and moving cells)

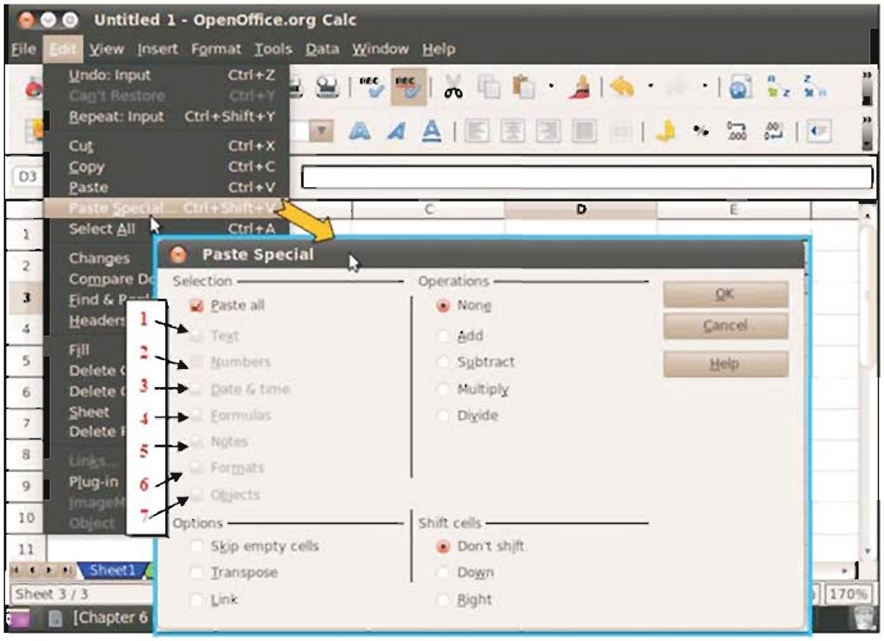
સ્પ્રેડશીટના સેલમાં અનેક પ્રકારનો ડેટા હોય છે, જેમ કે સંખ્યા, શાબ્દિક લખાણ, સૂત્રો, ફૉર્મેટ (માળખું) અને અન્ય સેલનો નિર્દેશ. આ રીતે સેલની માહિતી જટિલ છે, કારણ કે તે અનેક પ્રકારની કિંમતો ધરાવે છે. કોઈ એક સેલ, વર્કશીટ કે સેલ વિસ્તારની નકલ કરવા માટે નીચે પ્રમાણે કાર્ય કરો :

- તમે જેની નકલ કરવા ઇચ્છતા હોય તે સેલ / સેલનો ભાગ / સેલનો વિસ્તાર / વર્કશીટને પસંદ કરો.
- આદેશ આપો : **Edit → Copy** આના વિકલ્પે તમે સેલ પસંદ કરીને તેના ઉપર રાઇટ ક્લિક પણ કરી શકો છો અને તે પછી મેનૂમાંથી **Copy** વિકલ્પ આપો. નકલ થયેલો ડેટા ટપકાં ટપકાંથી બનેલા લંબચોરસમાં હાઇલાઇટ થયેલો દેખાશે.
- જે જગ્યાએ ડેટાની નકલ કરવાની હોય તે જગ્યા નક્કી કરો અને **Paste** આદેશ આપો.

Paste આદેશનાં વધારાના વિકલ્પો માટે આકૃતિ 6.12માં દર્શાવ્યા પ્રમાણે **Edit → Paste Special** આદેશ આપો. Paste Special ડાયલૉગ બૉક્સ જે માહિતીની નકલ કરવાની છે તેના માટે અનેક પરિમાણો (parameters) ધરાવે છે. આ પસંદગીઓની યાદી કોષ્ટક 6.3માં આપેલી છે.

લેબલ	વર્ણન
1	ફક્ત શાબ્દિક લખાણની નકલ (paste) કરે છે.
2	ફક્ત સંખ્યાની નકલ (paste) કરે છે.
3	ફક્ત તારીખ અને સમયની નકલ (paste) કરે છે.
4	ફક્ત સૂત્રોની નકલ (paste) કરે છે.
5	ફક્ત નોંધની નકલ (paste) કરે છે.
6	કોઈ કિંમત અથવા સેલનું ફૉર્મેટની નકલ (paste) કરે છે.
7	ફક્ત ઑબ્જેક્ટની નકલ (paste) કરે છે.

કોષ્ટક 6.3 : Paste Specialની પસંદગીઓ



આકૃતિ 6.12 : Paste Special ના વિકલ્પો

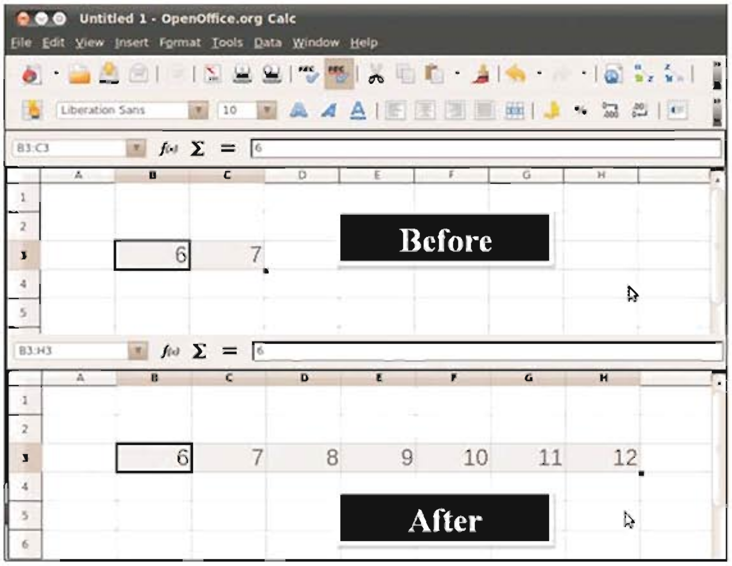
Paste Special ક્રિયા વડે તમે ગંતવ્ય સેલ ઉપર માહિતી કેવી રીતે નકલ કરવી તેનું નિયંત્રણ કરી શકો છો.

ઑટોફિલ ટૂલ (Autofill tool)

સેલમાં આપમેળે ડેટા ભરવા માટે ઑટોફિલ ટૂલ વપરાય છે. નીચે જણાવેલાં પગલાં પ્રમાણે કાર્ય કરો :

- પ્રથમ સેલમાં સંખ્યા દાખલ કરો.
- તે પછીના સેલમાં (રો કે કૉલમ) ત્યારપછીની સંખ્યા દાખલ કરો.
- બંને સેલ પસંદ કરો.
- સેલનાં હેન્ડલ (handle) ઉપર લેફ્ટ ક્લિક કરો.
- માઉસ બટન દબાવેલું રાખીને ઇચ્છિત સેલ વિસ્તાર સુધી ડ્રૅગ (ખેંચો) કરો.
- હવે માઉસ બટન છોડી દો.

આકૃતિ 6.13માં દર્શાવ્યા પ્રમાણેનો સ્ક્રીન તમે જોઈ શકશો.



આકૃતિ 6.13 : આપમેળે ભરાયેલી સંખ્યા

ઉપર જણાવેલી આપમેળે ભરાયેલી સંખ્યા સમાંતર શ્રેણી તરીકે પણ ઓળખાય છે. હવે તમે નીચે જણાવ્યા પ્રમાણે અલગ રીતે સમાંતર શ્રેણીનો પ્રયત્ન કરો :

તમે સેલમાં એકી સંખ્યા (ધારો કે 1) લખો. તેની બાજુના સેલમાં ક્રમિક એકી સંખ્યા (આપણા ઉદાહરણમાં અહીં પછીની એકી સંખ્યા 3 છે) લખો. હવે બંને સેલને એકસાથે પસંદ કરી તે પછીના 10 સેલમાં ઑટોફિલ ટૂલ વડે ભરવા પ્રયત્ન કરો. હવે તમે આપમેળે ભરાયેલી સંખ્યાનું નિરીક્ષણ કરો. તે તમામ એકી સંખ્યા છે ?

હવે અન્ય સંખ્યાઓની શ્રેણી લઈને પ્રયોગ કરી જુઓ. જેમ કે :

- 10, 20, 30, ...
- 25, 50, 75,...
- 2, 4, 5,....

આ ટૂલ વાપરીને તમે સંખ્યાના કોષ્ટક (1*1=1, 2*1=2, ...) બનાવી શકો.

જો તમે ફક્ત એક જ સેલની સંખ્યા સાથે ઑટોફિલ ટૂલનો ઉપયોગ કરશો તો 1ના વધારા સાથેની સંખ્યાની યાદી મેળવશો.

ફક્ત વિચારો કે સેલમાં સંખ્યાની જગ્યાએ કોઈ સેલ ઍડ્રેસ કે સેલ નિર્દેશ હોય તો શું થાય ? સેલમાં કેટલાક સેલ ઍડ્રેસ લખો અને બાજુના કેટલાંક સેલમાં ઑટોફિલ કરવાનો પ્રયત્ન કરી જુઓ.

સાપેક્ષ સ્થાનાંક અને નિરપેક્ષ સ્થાનાંક (Relative and absolute address)

અગાઉના ઉદાહરણમાં આપણે એક સેલના સૂત્રની નકલ બીજા અનેક સેલમાં કરી હતી. જો કોઈ સેલમાં સૂત્ર લખેલું હોય અને તે સેલની નકલ આપણે બીજા સેલમાં કરીએ તો સૂત્રમાં સામેલ સેલનાં સ્થાનાંક(એડ્રેસ-address)ને કેલ્સી આપોઆપ બદલી નાખે છે. (પ્રકરણ 5માં ખરીદીના બિલનાં ઉદાહરણમાં આપણે તે જોયું હતું.)

સેલ A1માં સેલ ઍડ્રેસ = C1 છે. આનો અર્થ એ થાય કે સેલ A1 સેલ C1ની માહિતીનો ઉલ્લેખ કરે છે; એટલે કે સેલ A1થી ત્રીજા સેલનો. હવે જો તમે સેલ A1ની નકલ સેલ B1માં કરો તો સેલ B1 તેનાથી ત્રીજા સેલ એટલે કે D1નો નિર્દેશ કરશે. આવા પ્રકારની સ્થાનાંક(એડ્રેસ)ની પ્રક્રિયાને સાપેક્ષ કહેવામાં આવે છે. જ્યારે સાપેક્ષ સ્થાનાંકની નકલ એક સેલમાંથી બીજા સેલમાં કરવામાં આવે ત્યારે તે આપોઆપ બદલાઈ જાય છે. આ જાતની રચના ટાળવા માટે અને સ્થાનાંકને સ્થાયી સ્વરૂપ આપવા માટે આપણે =$A$1 (રો અને / અથવા કૉલમની પહેલાં ડૉલરનું ચિહ્ન ($) ઉમેરીને) જેવું સ્થાનાંક લખવું પડે. આ પ્રકારના સ્થાનાંકને નિરપેક્ષ સ્થાનાંક કહેવામાં આવે છે. જ્યારે સૂત્રની નકલ કરવામાં આવે કે ખસેડવામાં આવે ત્યારે નિરપેક્ષ સ્થાનાંકમાં કોઈ પરિવર્તન આવતું નથી. નિરપેક્ષ સ્થાનાંક હંમેશાં કોઈ ચોક્કસ સેલની કિંમતનો નિર્દેશ કરે છે. ઉદાહરણ તરીકે, કોઈ પદાવલિમાં સ્થાનાંક =$A$4 હંમેશાં સેલ A4ની કિંમત મેળવશે, જ્યારે પણ તેની નકલ કરવામાં આવે કે ખસેડવામાં આવે.

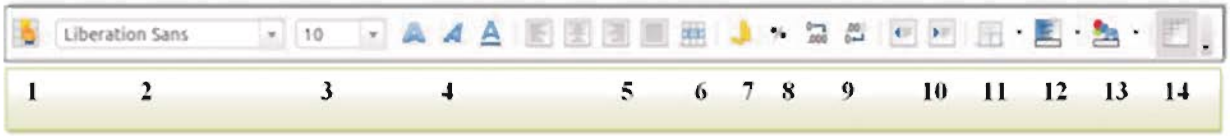
તમે કોઈ કૉલમની આગળ $ લખીને કૉલમને અચલ અને રો ને ચલિત અથવા એથી વિપરીત પણ કરી શકો. ઉદાહરણ તરીકે, જો તમે કોઈ સ્થાનાંક =A$4 તરીકે લખો અને તેને બીજી જગ્યાએ નકલ કરવાનો પ્રયત્ન કરો તો ફક્ત કૉલમ બદલાશે પણ રો નહીં બદલાય. તેમાં હંમેશાં રો 4 રહેશે.

સૂત્ર સાથેના સેલમાં ઑટોફિલ કરવા માટે નીચે પ્રમાણે કાર્ય કરો :

- સૂત્ર સાથેનો સેલ પસંદ કરો.
- સેલના હેન્ડલ ઉપર લેફ્ટ ક્લિક કરો.
- માઉસ બટનને દબાવી રાખીને ઇચ્છિત સેલ વિસ્તાર સુધી ડ્રેગ કરો.
- હવે માઉસ બટન છોડી દો.

સેલને ફૉર્મેટ કરવો (Formatting a Cell)

આપણે સેલના ગુણધર્મ બદલી શકીએ છીએ. આપણે સેલને રંગ લાગુ પાડી શકીએ, સેલની અંદર ઇમેજ ઉમેરી શકીએ, સેલને કિનારી આપી શકીએ અને સેલની અંદર રહેલ માહિતીના ફૉન્ટ પણ બદલી શકીએ છીએ. આપણે સેલની માહિતીને વિવિધ ફૉર્મેટ (format) લાગુ પાડી શકીએ જેમ કે તારીખ અને સંખ્યાનું ફૉર્મેટ બદલવું. એટલે કે, સંખ્યાને, અંદર રહેલ માહિતીને, ઍલાઇનમેન્ટ, સેલની કિનારી અને સેલની પશ્ચાદ્ભૂમિને ફૉર્મેટ કરી શકાય છે. આ કાર્ય માટે Format cells ડાયલૉગ બૉક્સ અથવા ફૉર્મેટિંગ ટૂલબારનો ઉપયોગ કરી શકીએ. આકૃતિ 6.14માં ફૉર્મેટિંગ ટૂલબાર દર્શાવ્યું છે.



આકૃતિ 6.14 : લેબલ સાથેનું ફૉર્મેટિંગ ટૂલબાર

આકૃતિ 6.14માં દર્શાવેલાં લેબલનું વર્ણન કોષ્ટક 6.4માં આપેલું છે.

લેબલ	વર્ણન
1	ઢબ (style) લાગુ પાડો
2	ફૉન્ટ
3	ફૉન્ટ સાઇઝ (ફૉન્ટનું કદ)
4	ઘાટા, ત્રાંસા (italic) અને શબ્દ નીચે લીટી દોરવી
5	ગોઠવણ (એલાઇમેન્ટ) : ડાબી બાજુ, મધ્યમાં, જમણી બાજુ, જસ્ટીફાય
6	પસંદ કરેલાં સેલ ભેળવવાં (મર્જ કરવા)
7	ચલણ
8	ટકા
9	દશાંશ ચિહ્ન ઉમેરવું / રદ કરવું
10	ઇન્ડેન્ટ વધારવું / ઘટાડવું
11	સેલની કિનારીનું ફૉર્મેટ
12	સેલની પશ્ચાદ્‌ભૂમિનો રંગ
13	સેલમાં ફૉન્ટનો રંગ
14	ફૉર્મેટ ન કરેલી સેલની કિનારી ઉમેરવી / રદ કરવી

ફૉર્મેટ 6.4 : ફૉર્મેટિંગ ટૂલબાર ઉપરના વિકલ્પો

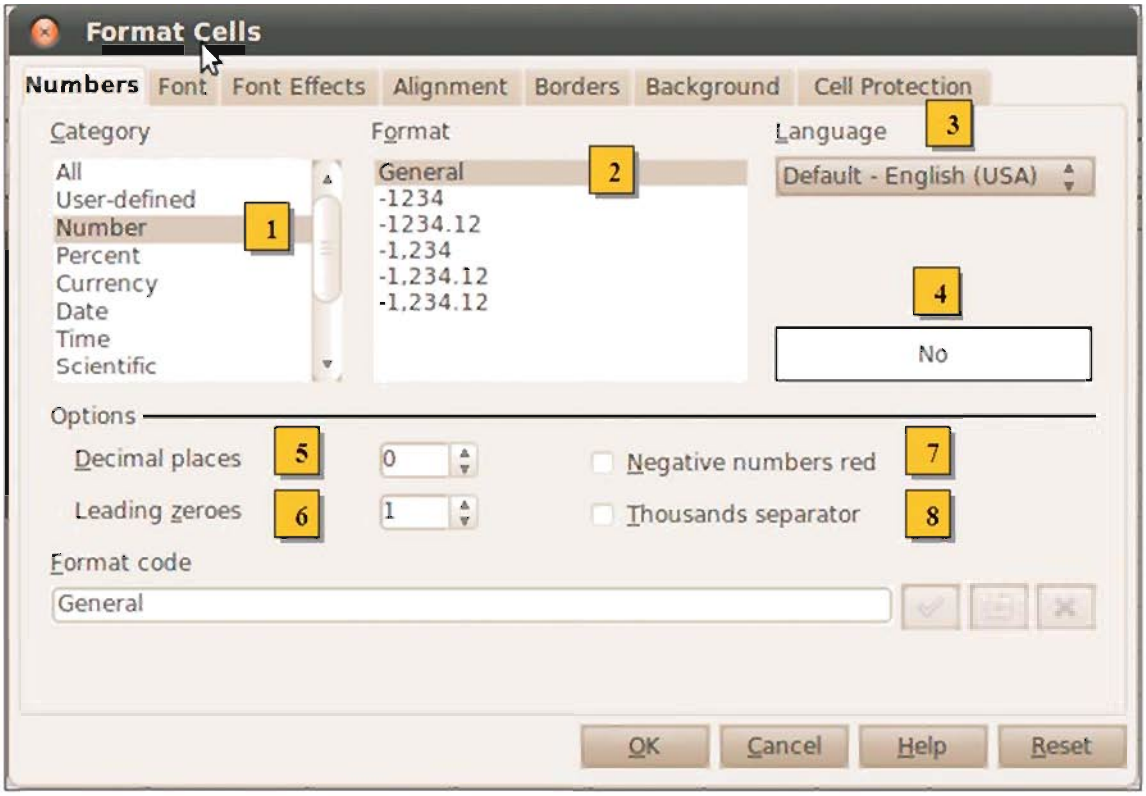
ચાલો, હવે આપણે ફૉર્મેટિંગની કેટલીક ઢબ જોઈએ. તે પછી તમે ઉપર જણાવેલી વિવિધ ફૉર્મેટિંગ સ્ટાઇલનો વર્કશીટના કેટલાક સેલ ઉપર પ્રયોગ કરી જુઓ.

નંબર ફૉર્મેટ (Number Format)

નંબર ફૉર્મેટ સેલમાં રહેલી સંખ્યાને અસર કરે છે. આ નંબર ફૉર્મેટ કોઈ એક સેલ, એક સાથે પસંદ કરેલા અનેક સેલ અને સેલના વિસ્તારને લાગુ પાડી શકાય છે. આકૃતિ 6.15માં દર્શાવ્યા પ્રમાણે Format cells પસંદ કરીને Numbers ટૅબ ખોલો અને સ્ક્રીન ઉપર પ્રદર્શિત ડાયલૉગ બૉક્સમાંથી તમારી પસંદગી પ્રમાણે ફૉર્મેટ સેટ કરો. આકૃતિ 6.15માં જે લેબલ આપેલાં છે તેની સમજ કોષ્ટક 6.5માં આપેલી છે.

લેબલ	વર્ણન
1.	યોગ્ય ફૉર્મેટની પસંદગી જેમ કે Numbers, dates વગેરે.
2.	પસંદ કરેલાં ફૉર્મેટનો લાક્ષણિક દેખાવ
3.	પ્રાદેશિક સેટિંગ્સ
4.	પસંદ કરેલા સેલનું પ્રિવ્યુ, હજાર માટે સેપરેટર (અલગ પાડવા માટેનું ચિહ્ન) ઉમેરવું
5.	દશાંશ ચિહ્ન પછી અંકની સંખ્યા
6.	શરૂઆતનાં શૂન્યની સંખ્યા
7.	ઋણ સંખ્યાને લાલ દર્શાવવી
8.	હજારમાં અલગ પાડવું

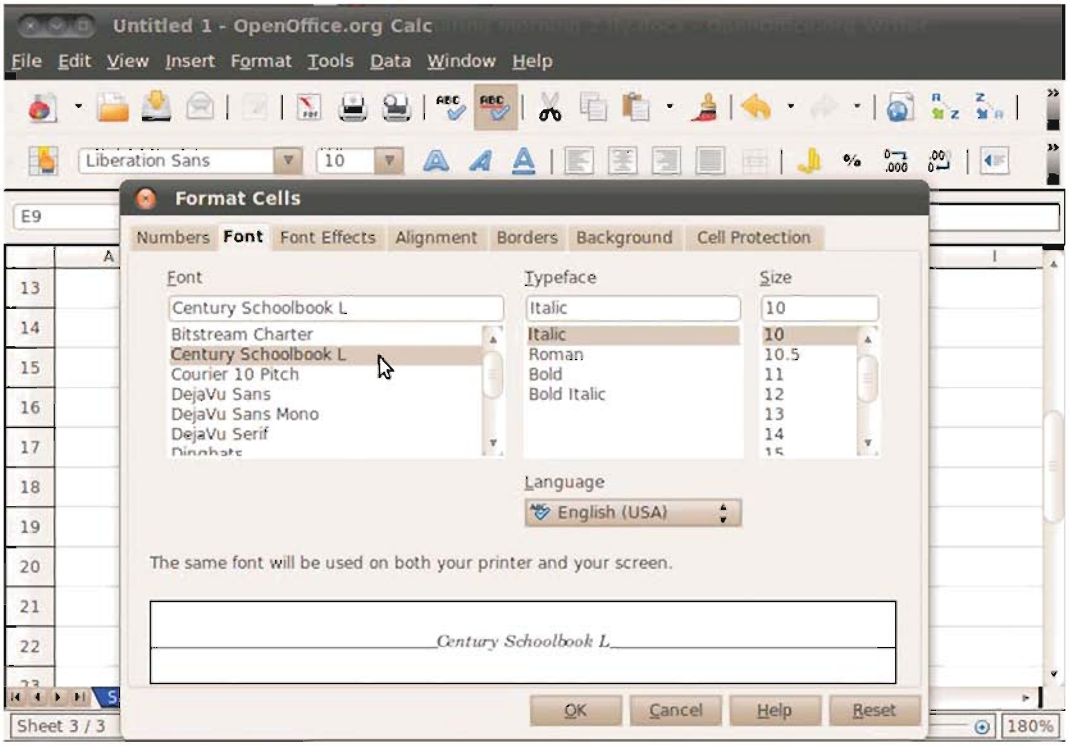
કોષ્ટક 6.5 : સેલ માટે Number ફૉર્મેટિંગના વિકલ્પો



આકૃતિ 6.15 : સેલ માટે Numbers ફૉર્મેટિંગના વિકલ્પો

સંખ્યા ધરાવતા સેલ માટેના Number ફૉર્મેટિંગના વિકલ્પો આકૃતિ 6.15માં દર્શાવ્યા છે. તારીખ, ચલણ કે ટકાનું ફૉર્મેટ બદલવા માટે યોગ્ય પસંદ કરવી જરૂરી છે.

Format Cells ડાયલૉગ બૉક્સના બીજા વિકલ્પ **Font** ટૅબ દ્વારા તમે ફૉન્ટ બદલી શકો છો. આકૃતિ 6.16માં દર્શાવ્યા પ્રમાણે **Font Effects** ટૅબ વડે ફૉન્ટની અસર બદલી શકાય છે.



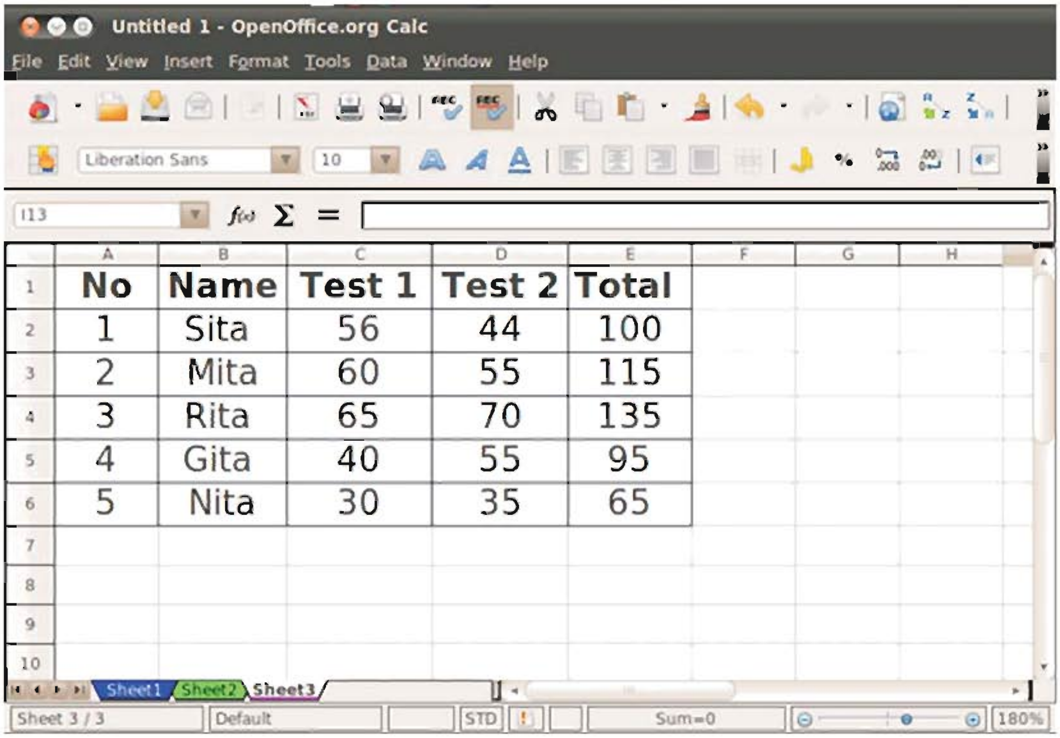
આકૃતિ 6.16 : ફૉન્ટ બદલવા

તમે કદાચ આમાંની કેટલીક ફૉન્ટની અસરો લાગુ પાડી પણ હોય જેમ કે ઘાટા (બોલ્ડ), ત્રાંસા (ઇટાલિક), શબ્દની નીચે લીટી (અન્ડરલાઇન) અથવા ફૉન્ટનું કદ. આમાંની કેટલીક અસર ફૉર્મેટિંગ ટૂલબાર ઉપર આઇકોન તરીકે આપેલી છે. ફક્ત યોગ્ય આઇકોન પસંદ કરીને આ અસરનો સીધો અમલ કરી શકાય છે.

ડેટાની ક્રમબદ્ધ ગોઠવણી અને નિસ્યંદન (ડેટા સૉર્ટિંગ અને ફિલ્ટરિંગ - Data Sorting and Filtering)

જો કોઈ ચોક્કસ ક્રમમાં ડેટા ગોઠવાયેલા હોય તો તેને સમજવા અને વાપરવા સરળ બને છે. ડેટાની કોઈ ચોક્કસ ક્રમમાં ગોઠવણીને સૉર્ટિંગ (sorting) કહેવામાં આવે છે. ડેટાને ચઢતા કે ઊતરતા ક્રમમાં ગોઠવવાની પ્રક્રિયા કેલ્સીમાં ઉપલબ્ધ છે. જો ડેટા એક જ કૉલમ કે રો માં હોય તો તે તમારી ઇચ્છિત રીતે સૉર્ટ થશે. પણ જો ડેટા એક કરતાં વધારે રો કે કૉલમમાં વિસ્તરેલો હોય તો બધી જ કૉલમ અને રો બદલાશે અને સૉર્ટિંગ થશે.

ઉદાહરણ તરીકે આપણે વિદ્યાર્થીઓના ગુણ લઈએ. ક્યા વિદ્યાર્થીએ સૌથી વધારે કુલ ગુણ મેળવ્યા છે તે શોધી કાઢવા માટે અને મેરિટ લિસ્ટ તૈયાર કરવા માટે આપણે ઊતરતા ક્રમમાં (સૌથી મોટી કિંમત પહેલી આવશે) સૉર્ટિંગ કરવું પડશે. આ અજમાયશ કરવા માટે એક નવી સ્પ્રેડશીટ ખોલો અને આકૃતિ 6.17માં દર્શાવ્યા પ્રમાણે ડેટા દાખલ કરો. તે પછી યોગ્ય નામ આપી ફાઈલનો સંગ્રહ કરો.



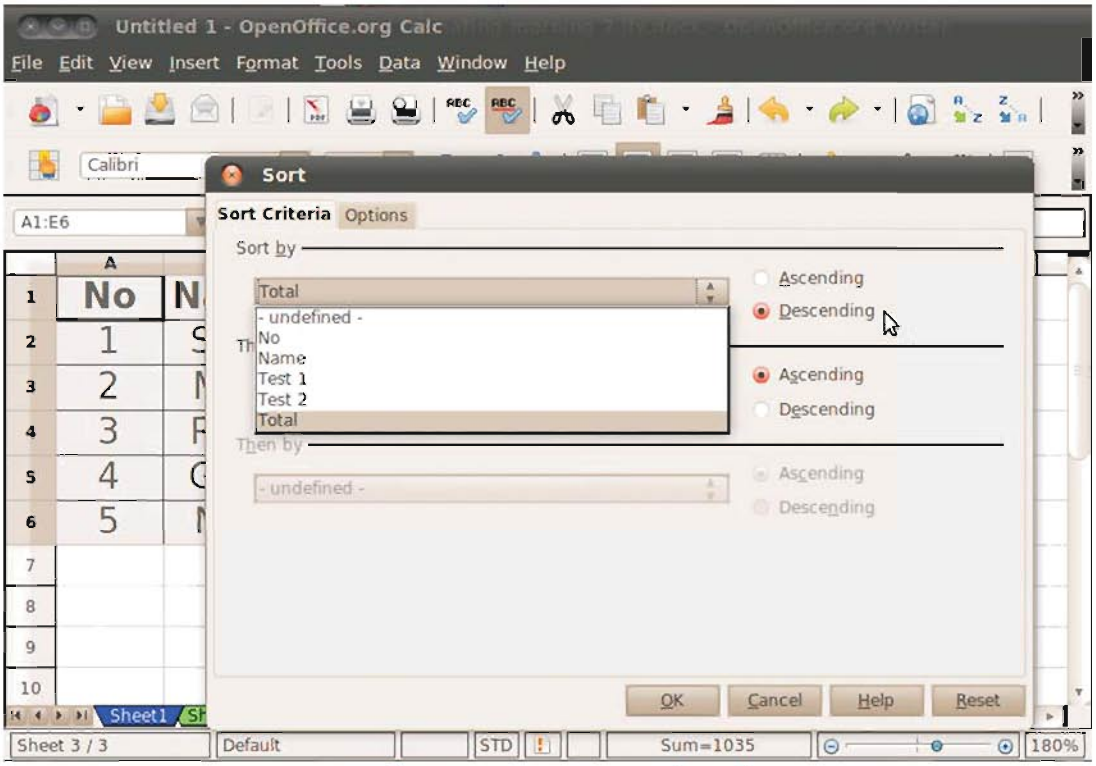
આકૃતિ 6.17 : વર્કશીટમાં દાખલ કરેલી માહિતી

આકૃતિ 6.17માં ઉપલબ્ધ ડેટાને કૉલમ Eની કિંમત, એટલે કે **'Total'** ઉપર સૉર્ટ કરવા માટે નીચે પ્રમાણે કાર્ય કરો :

- જે કૉલમના ડેટા ઉપર સૉર્ટ કરવાની જરૂર હોય તેનાં કોઈ પણ સેલ ઉપર ક્લિક કરો.
- આકૃતિ 6.8માં દર્શાવેલું **Formatting** ટૂલબાર ઉપરના **Sort descending** બટન ઉપર ક્લિક કરો.

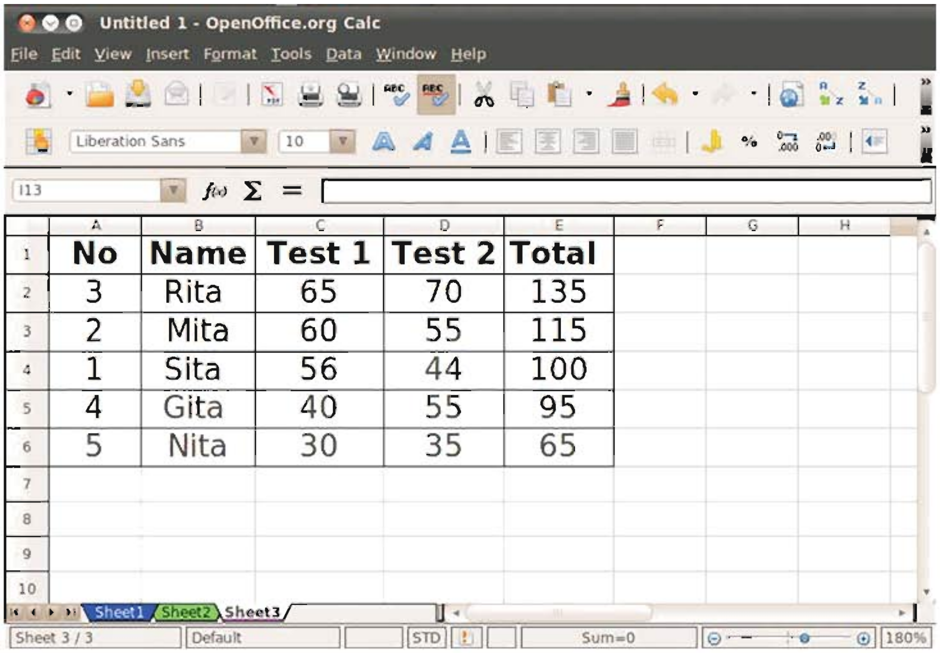
આકૃતિ 6.18 : ફૉર્મેટિંગ ટૂલબારનો ઉપયોગ કરીને સૉર્ટિંગ

આ જ ક્રિયા બીજી રીતે પણ થઈ શકે. આ માટે તમારે ફક્ત **Data → Sort** પસંદ કરવાનું છે. આથી એક ડાયલૉગ બૉક્સ પ્રદર્શિત થશે. ડાયલૉગ બૉક્સના sort criteriaમાં **Descending** (ઊતરતો ક્રમ) પસંદ કરો. આકૃતિ 6.19માં sort ડાયલૉગ બૉક્સનો એક લાક્ષણિક દેખાવ આપેલો છે.



આકૃતિ 6.19 : Sort ડાયલૉગ બૉક્સ

ક્રમબદ્ધ ગોઠવ્યા પછી તમે જોઈ શકશો કે કોલમમાં કુલ ગુણ **135** પહેલાં સ્થાન ઉપર આવેલ છે. સ્પષ્ટ રીતે આ ગુણ **Rita** નામના વિદ્યાર્થીએ મેળવેલા છે અને આથી વિદ્યાર્થીઓના નામની કોલમમાં તેનું નામ પ્રથમ આવવું જોઈએ. જો તમે પરિણામ તરફ નજર કરશો તો જણાશે કે રીટાના કુલ ગુણ સાથે તેનો અન્ય ડેટા પણ પહેલી હરોળમાં ખસેડાયો છે. આનો અર્થ એ થાય કે તમે કોઈ કૉલમ ઉપર સૉર્ટ કરો છો ત્યારે આકૃતિ 6.20માં દર્શાવ્યા પ્રમાણે ઘણી રો પણ સ્થાન બદલશે. આકૃતિ 6.20માં બદલાયેલાં રેકોર્ડ(ડેટા)ના ક્રમની નોંધ કરો.



આકૃતિ 6.20 : ઊતરતા ક્રમમાં ગોઠવાયેલો ડેટા

તમે સંખ્યા ઉપર જ ફક્ત સૉર્ટ નથી કરી શકતા પણ સોર્ટિંગ મૂળાક્ષરો આધારિત પણ કરી શકો છો. અંગ્રેજી ભાષાના શબ્દકોશનું ઉદાહરણ લો કે જેમાં શબ્દો કક્કાવાર ગોઠવાયેલાં હોય છે. આપણે હાલમાં જે વિદ્યાર્થીઓના ડેટા ઉપર પ્રયોગ કરી રહ્યા છીએ તેની ઉપર નજર ફેરવો અને તે ડેટાને વિદ્યાર્થીઓનાં નામના ક્રમમાં ગોઠવવાનો પ્રયત્ન કરો. ઉપર જણાવેલ ડેટાને વિદ્યાર્થીઓના નામના ચઢતા ક્રમમાં ગોઠવવા (સૉર્ટ કરવા) નીચે જણાવ્યા પ્રમાણે કાર્ય કરો :

- પસંદ કરો : **Data → Sort**
- હવે કૉલમનું નામ **Name** આપો.
- સૉર્ટ ક્રમ તપાસી જુઓ, પૂર્વનિર્ધારિત રીતે તે **Ascending** છે.

તમે ડેટા બે અથવા વધારે ક્ષેત્ર (ફિલ્ડ field) ઉપર પણ સૉર્ટ કરી શકો છો. એવી પરિસ્થિતિ વિચારો કે તમારી પાસે વિદ્યાર્થીઓનાં પૂર્ણ નામ, તેની જગ્યા અને તેનાં ગુણ જેવી માહિતી સાથેના વિદ્યાર્થીઓની યાદી છે. તમે ડેટાને પ્રથમ જગ્યા (location) અને પછી તેના નામ (name) ઉપર સૉર્ટ કરવા ઇચ્છો છો. નીચે આપેલું ઉદાહરણ લો.

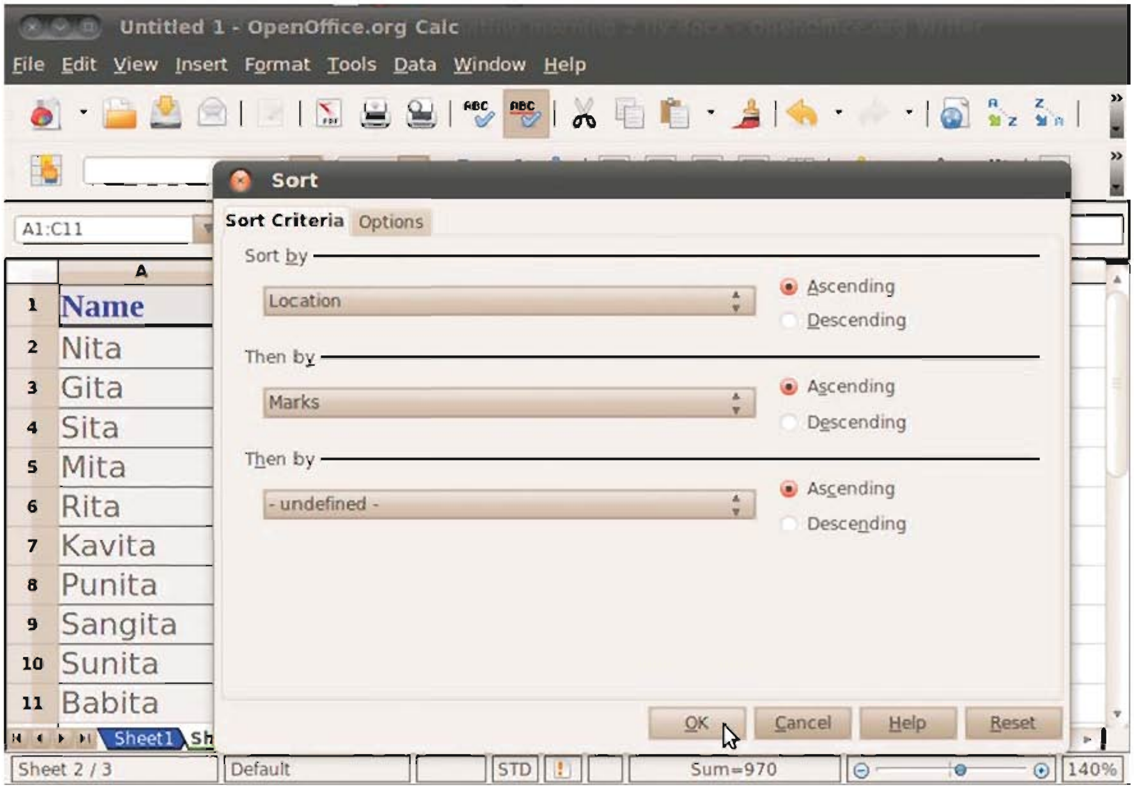
1. કોષ્ટક 6.6માં આપેલો ડેટા કેલ્સી વર્કશીટમાં દાખલ કરો.

Name	Location	Marks
Nita	Vallabh vidyanagar	65
Gita	Anand	95
Sita	Baroda	100
Mita	Ahmedabad	115
Rita	Gandhinagar	135
Kavita	Vallabh vidyanagar	110
Punita	Ahmedabad	105
Sangita	Anand	85
Sunita	Ahmedabad	70
Babita	Anand	90

કોષ્ટક 6.6 : સૉર્ટ કરવા માટે નમૂનાનો ડેટા

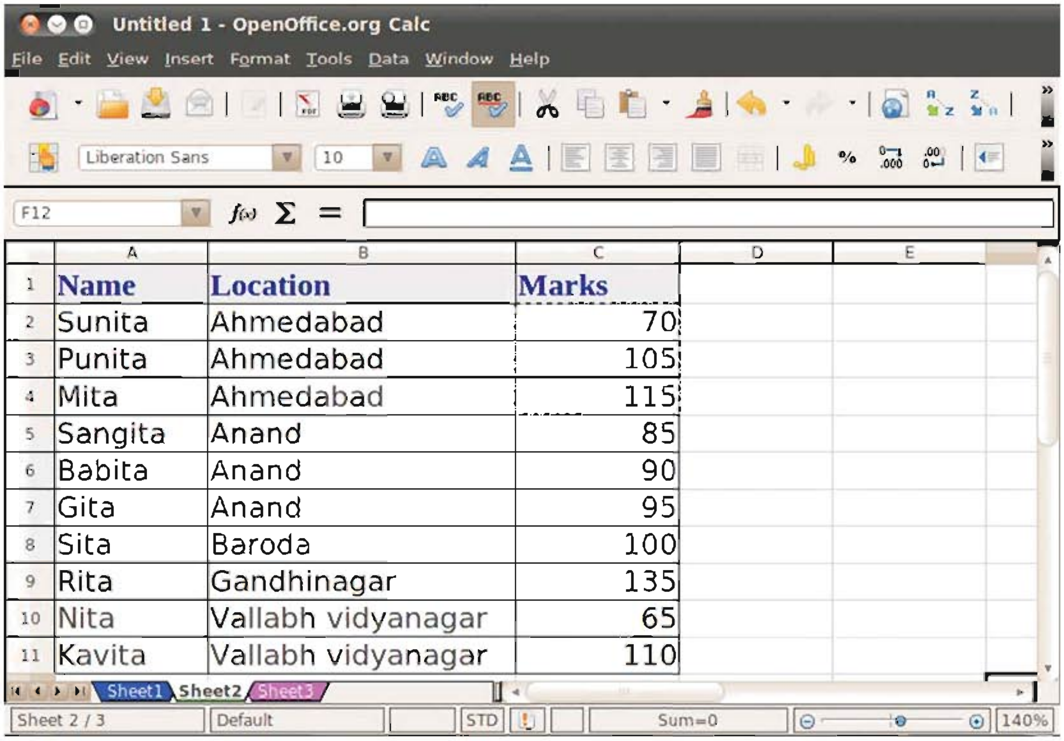
2. શીર્ષકો સાથે ડેટા પસંદ કરો.
3. પસંદ કરો : **Data → Sort**
4. **Sort by** ફિલ્ડમાં **"Location"** પસંદ કરો.
5. **Then by** ફિલ્ડમાં **"Marks"** પસંદ કરો.
6. હવે **OK** બટન દબાવો.

આ બધી ક્રિયાઓ આકૃતિ 6.21માં દર્શાવી છે.



આકૃતિ 6.21 : સૉર્ટના વિકલ્પો

આ બધાં પગલાંઓનો અમલ કરવાથી પસંદ કરેલો ડેટા પ્રથમ location (જગ્યા) અને પછી marks (ગુણ) ઉપર સૉર્ટ થશે (ગોઠવાશે). આનો અર્થ એ થાય કે Ahmedabad ના બધા વિદ્યાર્થીઓની યાદી તેણે મેળવેલા ગુણના ચઢતા ક્રમમાં (સૌથી ઓછા ગુણ ધરાવતો વિદ્યાર્થી પ્રથમ) મળશે. અહીં location ચઢતા ક્રમમાં સૉર્ટ કરતા હોવાથી Ahmedabad સૌપ્રથમ અને Vallabh Vidyanagar અંતમાં આવશે. સૉર્ટ કરેલો ડેટા આકૃતિ 6.22માં દર્શાવ્યા જેવો દેખાશે. જો જરૂર હોય તો તમે ત્રીજા લેવલનું સૉર્ટ પણ ઉમેરી શકો છો.

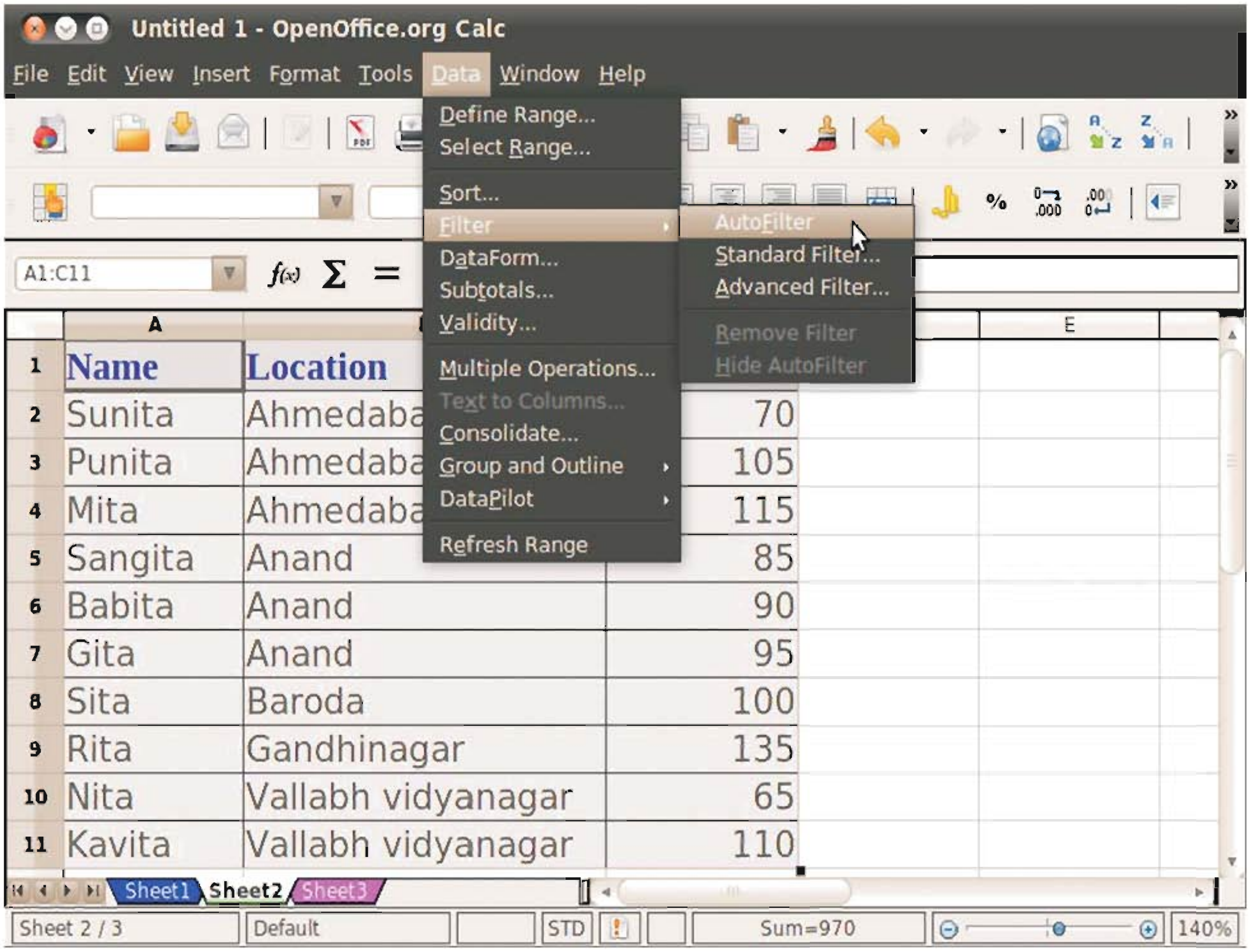


આકૃતિ 6.22 : ગોઠવણી પછીનો ડેટા (સૉર્ટેડ ડેટા)

ડેટા નિસ્યંદન (Data filtering)

ડેટા ફિલ્ટરિંગ આપણને ડેટા ગાળીને બિનજરૂરી ડેટા દૂર કરવામાં મદદ કરે છે અને તમે જે ઇચ્છો તે જ ડેટા રજૂ કરે છે. આપણે હમણાં જ અગાઉના વિભાગમાં સૉર્ટ કરેલો ડેટા કે જેમાં વિદ્યાર્થીઓનાં નામ અને જગ્યા વિશે હતો. તો લો. ચાલો આપણે ડેટા ઉપર એક નિસ્યંદક (ફિલ્ટર-filter) ઉમેરીએ. ધારો કે આપણે એ વિદ્યાર્થીઓની માહિતી જોવા ઇચ્છીએ છીએ કે જેમણે 100 અથવા 100 કરતાં વધારે ગુણ મેળવ્યા હોય. આ કાર્ય કરવા માટે ઓટો ફિલ્ટર આપણે નીચે પ્રમાણે ઉમેરી શકીએ :

1. આકૃતિ 6.23માં દર્શાવ્યા પ્રમાણે **Data → AutoFilter** પસંદ કરો. તમે હયાત વર્કશીટ કે જેમાં પહેલાં તમે ડેટા દાખલ કરેલો છે તે પણ વાપરી શકો છો.



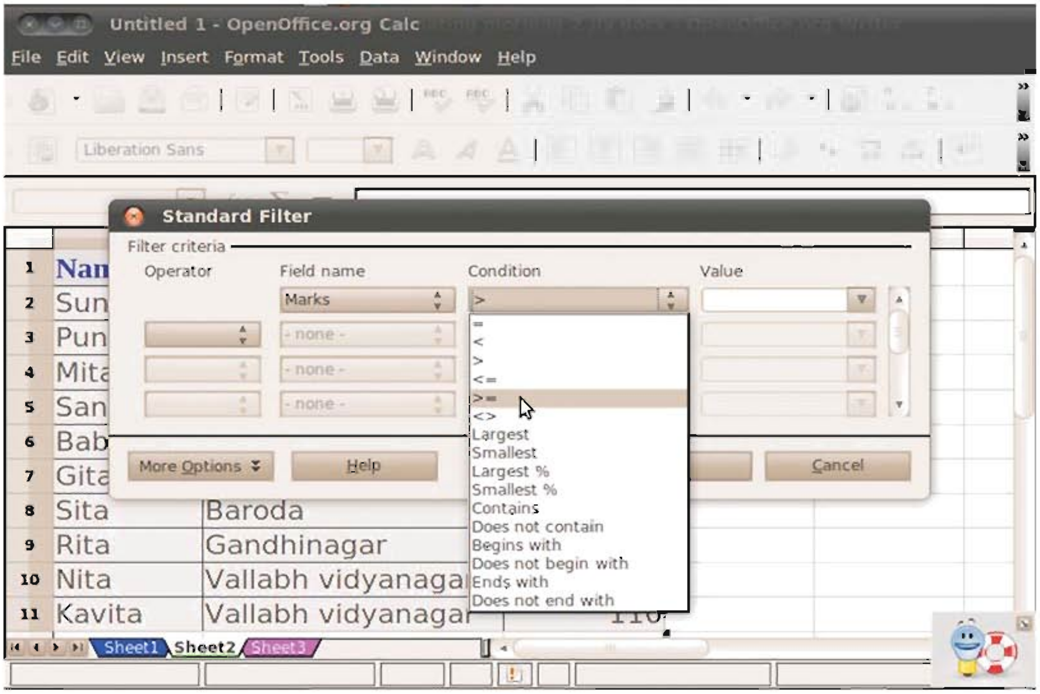
આકૃતિ 6.23 : પસંદ કરેલા ડેટા ઉપર ફિલ્ટર લાગુ કરવા

2. દરેક કૉલમના મથાળે તમે ડ્રોપડાઉન એરો (dropdown arrows) જોઈ શકશો.

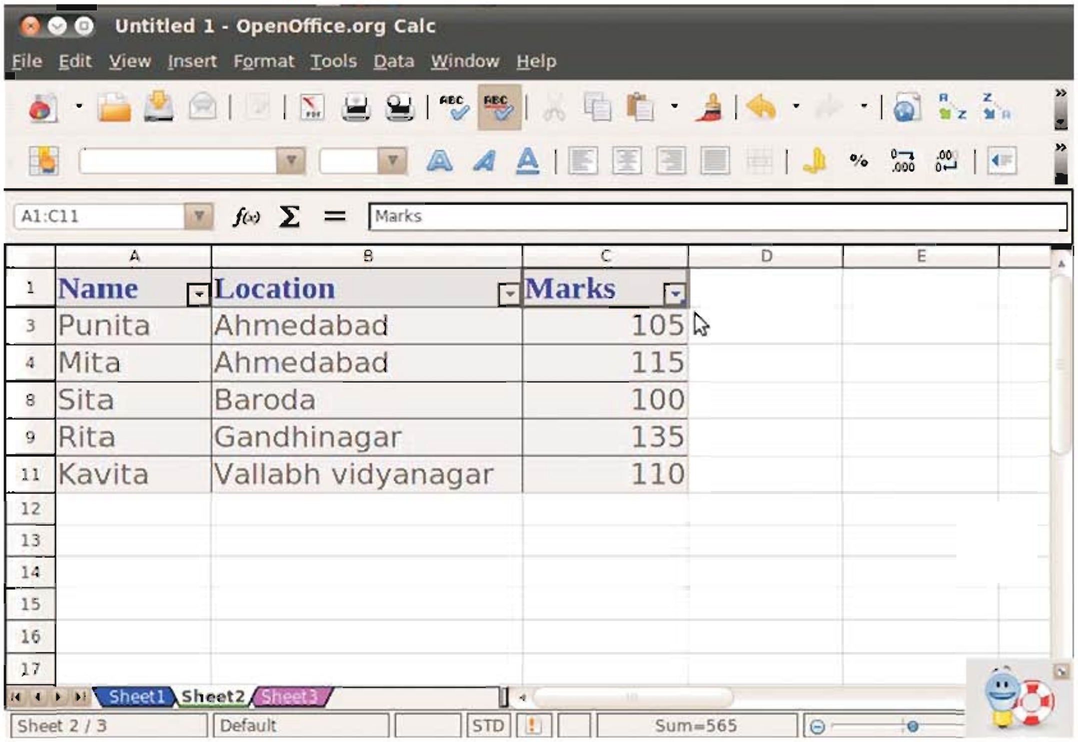
3. ક્લિક કરી કોઈ ચોક્કસ કૉલમ(અહીં Marks)નાં તીર(એરો)ને પકડી રાખો અને કિંમત પસંદ કરો. જેમકે, તમે અગાઉ દાખલ કરેલી કિંમતો અથવા વાક્યપદ (phrases) જેમ કે પહેલાં 10 શ્રેષ્ઠ વગેરે. જો તમારે કોઈ વિશિષ્ટ વિકલ્પ જોઈતો હોય તો તમે standard filter પસંદ કરી આકૃતિ 6.24માં દર્શાવ્યા પ્રમાણે તેમાંથી કિંમતો પસંદ કરો.

4. તમે જોઈ શકશો કે સ્પ્રેડશીટમાં જે વિદ્યાર્થીના ગુણ 100 કરતાં ઓછા હતા તે ગાળીને દૂર કરી દીધા છે. આકૃતિ 6.25માં દર્શાવ્યા પ્રમાણે તીરનો બદલાયેલો રંગ કે જે કેટલોક ડેટા ગાળીને દૂર કર્યો છે તે સૂચવે છે, તેનું કદાચ તમે નિરીક્ષણ કર્યું હશે.

ફરી પસંદ કરો : **Data → Filter → AutoFilter** જેથી ફિલ્ટરનું કાર્ય બંધ થાય.



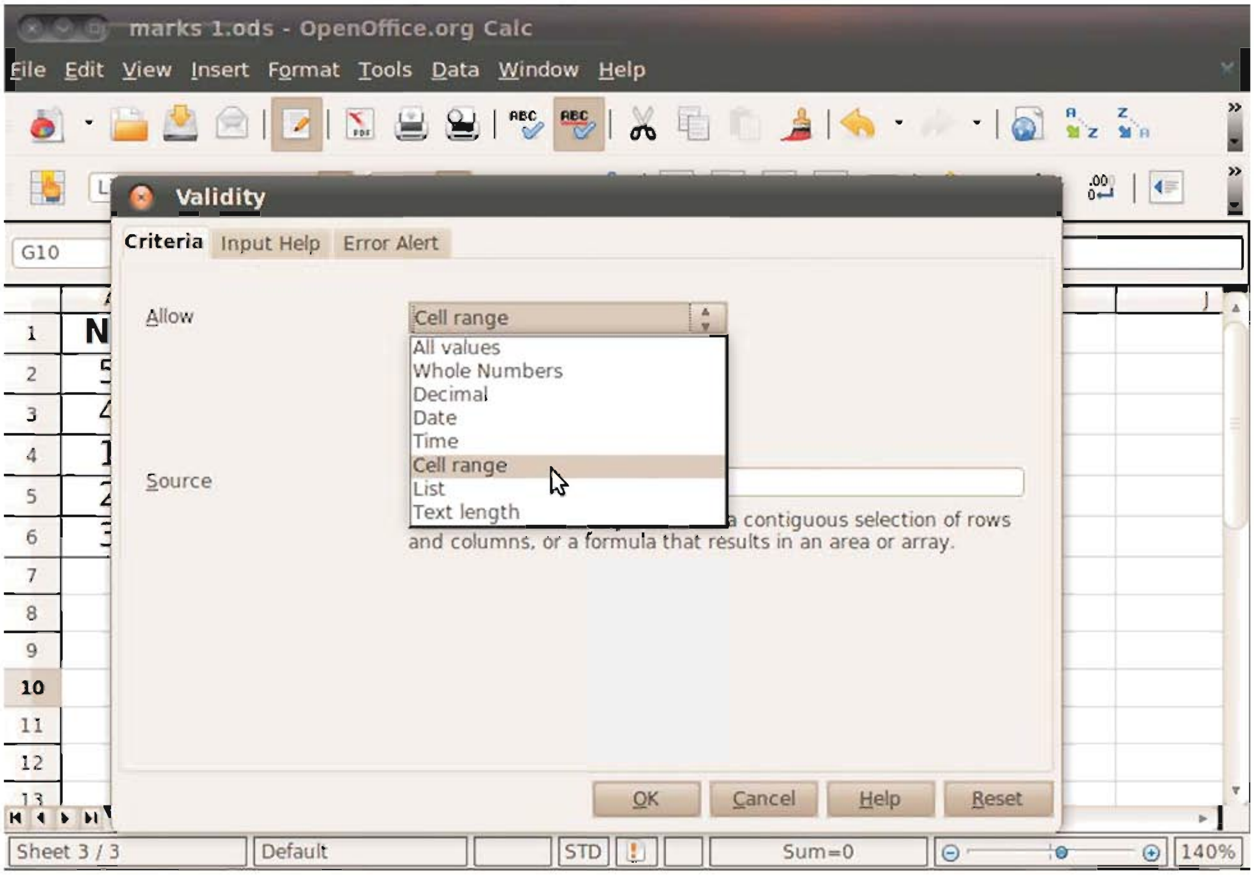
આકૃતિ 6.24 : Standard filter ડાયલૉગ બૉક્સ



આકૃતિ 6.25 : ગાળેલો (ફિલ્ટર્ડ) ડેટા

ડેટા ચકાસણી (Data Validations)

અગાઉ જણાવ્યું હતું તે પ્રમાણે સ્પ્રેડશીટ પૅકેજનો ઉપયોગ માહિતી વિશ્લેષણ જેવાં કે સરવૈયું તૈયાર કરવું અને નાણાંકીય વિશ્લેષણ, મેરિટ યાદી તૈયાર કરવી અને વિદ્યાર્થીઓના પરિણામના વિશ્લેષણમાં થાય છે. આ કિસ્સામાં જો તમે માન્ય (valid) માહિતી આપી હશે તો જ તમને આપેલાં સૂત્રો અને વિધેયો સાચું પરિણામ આપશે. જો કે, સિસ્ટમના ઉપયોગકર્તા હંમેશા માન્ય ડેટા દાખલ કરશે તેની કોઈ ખાતરી આપી શકતું નથી. ડેટા દાખલ કરતાં સમયે ભૂલો રોકવા (ટાળવા) માટે આપણે ડેટા **વેલિડેશન (data validations)**નો ઉપયોગ કરી શકીએ છીએ. સેલમાં રહેલી માહિતી સાચી છે (માન્ય છે) તેની ખાતરી કરવા માટે સેલ પસંદ કરો અને **Data → Validity** આદેશ આપો. આથી આકૃતિ 6.26માં દર્શાવ્યા પ્રમાણે Validity ડાયલૉગ બૉક્સ ખૂલશે.



આકૃતિ 6.26 : Validityના વિકલ્પો

જે ડેટા તમે ઉપયોગકર્તા પાસેથી મેળવો છો તેના માટે હવે તમે અલગ ધોરણ સેટ કરી શકો છો. અમાન્ય ડેટાને દાખલ થતો રોકવા માટે આપણે અહીં કેટલીક આવશ્યકતાઓ (શરતો) સેટ કરી શકીએ. આ આવશ્યકતાઓમાં સેલ વિસ્તાર, બધી કિંમતો, ફક્ત પૂર્ણાંક સંખ્યાઓ દાખલ કરવી, ફક્ત તારીખ જ માન્ય રાખવી વગેરેનો સમાવેશ થાય છે, જેના માટે Cell range, All values, ફક્ત Whole numbers અને Date તરીકે વિકલ્પો હોય છે. આ પ્રકારનાં સેટિંગ આપણે **Criteria** ટૅબ વડે આકૃતિ 6.26માં દર્શાવ્યા પ્રમાણે કરી શકીએ છીએ. બીજો ટૅબ **Input Help** છે જેના વડે ડેટા દાખલ કરતાં સમયે આપણે ઉપયોગકર્તાને સહાય પૂરી પાડી શકીએ છીએ.

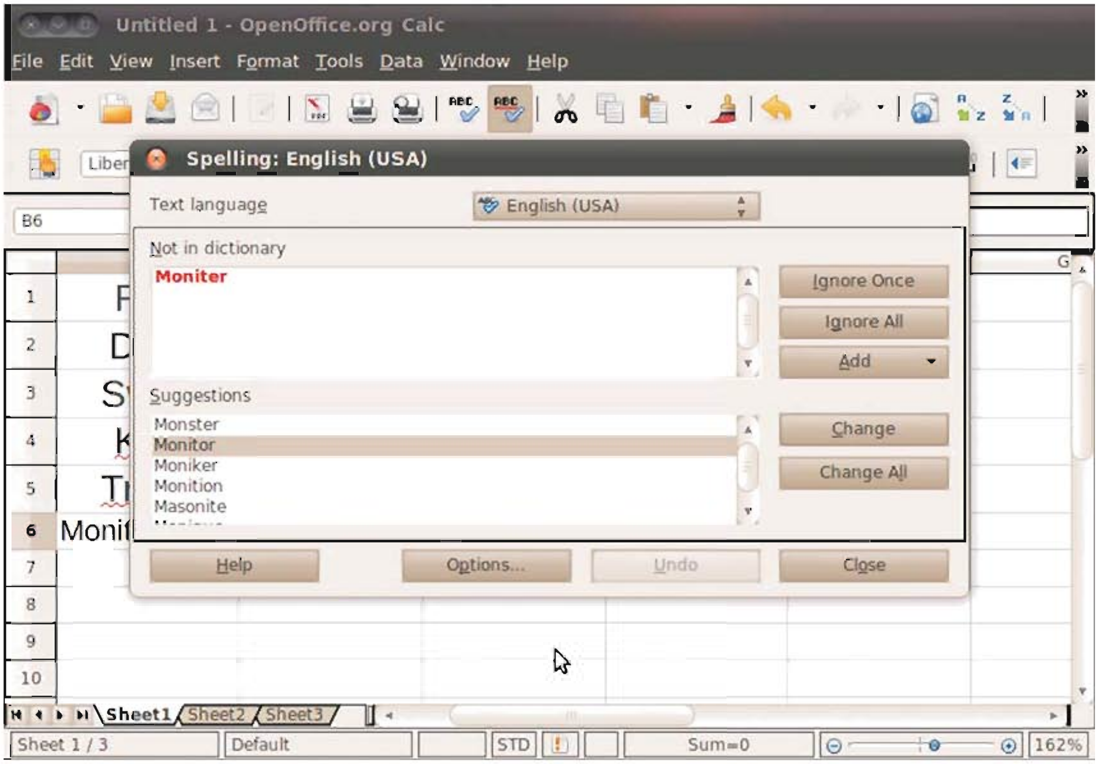
આ ઉપરાંત અન્ય એક **Error Alert** ટૅબ હોય છે કે જો કંઈ ભૂલ થાય તો ભૂલ માટે એક ચેતવણી (એલર્ટ) આપે છે. અહીં એ નોંધવું જોઈએ કે જો તમે **Delete Contents** વિન્ડોમાંથી **Format** અથવા **Delete All** પસંદ કરશો તો તમે સેટ કરેલા Validity વિકલ્પો દૂર કરવામાં આવશે.

કેલ્સીમાં અન્ય પ્રક્રિયાઓ (Other operations in Calc)

કેલ્સી અન્ય અનેક ઉપયોગી પ્રક્રિયાઓ પૂરી પાડે છે. જેમ કે, અંગ્રેજી શબ્દોની જોડણી(સ્પેલિંગ)ની ચકાસણી, શાબ્દિક લખાણ શોધવું અને બદલવું, વર્કશીટ છાપવી વગેરે. આમાંના કેટલાક વિકલ્પોની અહીં આ વિભાગમાં ચર્ચા કરી છે.

જોડણીની ચકાસણી (Spelling check)

કેલ્સીની વર્કશીટમાં માહિતી ટાઇપ કરતી વખતે ઘણીવાર ઉપયોગકર્તા અંગ્રેજીના સ્પેલિંગમાં ભૂલો કરે છે. કેલ્સી આપણને જોડણી (સ્પેલિંગ) ચકાસવાની સુવિધા આપે છે. જો તમે ટૂલબાર ઉપરનું Spelling check બટન (કે જે ટૉગલ બટન છે અને તેને ઑન કે ઑફ કરી શકાય છે) ઑન રાખેલું હશે તો ખોટી રીતે લખાયેલા શબ્દો લાલ રેખાથી રેખાંકિત થશે. જ્યારે કંઈક ખોટો શબ્દ (સ્પેલિંગમાં ભૂલ) ટાઇપ કરશો તો તે શબ્દની નીચે લાલ લીટી સાથે તે શબ્દ હાઇલાઇટ થશે. તમારે ફક્ત તેના ઉપર રાઇટ ક્લિક કરવાની જરૂર હોય છે અને આપેલી પસંદગીઓમાંથી યોગ્ય વિકલ્પ પસંદ કરો. જોડણી ચકાસવા માટે કેલ્સી શબ્દકોશનો ઉપયોગ કરે છે. જો તમે દાખલ કરેલો શબ્દ તે જ સ્પેલિંગ સાથે શબ્દકોશમાં મળી જાય તો તે શબ્દ સાચો ગણવામાં આવે છે. આકૃતિ 6.27માં જોડણીની ચકાસણી અને સુધાર માટેના શક્ય વિકલ્પ દર્શાવ્યા છે.



આકૃતિ 6.27 : જોડણી ચકાસણીના વિકલ્પો

અમુક સમયે એવું પણ બને કે શબ્દ સાચો હોય જેમ કે તમારું નામ, પણ તે શબ્દ પ્રમાણભૂત શબ્દકોશમાં ન હોય. આ કિસ્સામાં તમારા નામ નીચે પણ લાલ રેખા હોઈ શકે ! આવી પરિસ્થિતિ ટાળવા માટે તમારું નામ "Add to dictionery" વિકલ્પ વડે શબ્દકોશમાં ઉમેરી શકો. તે પછી તમારું નામ અમાન્ય શબ્દ તરીકે હાઇલાઇટ થશે નહીં. જોડણી ચકાસણી કરવા માટેનો એક સરળ માર્ગ છે માત્ર ફંક્શન કી F7ને દબાવવાનો. આપમેળે જોડણી ચકાસવાનું ટાળવા માટે તમે આ પ્રમાણે પસંદ કરી શકો : **Format → Cells → Font language=[None]**

શોધવું અને બદલવું (Find and Replace)

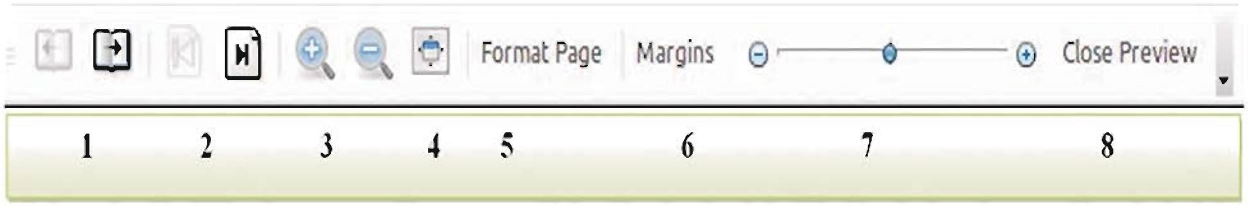
Find and Replace આદેશ વર્કશીટમાં ખૂબ ઝડપથી માહિતી (ડેટા) શોધવામાં અને સેલમાંનું શાબ્દિક લખાણ કે સંખ્યા સંપૂર્ણપણે અથવા આંશિક રીતે બદલવા માટે વપરાય છે. આ કાર્ય કરવા માટે નીચે પ્રમાણે પગલાં ભરો :

- પસંદ કરો : **Edit → Find & Replace**
- Find & Replace ડાયલૉગ બૉક્સમાં પૂછવામાં આવેલી જરૂરી ક્રિયાઓ કરો.

મુદ્રણ (Printing)

વર્કશીટના મોટા કદને કારણે વર્કશીટ છાપવા માટે (પ્રિન્ટિંગ માટે) ખાસ કાળજી જરૂરી છે. શબ્દ-પ્રક્રિયક (વર્ડ પ્રોસેસર) સૉફ્ટવેરની જેમ સ્પ્રેડશીટ વિનિયોગના દસ્તાવેજનું માપ નિશ્ચિત નથી. તમે અનેક આડી હરોળ (રો) અને ઊભી હરોળ (કૉલમ) વડે બનાવેલો દસ્તાવેજ એક પાનામાં બંધબેસતો ન પણ આવે. આ ઉપરાંત કેટલીક માહિતી કપાઈ જાય એમ પણ બને. વધુમાં, આખી વર્કશીટ છાપવી જરૂરી પણ નથી હોતી. આથી, કોઈ પણ દસ્તાવેજ છાપતાં પહેલાં, print preview (મુદ્રણ પૂર્વેનું અવલોકન) વિકલ્પ વાપરવો સલાહભર્યું છે. print preview વિકલ્પ પાનું પૂરું થાય તે નિશાની (પેઇજ બ્રેક) અને હાંસિયા (માર્જિન) સાથે સ્પ્રેડશીટનો દેખાવ પૂરો પાડે છે, વર્કશીટના પૂર્વાવલોકન (પ્રિવ્યૂ) માટે નીચે પ્રમાણે પગલાં ભરો :

- પસંદ કરો : **File → Page Preview**
- આથી કમ્પ્યૂટર સ્ક્રીનના ઉપરના ભાગમાં આકૃતિ 6.28માં દર્શાવ્યા પ્રમાણેનો એક બાર (bar) જોવા મળશે. તે દસ્તાવેજના પૂર્વાવલોકન માટે વપરાશકર્તા ઉપયોગમાં લઈ શકે તે ક્રિયાઓ દર્શાવે છે.



આકૃતિ 6.28 : પૂર્વાવલોકન (પ્રિવ્યૂ) બાર

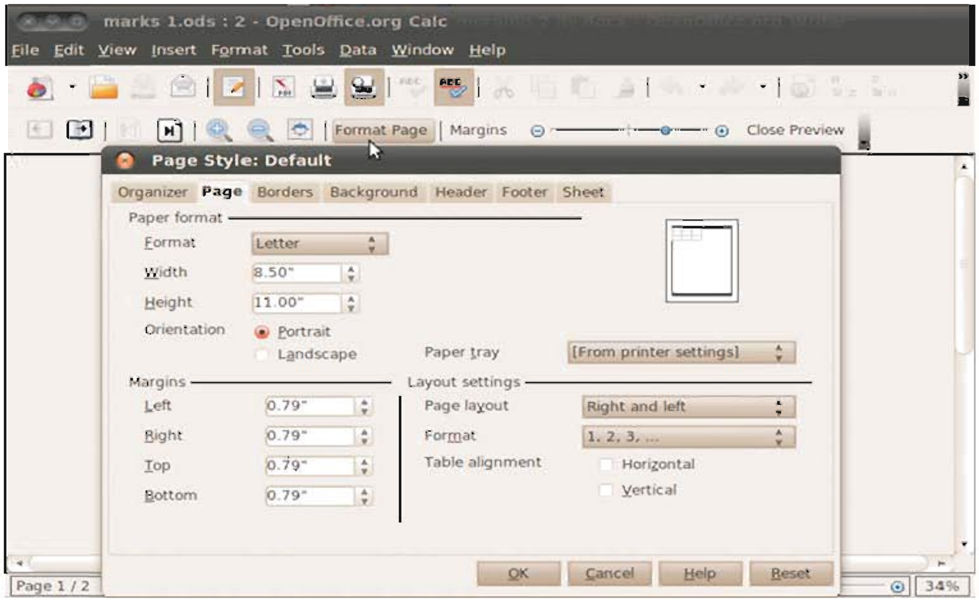
યોગ્ય આઇકોન પસંદ કરી આપણે જરૂરી પ્રક્રિયાઓ કરી શકીએ છીએ. આકૃતિ 6.28માં દર્શાવેલાં લેબલ જુદી જુદી ક્રિયાઓ માટે છે, તેની યાદી કોષ્ટક 6.7માં આપેલી છે.

લેબલ	વર્ણન
1	પાનાં બદલવાં
2	છેલ્લા પાના પર પહોંચવા
3	ઝૂમ ટૂલ
4	આખા સ્ક્રીન પરનો દેખાવ
5	પાનાના ફોર્મેટિંગ માટેનું ડાયલૉગ બૉક્સ ખોલવા માટે
6	હાંસિયા દર્શાવવા
7	સ્કેલિંગ
8	પ્રિવ્યુ બંધ કરવા

કોષ્ટક 6.7 : પૂર્વાવલોકન વિકલ્પો

પાનાનું પૂર્વાવલોકન પાનાના ફોર્મેટિંગની કિંમતો સેટ કરવા માટે પણ વપરાય છે. પાનાનું ફોર્મેટિંગ (page formatting) કરવા માટે નીચે પ્રમાણે પગલાં ભરો :

- પસંદ કરો : **File → Page Preview**
- સ્ક્રીન ઉપર પ્રદર્શિત આડા ડાયલૉગ બૉક્સમાંથી **Format Page** બટન દબાવો.
- આકૃતિ 6.29માં દર્શાવ્યા પ્રમાણે **Page Style** નામના ડાયલૉગ બૉક્સમાં **Page** લખેલા વિભાગને પસંદ કરો.
- જરૂરી પસંદગીઓ પસંદ કરી OK બટન દબાવી પરિવર્તનોનો સંગ્રહ કરો.

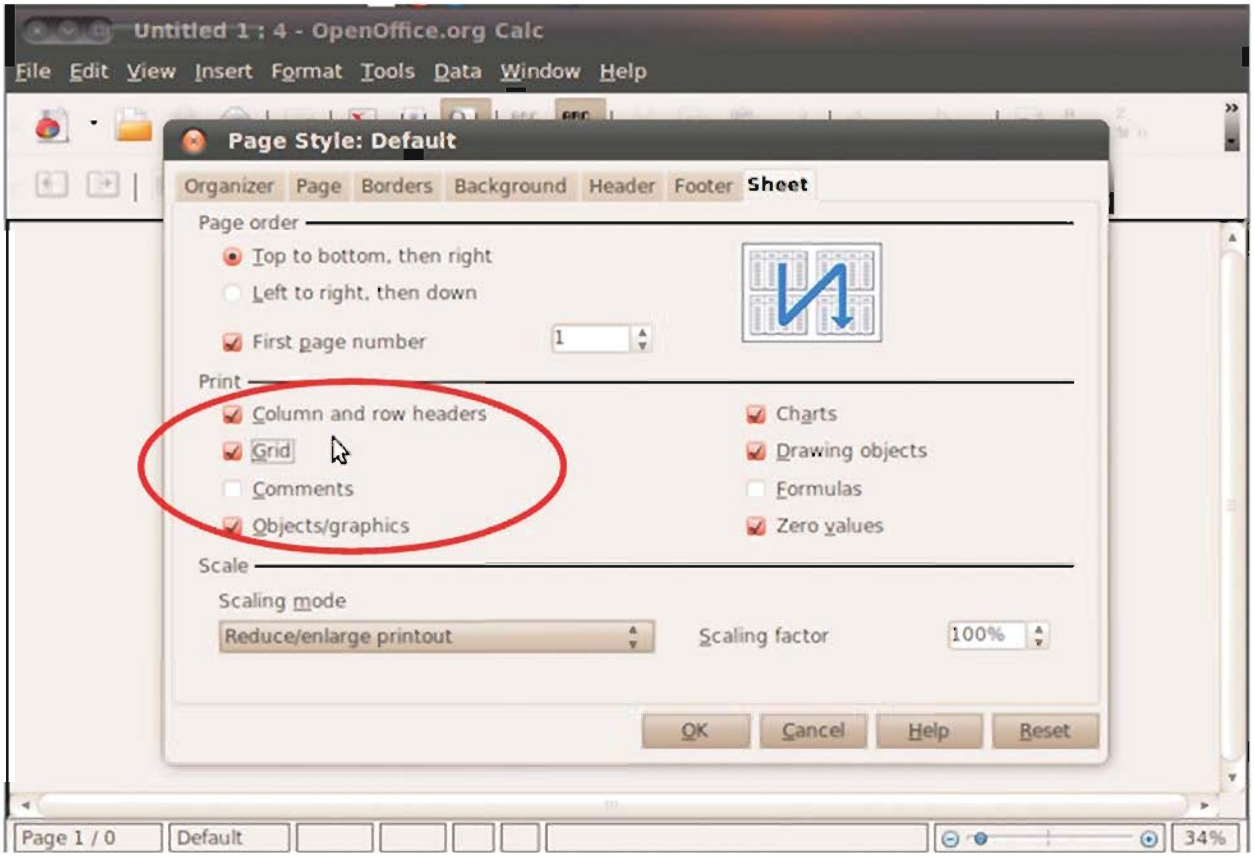


આકૃતિ 6.29 : Page formatingના વિકલ્પો

તમે પેપરનું માપ **(Format)** A4, લીગલ અથવા કોઈ પણ ઇચ્છિત પાનાના કદની પસંદગી કરી શકો છો. તમે **Margins** વડે ડાબી બાજુ, જમણી બાજુ, ઉપર અને નીચે માર્જિન (હાંસિયા) સેટ કરી શકો છો, આ વિકલ્પો ઉપરાંત તમે અન્ય Page formattingના વિકલ્પોના પ્રયોગો કરી શકો : **Paper Tray** વિકલ્પ કે જેમાં ઉપરની ટ્રેમાંથી કે નીચેની ટ્રેમાંથી તમારું પાનું પ્રિન્ટરમાં આવશે તે જણાવવાનું હોય છે અને **Page Layout** વિકલ્પનો ઉપયોગ એવા સમયે થાય છે જ્યારે એક ભૌતિક પાના ઉપર તમે એક કરતાં વધારે પાનાં છાપતાં હો. Formatની પસંદગી આડું પાનું કે ઊભું પાનું જેવું **Orientation** જણાવવા માટે પણ થાય છે.

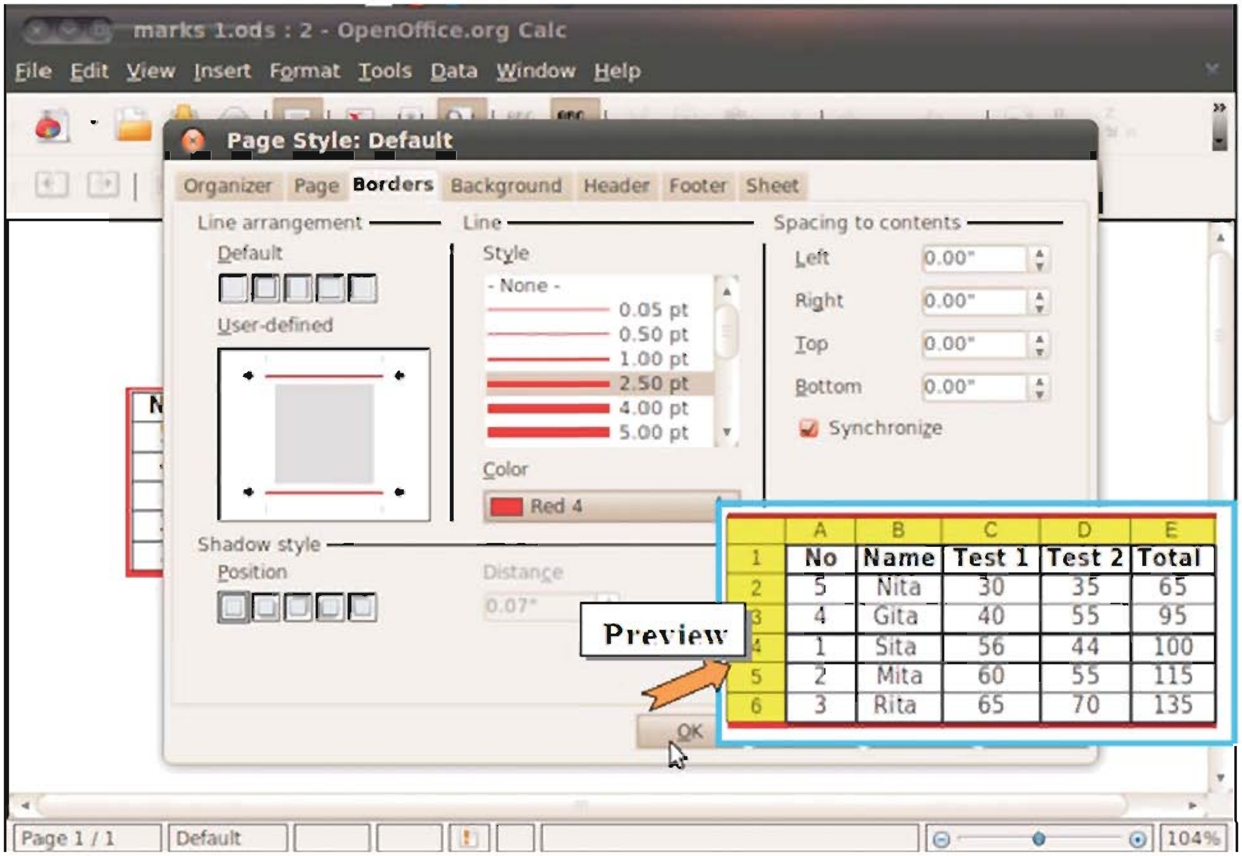
જો તમે આડી હરોળનાં મથાળાં, ઊભી હરોળનાં મથાળાં સાથે સેલ ગ્રીડ (સેલની આસપાસ કિનારીઓ સાથે) છાપવા ઈચ્છતા હોય તો નીચે જણાવ્યા પ્રમાણેની ક્રિયાઓ કરો :

- પસંદ કરો : **Format → Page** એક ડાયલોગ બૉક્સ સ્ક્રીન ઉપર પ્રદર્શિત થશે.
- આના વિકલ્પરૂપે, પસંદ કરો **File → Page Preview** અને **Format Page** બટન દબાવો.
- હવે **Sheet** ટેબ ખોલો (આકૃતિમાં છેલ્લો ટેબ)
- આકૃતિ 6.30માં દર્શાવ્યા પ્રમાણે **Grid** ચેકબૉક્સને ટિક કરો (ક્લિક કરવાથી ખરાની નિશાની ચેકબૉક્સમાં દેખાશે).
- હવે **Column and row headers** ચેકબૉક્સ ટિક કરો.
- ડાયલોગ બૉક્સ બંધ કરો.



આકૃતિ 6.30 : column and row headers તેમ જ cell gridsને સેટ કરવી

Borders ટેબનો ઉપયોગ કરીને તમે રંગીન કિનારીઓ પણ છાપી શકો છો. તમે કિનારીઓ (borders) સેટ કરવાના અન્ય વિકલ્પો સાથે કિનારીનો પ્રકાર, કિનારીનો રંગ અને લીટીની જાડાઈ પણ પસંદ કરી શકો છો. આકૃતિ 6.31માં સેલની કિનારી સેટ કરવાના વિકલ્પો દર્શાવ્યા છે. એક વાર તમે પ્રાચલ (પેરામીટર) સેટ કરી દો પછી OK બટન દબાવો. તે પછી તમે પાનાનું પૂર્વાવલોકન કરી શકો. આકૃતિ 6.31માં લાલ રંગની કિનારી તેમજ દાખલ કરેલા ડેટા પીળી પશ્ચાદ્‌ભૂમિ સાથે ડેટાનું પૂર્વાવલોકન (ભૂરા લંબચોરસને કાળા તીર વડે દર્શાવ્યો છે) દર્શાવેલું છે. જો કે પૂર્વાવલોકન જોવા માટે Borders, cell headers અને background સેટ કર્યા પછી તમારે Page Previewમાં જવું પડે છે.



	A	B	C	D	E
1	No	Name	Test 1	Test 2	Total
2	5	Nita	30	35	65
3	4	Gita	40	55	95
4	1	Sita	56	44	100
5	2	Mita	60	55	115
6	3	Rita	65	70	135

આકૃતિ 6.31 : Borders અને Background સેટ કરવાં

જ્યારે દરેક વિકલ્પ અને ફોર્મેટ સેટ થઈ જાય તે પછી તમે સૌથી ઉપરની લીટીમાં રહેલા પ્રિન્ટર આઇકોન ઉપર ફક્ત ક્લિક કરીને દસ્તાવેજ છાપી શકો છો.

પેઈજ બ્રેક બદલવું (Modifying page breaks)

પાનું પૂરું થાય તેની નિશાની (પેઇજ બ્રેક) જોવા અને તેને બદલવા માટે નીચેની ક્રિયાઓ કરો :

- પસંદ કરો : **View → Page Break Preview**
- જો તમે પાનાના દેખાવથી સંતુષ્ટ હો તો તમે તે પ્રિન્ટ કરી શકો છો નહીંતર તમારી જરૂરિયાત પ્રમાણે લીટીને ખેંચીને (ડ્રેગ કરીને) પેઇજ બ્રેક બદલો.

મેનૂનો આદેશ **View → Normal** આપવાથી તમે પાનાનો સામાન્ય દેખાવ જોઈ શકો છો.

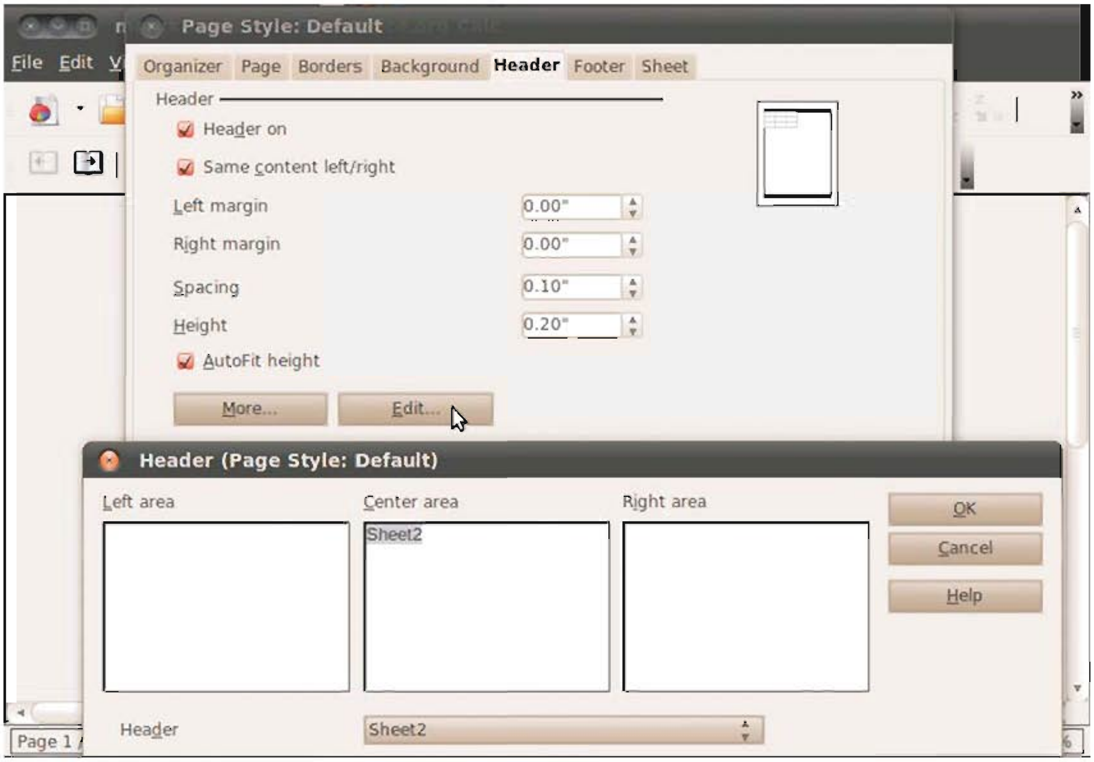
હેડર અને ફૂટર (Header and Footer)

દરેક પાનાની ઉપરની બાજુએ અને દરેક પાનાની નીચેની બાજુએ થોડો વિસ્તાર અનામત રાખવામાં આવે છે. આ ઉપરના અને નીચેના વિસ્તારને અનુક્રમે હેડર (header) અને ફૂટર (footer) કહેવામાં આવે છે.

હેડર બનાવવા માટે નીચે પ્રમાણે આદેશ આપો :

- પસંદ કરો : **File → Page Preview → Format**
- હવે Header ટેબ ખોલો અને તેમાં રહેલાં Edit બટનને દબાવો અને જરૂરી પરિવર્તનો કરો.

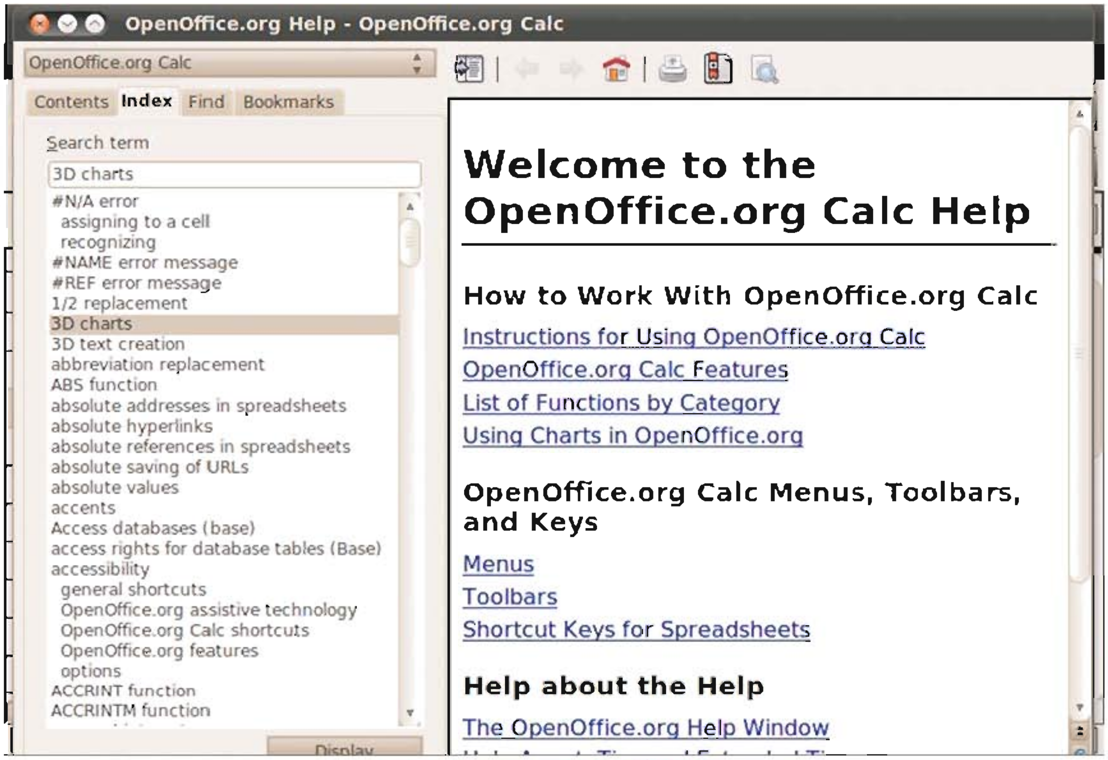
કેલ્સીનાં હેડર અને ફૂટર વિસ્તારને ત્રણ ભાગમાં વિભાજિત કરવામાં આવે છે : ડાબી બાજુનો વિસ્તાર, મધ્ય વિસ્તાર અને જમણી બાજુનો વિસ્તાર. તમે આ વિસ્તારમાં જે કંઈ લખો છો તે આપોઆપ દસ્તાવેજમાં ઉમેરાઈ જશે. આકૃતિ 6.32માં હેડર ઉમેરવાનું દર્શાવેલું છે.



આકૃતિ 6.32 : હેડર ઉમેરવું

મદદ મેળવવી – (Getting help)

કેલ્સીની અંદર જ વિવિધ કાર્યો અને શબ્દોને લગતી મદદ મેળવવા માટેની માર્ગદર્શિકા (manual) સમાયેલી હોય છે. કોઈ ક્રિયા બાબત માહિતી શોધવા માટે તેમાં (કેલ્સીમાં) સર્ચ (search) ક્રિયા પૂરી પાડવામાં આવે છે. આ માટે તમારે ફક્ત **Help → OpenOffice.org Help** આદેશ આપવો પડે છે. જ્યારે તમે મદદ મેળવવા માટેનો આદેશ આપો છો ત્યારે આકૃતિ 6.33માં દર્શાવ્યા પ્રમાણેનો સ્ક્રીન દેખાશે. આનો બીજો સરળ વિકલ્પ કી બોર્ડ ઉપરથી **F1** ફંક્શન કી દબાવવાનો છે.



આકૃતિ 6.33 : મદદ મેળવવી

સારાંશ

આ પ્રકરણમાં આપણે કેલ્સી સ્પ્રેડશીટમાં ડેટાને આકર્ષક રીતે રજૂ કરવા માટે ફૉર્મેટિંગ, સૉર્ટિંગ (ગોઠવણી) અને વેલિડેટ (ચકાસણી) ક્રિયાઓ બાબત શીખ્યા. આપણે સ્પ્રેડશીટ કક્ષાની ક્રિયાઓ વિશે જાણ્યું કે જે નવા કે હયાત દસ્તાવેજ ખોલવાનું, દસ્તાવેજને કેલ્સીમાં તેમજ અન્ય ફૉર્મેટમાં સંગ્રહ કરવાનું તેમજ ફાઈલ ઉપરની અન્ય ક્રિયાઓ જેવી કે દસ્તાવેજને શોધવાનું કે બંધ કરવાનું કાર્ય સરળ બનાવે છે. આ ઉપરાંત શીટ કક્ષાની ક્રિયાઓ જેવી કે શીટનું નામ બદલવું અને શીટનો રંગ બદલવો, શીટ ઉમેરવી કે દૂર કરવી વગેરે બાબત વિશે પણ શીખ્યા. આ પ્રકરણમાં આપણે આ ઉપરાંત સેલ કક્ષાની અને રો તેમજ કૉલમ કક્ષાની ક્રિયાઓ બાબત પણ શીખ્યા. આપણે ડેટાની ગોઠવણી, ડેટાની ચકાસણી અને અન્ય વિવિધ કાર્યો માટેની ક્રિયાઓ જેવી કે આપોઆપ સેલમાં માહિતી ભરવાની પ્રક્રિયા (ઑટોફિલ ઑપરેશન), હેડર અને ફૂટર સેટ કરવા, દસ્તાવેજ પ્રિન્ટ કરવું, જોડણીની ચકાસણી કરવી તેમજ મદદ મેળવવી વગેરે બાબત વિશે પણ શીખ્યા.

સ્વાધ્યાય

1. કેલ્સીમાં નવો દસ્તાવેજ કઈ રીતે બનાવી શકાય તે સમજાવો.
2. કેલ્સી વર્કશીટનું નામ કઈ રીતે બદલી શકાય અને તેનો રંગ કઈ રીતે બદલી શકાય તે સમજાવો.
3. કેલ્સીમાં વર્કશીટ કઈ રીતે ઉમેરી શકાય અને વધારાની વર્કશીટ દૂર કરી શકાય તે સમજાવો.
4. સેલમાં રહેલી માહિતી દૂર કરવા ઉપર ટૂંક નોંધ લખો.
5. કેલ્સીમાં ઑટોફિલ ટૂલ કઈ રીતે કાર્ય કરે છે તે સમજાવો.
6. કેલ્સીના સંદર્ભમાં સાપેક્ષ સ્થાનાંક અને નિરપેક્ષ સ્થાનાંક સમજાવો.
7. ડેટા નિસ્યંદન (ફિલ્ટરિંગ) ઉપર ટૂંક નોંધ લખો.
8. ડેટાની ક્રમબદ્ધ ગોઠવણી (સૉર્ટિંગ) ઉપર ટૂંક નોંધ લખો.
9. ડેટા ચકાસણી શું છે ? તે શા માટે જરૂરી છે અને કેલ્સી તે કઈ રીતે કરે છે તે સમજાવો.
10. કેલ્સીમાં પ્રિન્ટ કરતી વખતે હેડર અને ફૂટર કઈ રીતે ઉમેરી શકાય તે સમજાવો.
11. નીચે આપેલા વિકલ્પોમાંથી યોગ્ય વિકલ્પ પસંદ કરો :

(1) નીચેનામાંથી કઈ કાર્યરીતિ ડેટાને ચોક્કસ ક્રમમાં જેમ કે ચઢતા ક્રમમાં કે ઊતરતા ક્રમમાં ગોઠવવાનું કાર્ય કરે છે ?

(a) ડેટા ફૉર્મેટિંગ (b) ડેટા વેલિડેશન્સ
(c) ડેટા ફિલ્ટરિંગ (d) ડેટા સૉર્ટિંગ

(2) સેલમાં ફક્ત તારીખ જ દાખલ કરવા માટે નીચેનામાંથી કઈ કાર્યરીતિ વપરાય છે ?

(a) ડેટા ફૉર્મેટિંગ (b) ડેટા વેલિડેશન્સ
(c) ડેટા ફિલ્ટરિંગ (d) ડેટા સૉર્ટિંગ

(3) નીચેનામાંથી કયો વિકલ્પ જ્યારે પસંદ કરવામાં આવે છે ત્યારે બધા ડેટા વેલિડેશન્સ દૂર કરે છે ?

(a) Delete formatting (b) Delete all
(c) Delete formula (d) Delete me

(4) કેલ્સીની કઈ સગવડથી આપણે એક કરતાં વધારે જગ્યાએથી કોઈ શબ્દ બદલી શકીએ છીએ ?

(a) Find & Replace (b) ફક્ત Replace વડે

(c) Copy આદેશથી (d) Priview આદેશથી

(5) પેઇજ પ્રિવ્યૂ નીચેનામાંથી ક્યા સેટિંગ માટે પરવાનગી આપે છે ?

(a) Borders (b) Margins

(c) Column and row heading (d) આપેલા તમામ વિકલ્પો

(6) જોડણી ચકાસણી માટે કેલ્સીમાં કઈ ફંક્શન કીનો ઉપયોગ થાય છે ?

(a) F1 (b) F2 (c) F4 (d) F7

(7) કેલ્સીમાં મદદ મેળવવા માટે નીચેનામાંથી કઈ ફંકશન કીનો ઉપયોગ થાય છે ?

(a) F1 (b) F2 (c) F4 (d) F7

(8) ડેટાને ચોક્કસ ક્રમમાં ગોઠવવા માટેની પ્રક્રિયાને શું કહે છે ?

(a) સૉર્ટિંગ (b) સર્ચિંગ (c) ફિલ્ટરિંગ (d) વેલિડેટિંગ

(9) બિનજરૂરી ડેટાને ગાળીને દૂર કરવાની પ્રક્રિયાને શું કહે છે ?

(a) સૉર્ટિંગ (b) સર્ચિંગ (c) ફિલ્ટરિંગ (d) વેલિડેટિંગ

(10) કેવળ માન્ય ડેટાને જ દાખલ કરવાની પ્રક્રિયાને શું કહે છે ?

(a) સૉર્ટિંગ (b) સર્ચિંગ (c) ફિલ્ટરિંગ (d) વેલિડેટિંગ

પ્રાયોગિક સ્વાધ્યાય

1. કેલ્સી દસ્તાવેજને Save as વિકલ્પ વાપરીને અન્ય ફોર્મેટમાં સંગ્રહ કરો. તમે જે ફોર્મેટમાં સંગ્રહ કર્યો હોય તે સૉફ્ટવેરનો ઉપયોગ કરી તેને ખોલો.

2. કેલ્સી દસ્તાવેજ ખોલો અને Sheet ટૅબ વડે તેમાં 7 શીટ ઉમેરો, ત્યાર પછી તેને મેઘધનુષ્યના રંગો જેમ કે જાંબલી, વાદળી રંગ ... વગેરેથી દરેક શીટનો રંગ બદલો. તે પછી ખાતરી કરો કે દરેક શીટ તે પ્રમાણેના રંગની છે કે કેમ એટલે કે વાદળી રંગની શીટ વાદળી રંગથી બદલાઈ છે વગેરે.

3. ઋણ સંખ્યાઓ ઉપર ઑટોફિલ ટૂલનો ઉપયોગ કરો.

4. બે તારીખ સાથે ઑટોફિલ ટૂલ વાપરી જુઓ. એક સેલમાં 15-08-2013 લખો તેની બાજુના સેલમાં તે પછીની તારીખ લખો. આ માહિતીને તે પછીના દસ સેલમાં ડ્રેગ કરો. આ કાર્યરીતિનો ઉપયોગ કરી માસિક કૅલેન્ડર બનાવવા પ્રયત્ન કરો.

5. શબ્દકોશમાં તમારું નામ ઉમેરો જેથી તમારા નામ નીચે લાલ લીટી પ્રદર્શિત ન થાય.

6. તમારા મિત્રોની એક યાદી તેમની જન્મતારીખ સાથેની બનાવો. ત્યાર પછી તે યાદીને City અને પછી Namesના ક્રમમાં ગોઠવો. તેનાં શીર્ષકનાં મથાળાં નીચે પ્રમાણે લો :

Name	Address in one line	City	Mobile no	Email	Date of Birth	Month of Birth	Year of Birth
Khushi	1, Gokul Park	Anand	.....	.....	01	July	1995
....							

7. આ સ્વાધ્યાયના ઉદાહરણ 6માં તૈયાર કરેલી મિત્રોની યાદીને એ રીતે ફિલ્ટર કરો કે જેથી ફક્ત એ મિત્રોની માહિતી પ્રદર્શિત થાય જેની જન્મતારીખ ઑગસ્ટ માસમાં હોય.

8. Sorting અને Filtering વિશે Help વડે માહિતી મેળવો. આ વિષયો ઉપર Helpનો ઉપયોગ કરીને નોંધ તૈયાર કરો.

9. ડેટા ચકાસણી(વેલિડેશન્સ)ની નોંધનો ઉપયોગ અને આ સ્વાધ્યાયના ઉદાહરણ 6માં દાખલ કરેલા ડેટાનો ઉપયોગ કરીને date of birth કૉલમને ચકાસો જેથી કોઈ વ્યક્તિ તારીખમાં 33 જેવી સંખ્યા દાખલ કરી ન શકે. તમારે ડેટાને એ રીતે ચકાસવા પડે જેથી તે તારીખની કિંમતમાં ફક્ત 1 થી 31ની સંખ્યા જ સ્વીકારે.

કેલ્સીમાં વિધેય

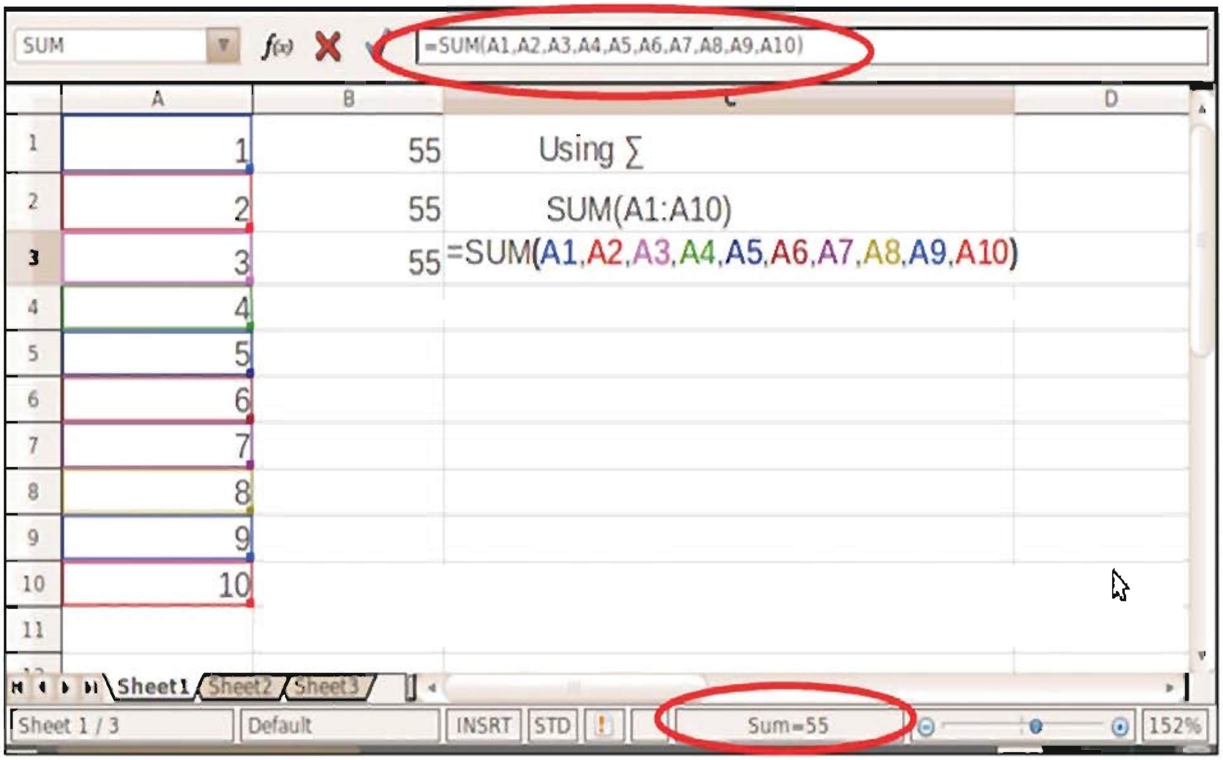
જટિલ નિર્ણયો લેવાની કામગીરી અને પુષ્કળ પ્રમાણમાં ડેટાનું વિશ્લેષણ ધરાવતાં વાસ્તવિક દુનિયાના વિનિયોગોમાં માત્ર ડેટા અને સરળ ગાણિતિક પદાવલિઓ દાખલ કરવી ઘણી વખત પૂરતી નથી. આ પ્રકારનાં કાર્યો કરવા માટે કેલ્સી જેવાં સ્પ્રેડશીટ પ્રોગ્રામ્સ ગાણિતિક, તાર્કિક, આંકડાકીય, સમય અને તારીખ, નાણાંકીય અને શાબ્દિક લખાણ તેમજ સંખ્યાની ગણતરીઓને લગતાં પ્રમાણિત વિધેયોને તેમની અંદર જ સમાવિષ્ટ હોય તે સ્વરૂપે પૂરાં પાડે છે.

કોઈ પદાવલિની જેમ વિધેયની શરૂઆતમાં બરાબરનું ચિહ્ન (=), તે પછી વિધેયનું નામ અને તેના કૌંસમાં એક અથવા વધારે વિધેયનાં ચલ (argument) જણાવવામાં આવે છે. ઘણીવાર વિધેય અને સૂત્રો સાથે "+" (વત્તાનું ચિહ્ન) જેવા પ્રક્રિયક (operator) પણ વાપરી શકાય છે. જો કે વિધેયને એક સાદા શાબ્દિક લખાણ તરીકે ટાળવા માટે પૂર્વગ તરીકે '=' (બરાબર)નું ચિહ્ન વાપરવાની એક સામાન્ય પ્રણાલી છે. વિધેયનાં ચલ (આર્ગ્યુમેન્ટ) તરીકે કોઈ કિંમત, સેલનો સ્થાનાંક, શાબ્દિક લખાણ, અચલ કિંમત અથવા એક કે વધારે વિધેયો હોઈ શકે. પ્રકરણ 5માં આપણે ટૂલબાર ઉપરથી (ખરીદીના બિલના ઉદાહરણમાં) sum વિધેયનો ઉપયોગ કર્યો હતો. તે જગ્યાએ આપણે ટૂલબાર ઉપરથી (સિગ્મા ચિહ્ન Σ વડે દર્શાવેલ છે જે એક પ્રકારનો ટૂંકો રસ્તો છે.) સીધું sum વિધેય લાગુ પાડ્યું હતું. એ જ વિધેય પદ્ધતિસર રીતે **=SUM(A1:A10)** લખી શકાય. જેમાં SUM એ વિધેયનું નામ, સરવાળાની પ્રક્રિયાનો ઉપયોગ કર્યો છે તેનો નિર્દેશ કરે છે, અને A1 તથા A10 કેલ્સીના એ જ વર્કશીટના સેલનો નિર્દેશ કરે છે. સેલ A1 થી A10 એક સાથે (A1:A10) એ એક સેલની વિસ્તાર (range) જણાવે છે જેમાં 10 અલગ અલગ સેલ છે. વિધેયના નામ પછી કૌંસમાં લખેલી માહિતી વિધેય પ્રદેશના ચલ (argument of the function) તરીકે ઓળખાય છે. વિધેયના ચલ તરીકે કોઈ સેલ સ્થાનાંક અથવા સેલ વિસ્તારને બદલે આપણે કોઈ અચલ કિંમત, જેમ કે, **=SUM(23,25)** પણ લખી શકીએ.

એ જ રીતે કદાચ તમે સેલ સ્થાનાંક A1 અને A10 વચ્ચે વિસર્ગ (:) ને બદલે અલ્પવિરામ (,) અથવા અર્ધવિરામ (;)નો ઉપયોગ કરીને વિધેય **=SUM(A1:A10)** પ્રયત્ન કરવા ઇચ્છો પણ ખરા. તમે નિરીક્ષણ કરી શકો કે ',' અથવા ';' નો ફક્ત ઉપયોગ કરવાથી સેલ A1 અને સેલ A10ની કિંમતોનો સરવાળો જ કરી શકાય. પણ જો સેલ A1થી A10 સુધીના બધા જ સેલની કિંમતોનો સરવાળો કરવો હોય તો આપણે ':' અથવા બધા જ સેલ સ્થાનાંક જુદા જુદા એટલે કે **=SUM(A1,A2,A3,A4,A5,A6,A7,A8,A9,A10)** વાપરવા જોઈએ.

આકૃતિ 7.1માં સરવાળાની ક્રિયા ત્રણ રીતે કરી શકાય તે દર્શાવ્યું છે. તમે પસંદ કરેલા સેલમાં કિંમત (SUM=55) અને ફૉર્મ્યુલા બારમાં સરવાળાની આખી પદાવલિ જોઈ શકો છો. તમે એ પણ નોંધ કરી હશે કે જ્યારે સૂત્રમાં A1નો નિર્દેશ કરવામાં આવે છે તો તેનો રંગ ભૂરો છે અને સેલ A1નો રંગ પણ ભૂરો છે. આ પદ્ધતિ દાખલ કરેલાં સૂત્રોનો તાળો મેળવવામાં (ક્રૉસચેક કરવામાં) ખરેખર ખૂબ ઉપયોગી થાય છે.

વિધેય(ફંક્શન)ને સામેલ કરવાની પ્રક્રિયા બધા વિધેયો માટે એકસરખી રહે છે. વિધેયના ચોક્કસ ચલ (આર્ગ્યુમેન્ટ) અને વિધેયના નામની સાચી જોડણી હોવી અનિવાર્ય છે. એટલે કે કોઈ પણ વિધેયનો ઉપયોગ કરવા માટે આપણે તે વિધેયના જરૂરી ચલ સાથે વિધેયનું ભૂલ વગરનું નામ (સાચો સ્પેલિંગ) દાખલ કરવું જરૂરી છે. આને વિધેયની વાક્યરચના (વિધેયના નિયમો syntax) કહેવામાં આવે છે. જો આપણે વિધેયના નામની સાચી જોડણી અને તેના શક્ય ચલ જાણતાં ન હોઈએ તો તે ભૂલમાં પરિણમશે.



આકૃતિ 7.1 : ત્રણ રીતે સરવાળો કરવાની ક્રિયા

સેલમાં વિધેય દાખલ કરવા માટે નીચે પ્રમાણે કાર્ય કરો :

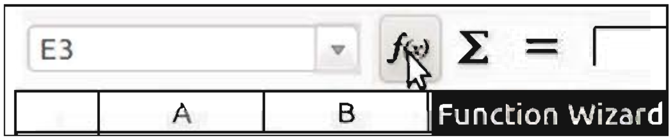
- જે સેલમાં તમારે પરિણામ પ્રદર્શિત કરવું હોય તે સેલ પસંદ કરો.
- હવે વિધેય દાખલ કરવાની શરૂઆત '=' ચિહ્ન સાથે કરો, જો તમે '=' ચિહ્ન ન લખો (અથવા અન્ય માન્ય પ્રચાલક જેમ કે "+"), તો તે વિધેય એક શાબ્દિક લખાણ તરીકે ગણાશે.
- વિધેયનું નામ દાખલ કરો.
- તે પછી વિધેયનાં ચલ દાખલ કરો.
- હવે એન્ટર કી દબાવો.

ફંક્શન વિઝાર્ડ (Function Wizard)

ટૂલબાર ઉપર અનેક વિધેયોનાં આઇકોન રાખવા (સામાન્ય રીતે સ્ક્રીનની સૌથી ઉપરની લીટી) અથવા આપણે તમામ વિધેયોને વાક્યરચના સાથે યાદ રાખવા શક્ય નથી, આથી ઘણાં ખરાં વિધેયો ફંક્શન વિઝાર્ડ (function wizard) વડે દાખલ કરવામાં આવે છે.

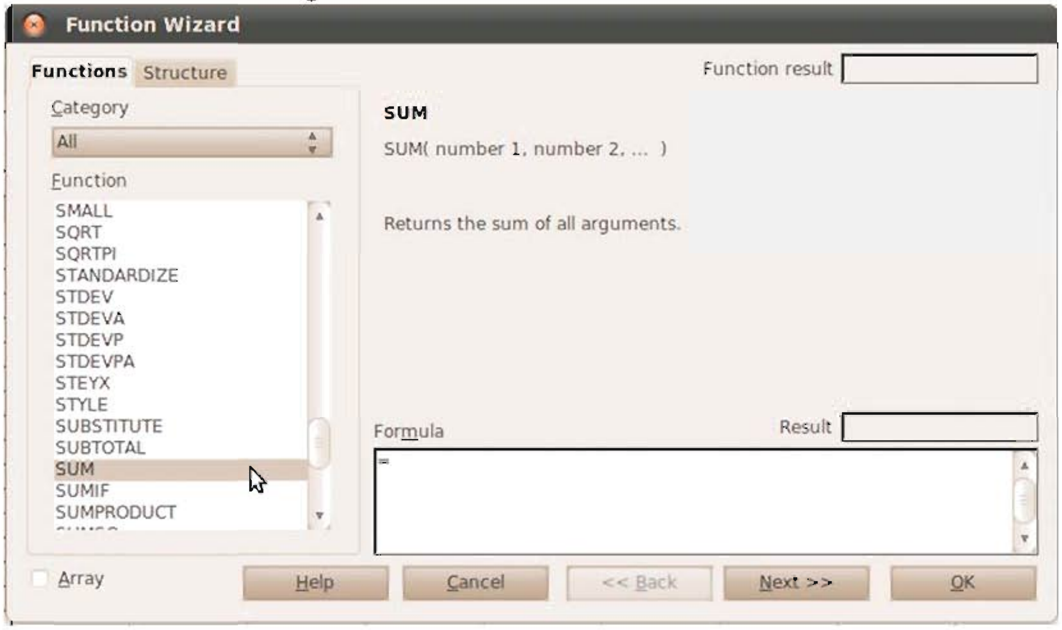
ફંક્શન વિઝાર્ડનો ઉપયોગ નીચે દર્શાવેલી રીતે થાય છે :

- જે સેલમાં પરિણામ પ્રદર્શિત કરવું હોય તે સેલ પસંદ કરો.
- પસંદ કરો : **Insert → Function**. આ કાર્ય તમે ખોલેલા સ્પ્રેડશીટ દસ્તાવેજમાં **Ctrl** અને **F2** દબાવીને પણ કરી શકો છો.
- આના વિકલ્પરૂપે તમે આકૃતિ 7.2માં દર્શાવ્યા પ્રમાણે ફૉર્મ્યુલા ટૂલબાર ઉપરનાં **Function Wizard** આઈકોન દબાવીને પણ તે કાર્ય કરી શકો છો.



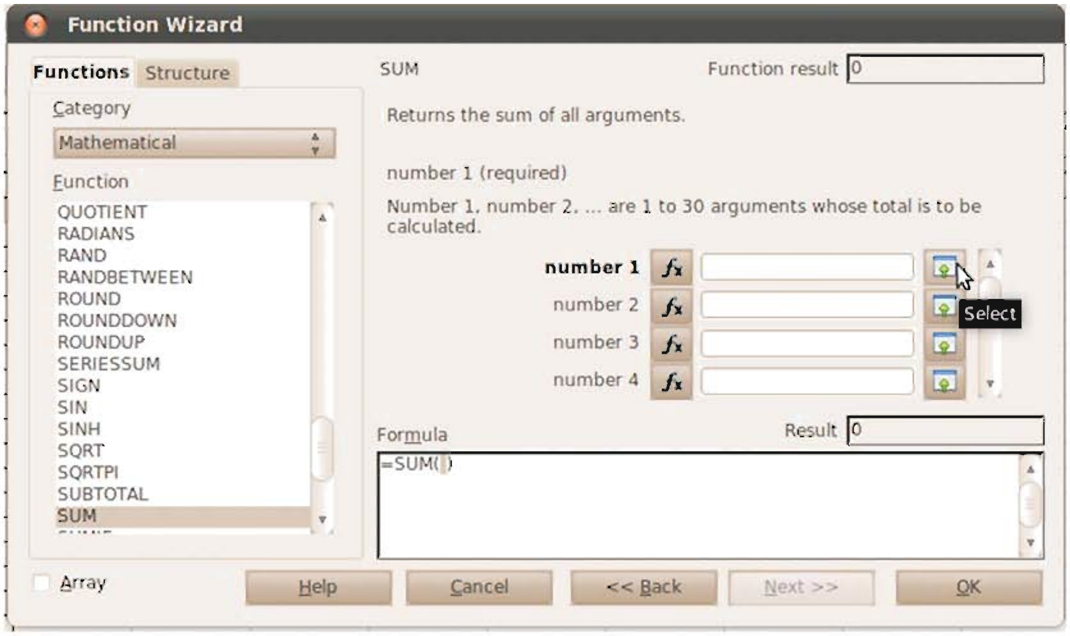
આકૃતિ 7.2 : ફંક્શન વિઝાર્ડ

- આથી એક ડાયલૉગ બૉક્સ સ્ક્રીન ઉપર દેખાશે. ડાયલૉગ બૉક્સમાંથી જરૂરી વિધેય પસંદ કરો. આકૃતિ 7.3માં એક લાક્ષણિક ડાયલૉગ બૉક્સ દર્શાવેલું છે.



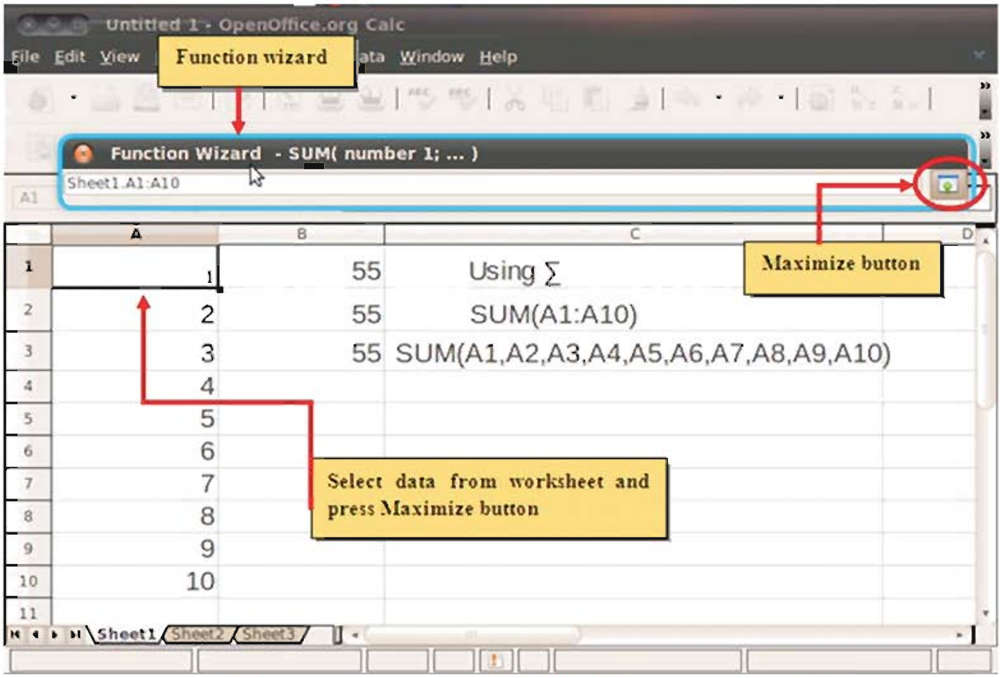
આકૃતિ 7.3 : ફંક્શન વિઝાર્ડનું ડાયલૉગ બૉક્સ

- **SUM** વિધેય પસંદ કરો. તે પછી **Next** બટન દબાવો.
- કિંમતો દાખલ કરો અથવા વર્કશીટમાંથી વિધેયનાં ચલ પસંદ કરો. અહીં તમે 30 ચલની કિંમતો, સ્થાનાંક (નિર્દેશો) અથવા વિસ્તાર એક પછી એક દાખલ કરી શકો છો. આકૃતિ 7.4માં દર્શાવ્યા પ્રમાણે તમે number 1, number 2, ... વગેરે દર્શાવીને ચલની યાદી જોઈ શકો છો.



આકૃતિ 7.4 : વિધેયમાં ચલ દાખલ કરવા

- જો તમે સંખ્યાની કિંમત કે સ્પ્રેડશીટમાં તે ક્યાં છે તે ન જાણતા હો તો તમે વર્કશીટમાંથી સીધા પસંદ કરવાનું ઇચ્છશો.
- Values પસંદ કરવા માટે આકૃતિ 7.4માં માઉસના તીરથી દર્શાવેલા **Select** બટનનો ઉપયોગ કરો.
- જો તમે Select બટન દબાવશો તો તે તમને વર્કશીટના વિસ્તારમાં લઈ જશે. ફંક્શન વિઝાર્ડનું આખું ડાયલોગ બૉક્સ એક નાના ટૂલબાર પ્રકારના લંબચોરસમાં પરિવર્તિત થઈ જશે. સ્ક્રીન ઉપર તમે વર્કશીટ વિસ્તાર તેમજ નાનું થયેલું ફંક્શન વિઝાર્ડ જોઈ શકશો. હવે તમે માઉસ વાપરીને સેલ કે સેલ વિસ્તાર પસંદ કરવા માટે મુક્ત છો. આકૃતિ 7.5માં નાનું થયેલું (minimised) ફંક્શન વિઝાર્ડ (ભૂરી રેખા સાથે લંબચોરસ દર્શાવેલ છે) અને વર્કશીટ વિસ્તાર દર્શાવેલો છે. જે ડેટામાં તમે રસ ધરાવતા હો તે વર્તમાન શીટમાં ઉપલબ્ધ ન હોય તો તમે અન્ય શીટ ઉપર પણ જઈ શકો છો. ફંક્શન વિઝાર્ડ ઉપર તે ડેટા ચકાસો. આપણા કિસ્સામાં તે **Sheet1:A1:10** છે.



આકૃતિ 7.5 : Select બટન વડે સેલ વિસ્તાર પસંદ કરવી

- આ કાર્ય કર્યા પછી **Enter** કી અથવા **Maximize** બટન (આકૃતિ 7.5માં દોરેલું વર્તુળ) દબાવો. આકૃતિ 7.5માં દર્શાવ્યા પ્રમાણે ફરી આખું ફંક્શન વિઝાર્ડ સ્ક્રીન ઉપર દેખાશે. તમે જોઈ શકશો કે તમે પસંદ કરેલી ડેટા વિસ્તાર number 1નાં બૉક્સમાં દશ્યમાન છે.
- કાર્ય પૂર્ણ કરવા માટે અને બંધ કરવા માટે OK બટન દબાવો.

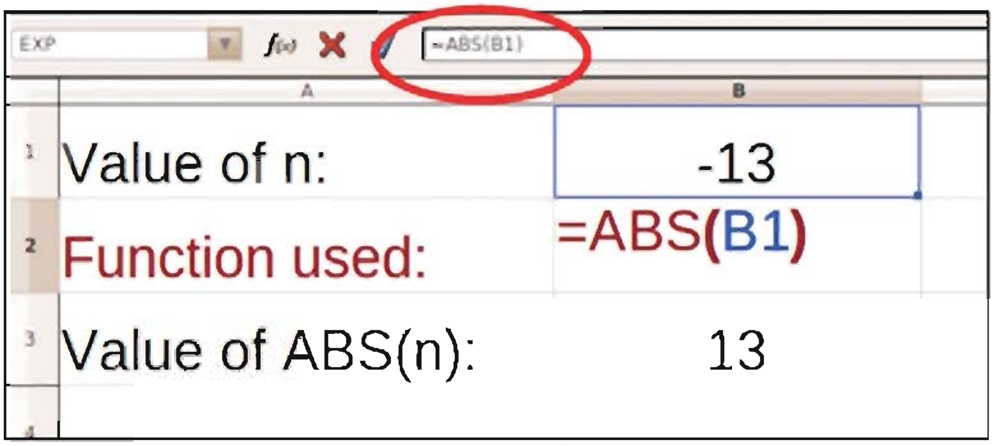
કેલ્સી વિવિધ શ્રેણીઓનાં તેની અંદર જ સમાવેલાં (built-in) અનેક વિધેયો પૂરાં પાડે છે. મુખ્ય શ્રેણીઓનાં વિધેયોમાં ગાણિતિક, આંકડાકીય, તાર્કિક અને શાબ્દિક લખાણનાં ફોર્મેટિંગને લગતાં વિધેયો છે. જ્યારે તમે એક વિધેયના ચલ તરીકે અન્ય કોઈ વિધેયનો ઉપયોગ કરવા ઇચ્છતા હો ત્યારે મૂળ વિધેયના કૌંસની અંદર ટેક્સ્ટ કર્સર (text cursor) મૂકીને વિધેયોની યાદીમાંથી નવું વિધેય પસંદ કરો.

ગાણિતિક વિધેયો (Mathematical functions)

જ્યારે આપણે કેલ્સીમાં સંખ્યાઓ સાથે કાર્ય કરીએ છીએ ત્યારે ગાણિતિક વિધેયો આપણને ખૂબ મદદરૂપ બને છે. ચાલો, હવે આપણે કેટલાંક ગાણિતિક વિધેયનો ઉપયોગ જોઈએ:

સંખ્યાની નિરપેક્ષ કિંમત (Absolute values of numbers)

ABS વિધેય ઋણ સંખ્યામાંથી ઋણ ચિહ્ન (–) દૂર કરે છે અને સંખ્યાને ધન બનાવે છે. તે 0 (શૂન્ય) કે ધન સંખ્યામાં કોઈ ફેરફાર કરતું નથી. તેની વાક્યરચના (સિન્ટેક્સ) **ABS (n)** છે, જ્યાં n એક સંખ્યા છે. આકૃતિ 7.6માં ABS વિધેયનું કાર્ય દર્શાવેલું છે. તમે ફૉર્મ્યૂલા બાર ઉપર વિધેયનું સૂત્ર (અંડાકાર ચિહ્નિત) પણ જોઈ શકો છો.

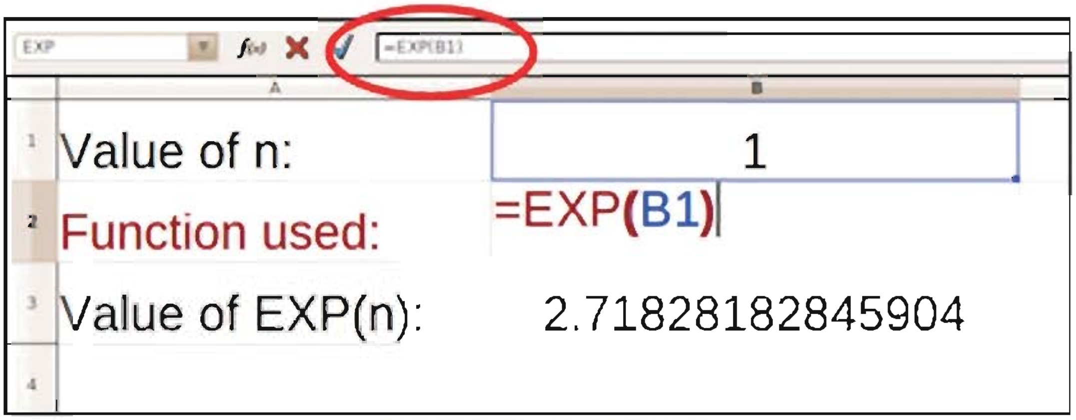


આકૃતિ 7.6 : ABS વિધેયનું કાર્ય

જ્યારે તમારે ફક્ત ધન સંખ્યાની જ જરૂર હોય તે સમયે ABS વિધેય ઘણું ઉપયોગી થાય છે. ધારો કે તમે એક પદાવલિની અંદર બીજી પદાવલિ વાપરી રહ્યા છો. આવા સમયે કેલ્સી સૌપ્રથમ અંદરની પદાવલિની કિંમત શોધશે અને પછી અંદરની પદાવલિની કિંમતનો ઉપયોગ બહારની પદાવલિમાં એક ચલ (આર્ગ્યુમેન્ટ) તરીકે કરશે. પણ ધારો કે બહારની પદાવલિ (જેમ કે વર્ગમૂળ) ફક્ત ઋણ ન હોય તેવી કિંમત જ (ધન અથવા 0) માન્ય રાખે છે. આવા સમયે જો અંદરની પદાવલિનું પરિણામ ઋણ આવે તો શું થાય ? અંદરની પદાવલિનું ઋણ પરિણામ બહારની પદાવલિને અમાન્ય બનાવી દે છે. આવી સ્થિતિ ટાળવા માટે આપણે અંદરની પદાવલિના પરિણામની નિરપેક્ષ કિંમતને બહારની પદાવલિમાં મોકલીએ છીએ.

ઘાતાંકીય વિધેય (Exponential function)

EXP વિધેય ઘાતાંકીય વિધેય e^xની કિંમત પાછી આપે છે. જ્યાં x એ આપેલી સંખ્યા છે અને e ની કિંમત આશરે 2.718281828 બરાબર છે. તેની વાક્યરચના **EXP(Number)** છે. દા.ત. જો આપણે કોઈ સેલમાં **= EXP(1)** લખીએ તો તે e^1 ગણાશે અને તેની કિંમત 2.72 પાછી આપશે. આકૃતિ 7.7માં આ વિધેય સમજાવ્યું છે. તમે ફોર્મ્યુલા બાર (અંડાકાર ચિહ્નિત) ઉપર વિધેયનું સૂત્ર જોઈ શકશો.

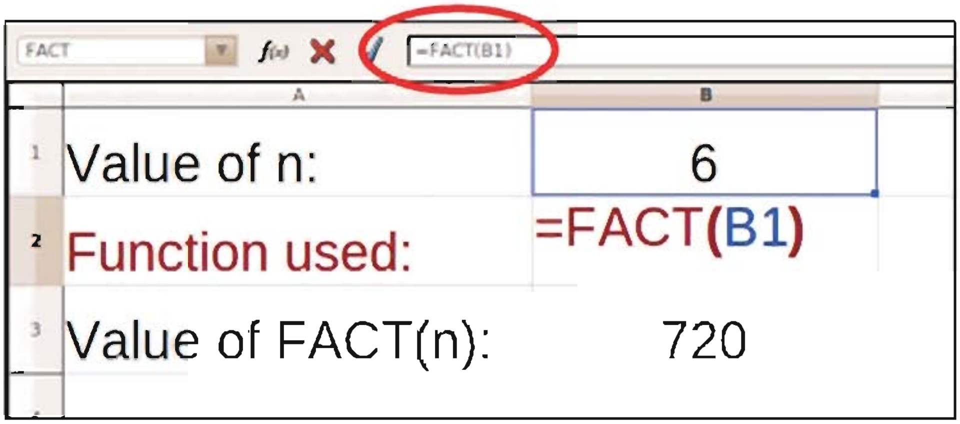


આકૃતિ 7.7 : EXP ફંક્શન

ક્રમગુણિત વિધેય (ફેક્ટોરિઅલ ફંક્શન - Factorial function)

FACT વિધેય આપેલી સંખ્યાની ક્રમગુણિત વિધેયની કિંમત પાછી આપે છે (1થી આપેલી પૂર્ણાંક સંખ્યા n સુધીનો ગુણાકાર). આ વિધેયની વાક્યરચના **FACT(n)** છે જ્યાં n એ પૂર્ણાંક સંખ્યા છે.

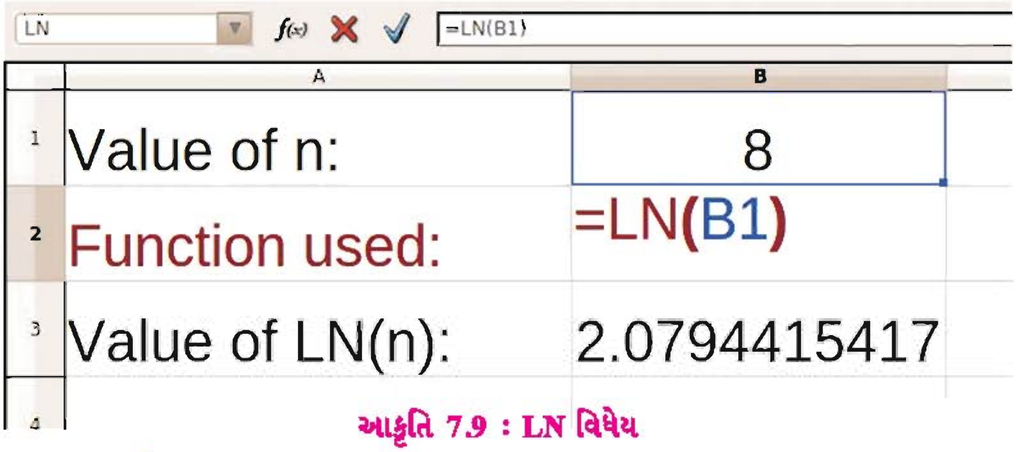
ઉદાહરણ તરીકે, જો આપણે **=FACT(3),** લખીએ તો કેલ્સી તેને પદાવલિ (1 * 2 * 3) ગણશે અને પરત કિંમત 6 કરશે. એ જ રીતે FACT(6) આપણને 1થી 6 સંખ્યાનો ગુણાકાર (એટલે કે 1*2*3*4*5*6) પરત આપશે, કે જે 720 છે. આકૃતિ 7.8માં સંખ્યા 6 સાથે ક્રમગુણિત વિધેય દર્શાવેલું છે.



આકૃતિ 7.8 : FACT વિધેય

પ્રાકૃતિક લઘુગણક (નેચરલ લોગેરિધમ - Natural logarithm)

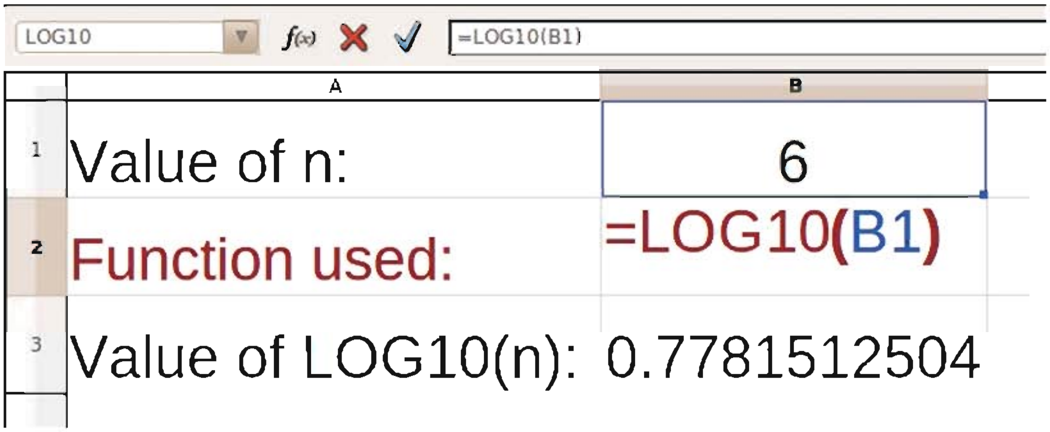
LN વિધેય આપેલી ધન સંખ્યાની પ્રાકૃતિક લઘુગણક (લઘુગણકનો આધાર e) કિંમત પરત આપે છે. આ વિધેયની વાક્યરચના **LN (Number)** છે, જ્યાં number એ ધન સંખ્યા છે. દા.ત., આપણે કોઈ સેલમાં =LN(8) લખીએ તો, આકૃતિ 7.9માં દર્શાવ્યા પ્રમાણે તે 2.08 ($\log_e 8$) જવાબ તરીકે આપશે.



આકૃતિ 7.9 : LN વિધેય

લઘુગણક આધાર 10 સાથે (Logarithm base 10)

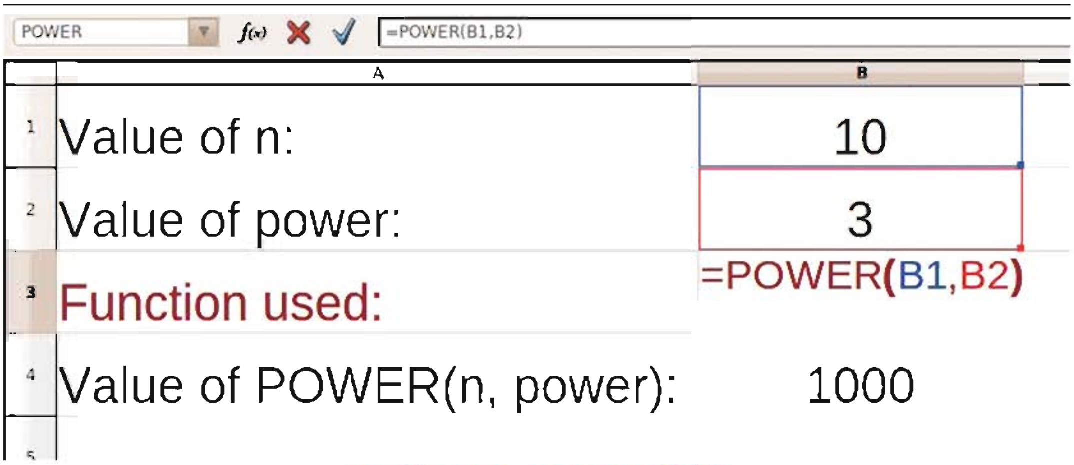
LOG10 વિધેય આપેલી ધન સંખ્યાની લઘુગણક આધાર 10 સાથે કિંમત પરત આપે છે. તેની વાક્યરચના **LOG10(Number)** છે. જ્યાં number એ ધન સંખ્યા છે. ઉદાહરણ તરીકે, LOG10(6)ને $\text{LOG}_{10}6$ ગણવામાં આવે છે અને આકૃતિ 7.10માં દર્શાવ્યા પ્રમાણે 0.77 કિંમત પરત આપે છે.



આકૃતિ 7.10 : LOG_{10} વિધેય

ઘાત (Power)

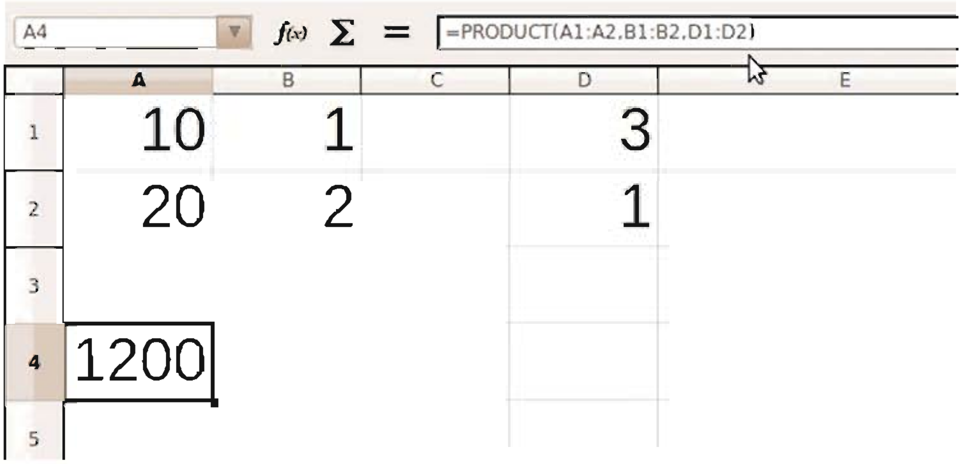
ઘાત (પાવર) વિધેય સંખ્યાની ઘાતની ગણતરી કર્યા પછીની કિંમત પરત આપે છે. તેની વાક્યરચના **POWER(Number, Power)** છે. જો કોઈ સેલમાં આપણે =POWER (10,3) લખીએ તો તે 1000 (10^3) કિંમત પરત આપશે. તે 10^3 ના સંકેત બરાબર છે. જુઓ આકૃતિ 7.11.



આકૃતિ 7.11 : POWER વિધેય

વિધેયના અનેક ચલનો ગુણાકાર (Product of many arguments)

PRODUCT વિધેય વધારેમાં વધારે 30 વિધેય પ્રદેશનાં ચલ(x1, x2, ..., x30)નો ગુણાકાર પાછો આપે છે. વિધેયનો દરેક ચલ કોઈ એક સેલ કે સેલ વિસ્તાર હોઈ શકે. આ વિધેયની વાક્યરચના **PRODUCT (n1, n2, n3, ..., n30)** છે. ધારો કે આપણે આકૃતિ 7.12માં દર્શાવ્યા પ્રમાણે વર્કશીટમાં ડેટા દાખલ કરેલો છે.

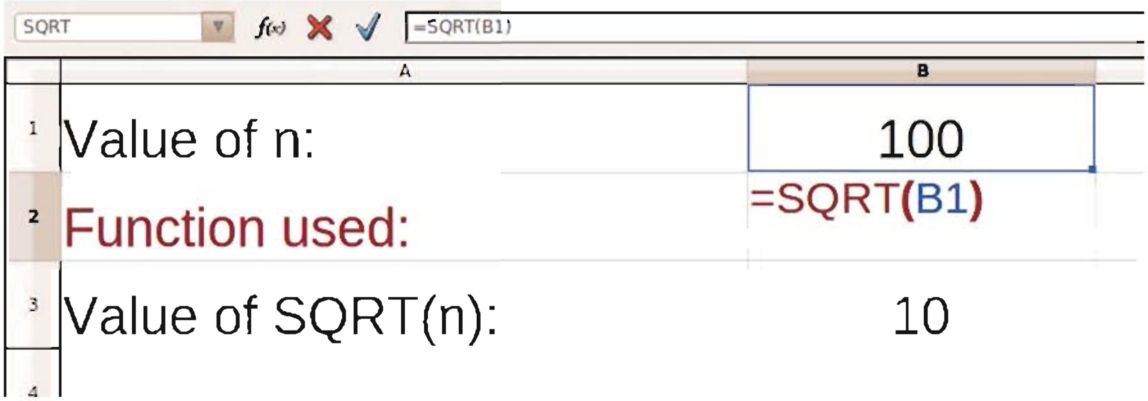


આકૃતિ 7.12 : PRODUCT વિધેયનો ઉપયોગ કરીને વિધેયનાં અનેક ચલનો ગુણાકાર

ધારો કે કોઈ સેલમાં આપણે =PRODUCT(A1:A2, B1:B2, D1:D2) લખીએ છીએ. આ વિધેય પરિણામ 1200 આપશે. આથી, આપણે કહી શકીએ કે આ product વિધેય સેલમાં રહેલી બધી કિંમતો, સેલનો વિસ્તાર અથવા એક કરતાં વધારે સેલના વિસ્તારને ધ્યાનમાં લે છે અને બધાનો ગુણાકાર કરીને પરિણામ આપે છે. આકૃતિ 7.12માં ઊભી હરોળ (કૉલમ) A, કૉલમ B અને કૉલમ Dમાં કુલ 6 કિંમતો લખેલી છે. નોંધ કરો કે કૉલમ C ખાલી છે.

વર્ગમૂળ (Square root) (SQRT વિધેય)

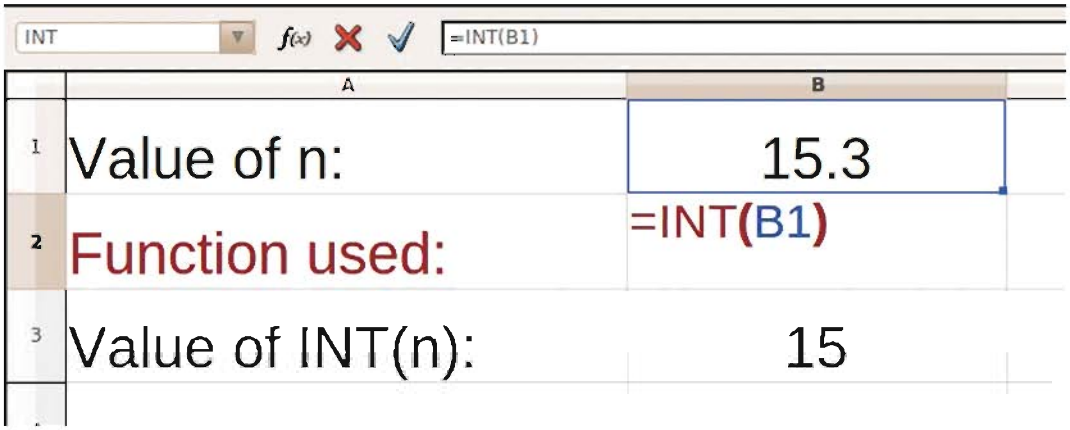
SQRT વિધેય ધન સંખ્યાનું વર્ગમૂળ પાછું આપે છે. આ વિધેયની વાક્યરચના **SQRT(Number)** છે, જ્યાં Number એ ધન સંખ્યા છે. આપણે 100નું વર્ગમૂળ શોધવા માટે કોઈ એક સેલમાં **=SQRT(100)** ફક્ત લખવું પડે. આકૃતિ 7.3માં દર્શાવ્યા પ્રમાણે તે કિંમત 10 પરત આપશે.



આકૃતિ 7.13 : વર્ગમૂળ (SQRT) વિધેય

પૂર્ણાંક સંખ્યા (Integer) (INT વિધેય)

INT વિધેય કોઈ આપેલી સંખ્યાને તેનાથી સૌથી નજીકની નાની પૂર્ણાંક સંખ્યામાં ફેરવે છે. તેની વાક્યરચના **=INT(Number)** છે. એટલે કે, જો તમે કોઈ સેલમાં INT (15.3) લખો તો આકૃતિ 7.14માં દર્શાવ્યા પ્રમાણે તે વિધેય 15 સંખ્યા પરત આપશે.



આકૃતિ 7.14 : INT વિધેય વાપરીને પૂર્ણાંક સંખ્યા મેળવવી

રાઉન્ડ (Round આસન્ન મૂલ્ય)

Round વિધેય તમે આપેલી સંખ્યાને તમે જણાવેલાં દશાંશચિહ્ન સુધી રાઉન્ડ (ટૂંકાવવાનું) કાર્ય કરે છે. ઘણા સમયે સૂત્રના પરિણામની સંખ્યામાં દશાંશચિહ્ન પછી ઘણા આંકડાઓ હોય છે. ઉદાહરણ તરીકે આવી સંખ્યા છે 12.6546. જો તમે દશાંશચિહ્ન પછી ફક્ત બે જ અંક ઇચ્છતા હો તો દેખીતી રીતે તમે તે સંખ્યાને કાપી ન શકો. તમારે તે સંખ્યાનું રાઉન્ડિંગ કરવું પડે. આ સંખ્યા 12.6546નું રાઉન્ડિંગ થઈને 12.65 સંખ્યા મળે. વિદ્યાર્થીઓનાં પરિણામ (કુલ ગુણ અથવા ટકા) અનેક સમયે એ રીતે રાઉન્ડ કરવામાં આવે છે કે જેથી તે એક પૂર્ણ સંખ્યા બની રહે. એટલે કે, જો કોઈ વિદ્યાર્થીએ 76.66 ટકા ગુણ મેળવ્યા હોય તો તે વિદ્યાર્થીએ 77 ટકા મેળવ્યા છે એવું ગણવામાં આવે છે. ROUND વિધેયની વાક્યરચના **ROUND (Number, Places)** છે. ROUND વિધેયનાં કેટલાંક ઉદાહરણો નીચે આપેલાં છે :

=ROUND (76.6633,2) નું પરિણામ 76.66 મળે.

=ROUND (45.1634,0) નું પરિણામ 45 મળે.

=ROUND (–56.5467,3) નું પરિણામ –56 મળે.

ઉપરનાં બીજા ઉદાહરણમાં આપણે પૂર્ણ સંખ્યા મેળવવા માટે 0 દશાંશચિહ્ન સુધી સંખ્યાને રાઉન્ડ કરેલી છે. આકૃતિ 7.15માં ROUND વિધેય દર્શાવ્યું છે.

	A	B	C	D
1	Value of n:	76.660000	45.560000	-56.547890
2	Decimal places:	2	0	3
3	Function used:	ROUND(B1,B2)	ROUND(C1,C2)	ROUND(D1,D2)
4	Value of ROUND(n,decimal places):	76.66	46	-56.548

આકૃતિ 7.15 : ROUND વિધેયનાં ઉદાહરણો

ROUND વિધેય જેવાં જ અન્ય બે વિધેયો ROUNDDOWN અને ROUNDUP છે જેની વાક્યરચના ROUND જેવી જ છે. આ વિધેયો ઉપર વિવિધ સંખ્યાઓ લઈ પ્રયોગો કરી જુઓ અને તેમની વચ્ચેનો તફાવત જાણો.

ટ્રન્કેટ (Truncate – સ્થૂળમૂલ્ય) (TRUNC વિધેય)

ટ્રન્કેટ વિધેય(TRUNC)માં બે ચલ હોય છે, જેમાં પહેલો ચલ એ સંખ્યા છે અને બીજો ચલ એ સંખ્યાના અપૂર્ણાંક ભાગમાંથી કેટલા અંક સંખ્યા સાથે કાપવાના છે એ સંખ્યા દર્શાવે છે. ટ્રન્કેટ વિધેય જો જણાવેલ હોય તો દશાંશબિંદુ પછી કેટલાક અંક રાખીને આપેલી સંખ્યાના અપૂર્ણાંક ભાગને કાપી નાખે છે. જો સંખ્યામાં દશાંશબિંદુ પછી y અંક હોય અને TRUNC વિધેયમાં બીજું ચલ ન આપેલું હોય તો આપેલી સંખ્યામાંથી આખો જ અપૂર્ણાંક ભાગ કાપી નાખવામાં આવે છે. આ વિધેયની વાક્યરચના **TRUNC(Number,Places)** છે, જ્યાં Number એ આપેલી સંખ્યા છે અને Places એ દશાંશબિંદુ પછી કેટલા અંક રાખવા છે તે જણાવે છે.

નીચે આપેલાં ઉદાહરણોનો અભ્યાસ કરો :

=TRUNC(1.239,2)નું પરિણામ 1.23 મળે છે અને અંક 9 ગુમાવીએ છીએ.

=TRUNC(12.5)નું પરિણામ 12 મળે છે અને અંક 5 ગુમાવીએ છીએ.

=TRUNC(-15.72)નું પરિણામ -15 મળે છે અને અંક 7 અને 2 ગુમાવીએ છીએ.

ઉપરનાં ઉદાહરણમાં TRUNC વિધેયની જગ્યાએ INT વિધેય વાપરી જુઓ અને પરિણામના તફાવત વિશે વિચાર કરો. તમે જોઈ શકશો કે TRUNC વિધેય આપેલી સંખ્યામાંથી નિર્દેશિત ભાગ ફક્ત કાપી જ નાખે છે. તે સંખ્યાને રાઉન્ડ કરશે નહીં. આ ઉદાહરણો આકૃતિ 7.16માં દર્શાવ્યા છે.

	A	B	C	D
1	Value of n:	1.239	12.5	-15.72
2	Decimal places:	2	--	--
3	Function used:	TRUNC(B1,B2)	TRUNC(C1)	TRUNC(D1)
4	Value of TRUNC(n,decimal places):	1.23	12.00	-15.00

આકૃતિ 7.16 : TRUNC વિધેયનાં ઉદાહરણો

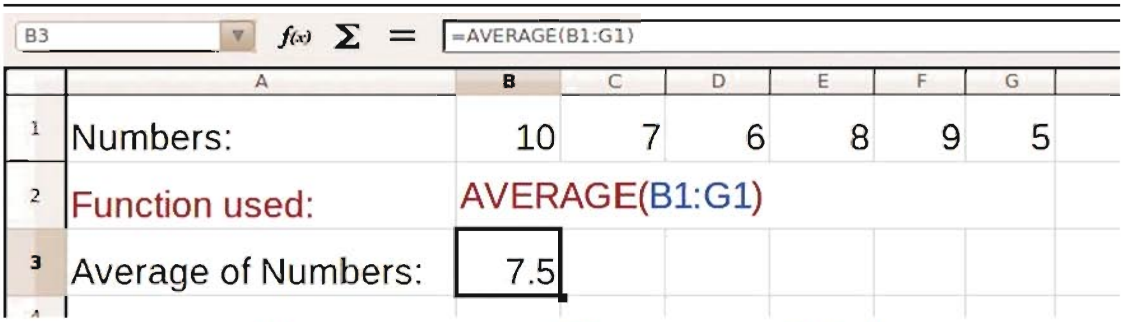
ROUND વિધેયમાં આપેલાં ઉદાહરણોનો પણ અભ્યાસ કરો અને એ જ ઉદાહરણોનો TRUNC વિધેય સાથે પ્રયોગ કરી જુઓ.

આંકડાકીય વિધેયો (Statistical functions)

કેલ્સી લાક્ષણિક ગાણિતિક વિધેયો ઉપરાંત કિંમતોની શ્રેણી ઉપર અનેક વિવિધ આંકડાકીય વિધેયો પણ તમને પૂરા પાડે છે. આ આંકડાકીય વિધેયો ડેટાનો સંચય કરવામાં, તેનું વિશ્લેષણ કરવામાં તેમજ તેને સમજવામાં તથા રજૂઆત કરવામાં મદદરૂપ થાય છે જેથી આગાહી કે ભવિષ્યવાણી કરવાનું અને નિર્ણયો લેવાનું કાર્ય સરળ બની રહે છે. ચાલો, હવે આપણે કેટલાક પ્રચલિત આંકડાકીય વિધેયોનો અભ્યાસ કરીએ.

સરેરાશ (Average) – AVERAGE વિધેય

AVERAGE વિધેય આપેલી સંખ્યાઓની સરેરાશ કિંમત પાછી આપે છે. AVERAGE એ અંકગણિતના મધ્યક તરીકે પણ ગણવામાં આવે છે. તમારે યાદીમાં રહેલી બધી સંખ્યાઓનો સરવાળો કરી તેને યાદીમાં રહેલી કુલ સંખ્યા (યાદીનું કદ) વડે ભાગાકાર કરવો પડે. તમે આવી મહત્તમ 30 કિંમતો લઈ શકો જેને અર્ધવિરામ(;)થી જુદી પાડવામાં આવે છે. હવે તમે જાણો છો કે કિંમતોને બદલે તમે સેલ સ્થાનાંક અથવા માન્ય સેલનો વિસ્તાર પણ દાખલ કરી શકો. આ વિધેયની વાક્યરચના **AVERAGE(x1,x2,...,x30)** છે. ઉદાહરણ તરીકે, જો તમે કોઈ સેલમાં **=AVERAGE(10,7,6,8,9,5)** લખો તો તમને જવાબ 7.5 મળશે. આકૃતિ 7.17માં AVERAGEનું કાર્ય દર્શાવેલું છે.



આકૃતિ 7.17 : AVERAGE વિધેયનું ઉદાહરણ સાથે નિદર્શન

મધ્યક (Mean) – GEOMEAN વિધેય

GEOMEAN વિધેય આપેલી સંખ્યાઓની સમગુણોત્તર મધ્યક (geometric mean) કિંમત પાછી આપે છે. ઋણ ન હોય તેવી n સંખ્યાઓનો સમગુણોત્તર મધ્યક મેળવવા માટે બધી જ n સંખ્યાઓનો ગુણાકાર કરી પછી તેનું n^{th} મૂળ કાઢવામાં આવે છે. તમે મહત્તમ 30 ચલ (કિંમતો, સ્થાનાંક અથવા વિસ્તાર) લઈ શકો છો. આ વિધેયની વાક્યરચના **GEOMEAN(x1, x2 ,..., x30)** છે. જો તમે સ્પ્રેડશીટના કોઈ સેલમાં **=GEOMEAN(10,7,6,8,6,5)** લખો તો 6.8 જવાબ મળશે.

કેલ્સીમાં અન્ય પ્રકારનું મધ્યક પણ ઉપલબ્ધ છે જે HARMEAN (હરાત્મક મધ્યક harmonic mean) છે. હરાત્મક મધ્યકની વાક્યરચના **HARMEAN(x1,x2,...,x30)** છે. ઉપર જણાવેલા ડેટાનું હરાત્મક મધ્યક ઉપરનું વિધેય વાપરીને શોધો.

મધ્યસ્થ (Median) – MEDIAN વિધેય

મધ્યસ્થ એ આપેલી શ્રેણીને ક્રમમાં ગોઠવ્યા પછીની મધ્ય સંખ્યા છે. જો તમે એક નિશ્ચિત મધ્ય સંખ્યા ન મેળવી શકો (શ્રેણીમાં કુલ બેકી સંખ્યામાં ચલ હોય ત્યારે) તો વચ્ચેની બે સંખ્યાની સરેરાશ લો. MEDIAN વિધેય મહત્તમ 30 ચલનું મધ્યસ્થ શોધી શકે છે. વિધેયની વાક્યરચના **MEDIAN(x1,x2,...,x30)** છે. જો સ્પ્રેડશીટના કોઈ સેલમાં તમે **=MEDIAN(10,7,6,8,6,5)** લખો તો 6.5 જવાબ મળશે.

બહુલક (Mode) – MODE વિધેય

MODE વિધેય આપેલા મહત્તમ 30 ચલ કે જે ફક્ત એક કિંમત અથવા સેલનો વિસ્તાર હોઈ શકે, તે ડેટા સેટની સૌથી વધારે સામાન્ય કિંમત પાછી આપે છે. તેની વાક્યરચના MODE(x1, x2,...,x30) છે. દા.ત. જો તમે MODE વિધેયને =MODE(10,7,6,8,6,5) લખશો તો તે 6 જવાબ આપશે. આકૃતિ 7.18માં વિવિધ આંકડાકીય વિધેયો દર્શાવેલા છે.

	A	B	C	D	E	F	G
1	Numbers:	10	7	6	8	6	5
2	Function used:	AVERAGE(B1:G1)					
3	Average of Numbers:	7					
4							
5	Function used:	MODE(B1:G1)					
6	Mode of Numbers:	6					
7							
8	Function used:	GEOMEAN(B1:G1)					
9	Geo.Mean of Numbers:	6.822					
10							
11	Function used:	MEDIAN(B1:G1)					
12	Median of Numbers:	6.5					

આકૃતિ 7.18 : GEOMEAN, MEDIAN અને MODE વિધેયો

CountA વિધેય

COUNTA વિધેય કોઈ કિંમતનો સંગ્રહ કુલ કેટલા સેલમાં અથવા ચલમાં કરવામાં આવ્યો છે તેની સંખ્યા આપે છે. તમે મહત્તમ 30 ચલ (કિંમતો, સ્થાનાંક અથવા વિસ્તાર) અર્ધવિરામનાં ચિહ્ન(;)થી અલગ પાડીને આપી શકો છો. તેમાંથી કેટલાક ચલમાં અમુક કિંમત (કોઈ પણ પ્રકારની) હોઈ શકે અને કેટલાક ખાલી પણ હોઈ શકે. COUNTA વિધેય અમુક કિંમત ધરાવતાં સેલની કુલ સંખ્યા આપે છે. ખાલી સેલની ગણતરી તેમાં થશે નહીં. આ વિધેયની વાક્યરચના **COUNTA(x1, x2, ... , x30)** છે. જો તમે સેલમાં **=COUNTA(10,7,6,8,6,5)** લખશો તો જવાબ 6 મળશે.

મોટામાં મોટી કિંમત (Largest value) – LARGE વિધેય

LARGE વિધેય આપેલી x સંખ્યાના ગણમાંથી k^{th} સૌથી મોટી આંકડાકીય કિંમત પાછી આપે છે. તેની વાક્યરચના **=LARGE(x, k)** છે, જ્યાં x એ અમુક કિંમતો ધરાવતી સેલ વિસ્તારનો નિર્દેશ કરે છે અને k એ સંખ્યાનું સ્થાન દર્શાવે છે.

આ વિધેય આપેલી સંખ્યાઓને ઊતરતા ક્રમમાં ગણીને k સ્થાનની સંખ્યા પાછી આપે છે. ઉદાહરણ તરીકે, **10, 7, 6, 8, 6 અને 5**ની કિંમતોના ગણમાં LARGE વિધેયમાં 2 કિંમત આપીએ તો આપણે જવાબ 8 મેળવીશું. આ સંખ્યા 8 એ આપેલી સંખ્યાનાં ગણમાં બીજી (2nd) મોટી કિંમત છે.

અહીં વિધેયમાં ચલ kની કિંમત તમે ઇચ્છિત પૂર્ણાંક સંખ્યા આપી શકો સિવાય કે તે શૂન્ય ન હોઈ શકે અથવા આપેલી સંખ્યાના ગણની કુલ સંખ્યા કરતાં વધારે ન હોઈ શકે.

નાનામાં નાની કિંમત (Smallest value) – SMALL વિધેય

SMALL વિધેય આપેલી x સંખ્યાના ગણમાંથી k^{th} નાનામાં નાની આંકડાકીય કિંમત પાછી આપે છે. તેની વાક્યરચના **=SMALL (x, k)** છે, જ્યાં x એ અમુક કિંમતો ધરાવતી સેલ વિસ્તારનો નિર્દેશ કરે છે અને k એ સંખ્યાનું સ્થાન દર્શાવે છે.

આ વિધેય આપેલી સંખ્યાઓને ચઢતા ક્રમમાં ગણીને k સ્થાનની સંખ્યા પાછી આપે છે. ઉદાહરણ તરીકે, **10, 7, 6, 8, 6, 5**ની કિંમતોના સેટમાં SMALL વિધેયમાં 4 કિંમત આપીએ તો જવાબ 7 મેળવીશું કે જે આપેલી સંખ્યાઓનાં સેટમાં ચોથી નાનામાં નાની સંખ્યા છે. આકૃતિ 7.19માં LARGE, SMALL અને COUNTA વિધેય ઉદાહરણો સાથે દર્શાવ્યા છે.

	A	B	C	D	E	F	G
1	Numbers:	10	7	6	8	6	5
2	Function used:	COUNTA(B1:G1)					
3	Count of Numbers:	6					
4							
5	Function used:	LARGE(B1:G1;2)					
6	Second Largest Value:	8					
7							
8	Function used:	SMALL(B1:G1;4)					
9	Fourth Smallest Value:	7					
10							

આકૃતિ 7.19 : LARGE, SMALL અને COUNTA વિધેયોનાં ઉદાહરણો

રૅન્ક (Rank) – RANK વિધેય

રૅન્ક વિધેય કોઈ આપેલી સંખ્યાઓના ગણમાં કોઈ નક્કી કરેલી સંખ્યાનો ક્રમ પરત આપે છે. આ માટે તમારે કોઈ કિંમત (અથવા કિંમતનો સ્થાનાંક) કે જેનો ક્રમ તમારે જાણવો છે, સંખ્યાઓનો ગણ (સેલનો વિસ્તાર) અને સંખ્યાઓને ગોઠવવાનો ક્રમ; ચઢતા ક્રમ માટે 1 અને ઊતરતા ક્રમ માટે 0 જણાવવો પડે છે. આ વિધેય આપેલી સંખ્યાઓના સેટને ચઢતા કે ઉતરતા ક્રમમાં ગોઠવશે અને આપણે જણાવેલી સંખ્યાનું સ્થાન (પહેલું, બીજું, ત્રીજું... વગેરે) કેટલામું છે તે કિંમત પરત આપે છે.

આ વિધેયની વાક્યરચના **RANK(number, set, order)** છે. ધારો કે તમે પરીક્ષામાં 67 ગુણ મેળવ્યા છે. તમારા પાંચ મિત્રોએ 47, 56, 78, 59 અને 66 ગુણ મેળવ્યા છે. તમે જાણો છો કે તમારા મિત્રોમાં તમારો રૅન્ક બીજો છે; સૌથી વધારે ગુણનો પહેલો નંબર એટલે કે ગુણ ઊતરતા ક્રમમાં છે. આ માહિતી તમે ડેટા તરફ ફક્ત નજર કરીને જ મેળવી શકો કારણ કે તમે ફક્ત તમારા મિત્રોની જ (છ મિત્રોની) માહિતી ધ્યાનમાં લીધી છે. પણ વિચારો કે જો ઘણા વિદ્યાર્થીઓની માહિતી હોય તો શું થાય ? જવાબ ઘણો સહેલો છે. આ માટે RANK વિધેય વાપરો. ઉદાહરણ તરીકે, તમારો નંબર જાણવા માટે (દેખીતી રીતે ઉપરથી), તમે RANK વિધેય **=RANK (67, A1:F1, 0)** વાપરી શકો અને જવાબ તમારો નંબર 2 હશે. આકૃતિ 7.20માં RANK વિધેય દર્શાવ્યું છે.

મહત્તમ (Maximum) - MAX વિધેય

MAX વિધેયમાં વધારેમાં વધારે 30 ચલ (એક સેલ અથવા સેલનો વિસ્તાર) આપી શકાય અને આ ચલમાંથી મોટામાં મોટી સંખ્યા (મહત્તમ કિંમત) પાછી મળે છે.

આ વિધેયની વાક્યરચના **MAX(x1, x2 ,..., x30)** છે. ઉપર જણાવેલા વિદ્યાર્થીઓના ગુણના ઉદાહરણમાં મહત્તમ ગુણ શોધવા માટે તમે MAX વિધેયનો ઉપયોગ **=MAX(67,47,56,78,59,66)** આપીને કરી શકો છો. આકૃતિ 7.20માં જણાવ્યા પ્રમાણે તે વિધેય 78 જવાબ પરત આપશે.

	A	B	C	D	E	F	G
1	Numbers:	67	47	56	78	59	66
2	Function used:	RANK(67,B1:G1,0)					
3	RANK of 67:	2					
4							
5	Function used:	MAX(B1:G1)					
6	Largest Value:	78					
7							
8	Function used:	MIN(B1:G1)					
9	Smallest Value:	47					
10							

આકૃતિ 7.20 : RANK, MAX અને MIN વિધેયનો ઉપયોગ

લઘુતમ (Minimum) - MIN વિધેય

MIN વિધેયમાં વધારેમાં વધારે 30 ચલ (એક સેલ અથવા સેલ વિસ્તાર) આપી શકાય અને આ ચલમાંથી નાનામાં નાની સંખ્યા (લઘુતમ સંખ્યા) પાછી મળે છે. આ વિધેયની વાક્યરચના **MIN(x1, x2 ,..., x30)** છે. ઉપર જણાવેલા વિદ્યાર્થીઓના ગુણના ઉદાહરણમાં લઘુતમ ગુણ શોધવા માટે તમે MIN વિધેયનો ઉપયોગ **=MIN(67,47,56,78,59,66)** આપીને કરી શકો છો. આકૃતિ 7.20માં દર્શાવ્યા પ્રમાણે તે વિધેય 47 જવાબ પરત આપશે.

નાણાંની ગણતરી (Calculation with money)

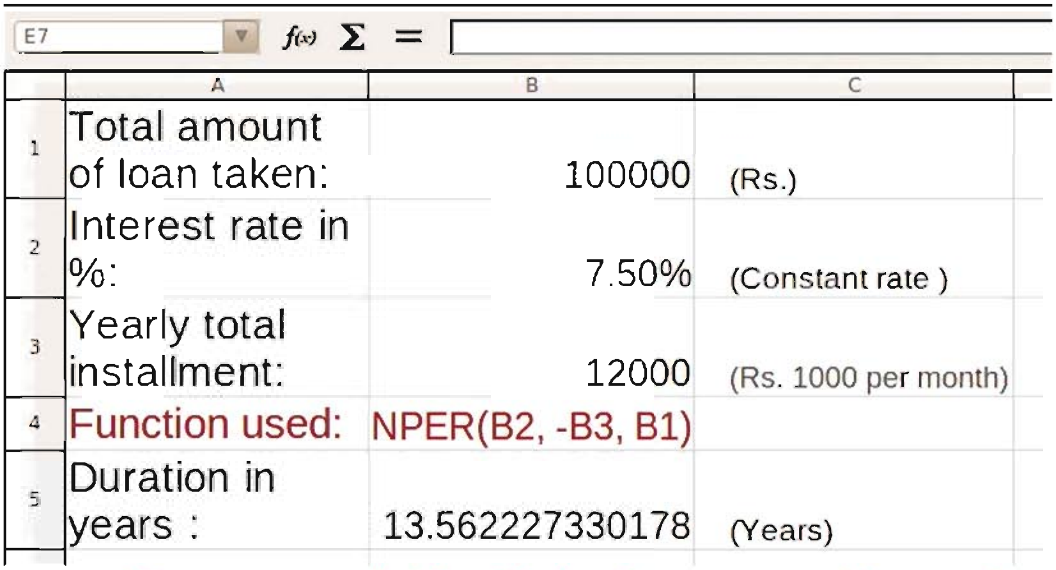
ચલણી નાણાંની ગણતરી માટે કેલ્સી નાણાંકીય વિધેયો પૂરાં પાડે છે. ધારો કે તમારા એક કાકા તમારી પાસે આવે છે અને કહે છે :

> *"બેટા, તારા કમ્પ્યૂટરનો ઉપયોગ કરીને મને કહે કે જો કોઈ બૅંકમાંથી હું નાણાંનું ઋણ લઉં અને તેને ચૂકવવા માટે દરેક સમયે એકસરખા હપ્તા ચૂકવું અને વ્યાજનો દર એકસરખો રહે તો કેટલાં વર્ષનો સમય લાગશે ?"*

આ કાર્ય માટે તમારે ઘણી ગણતરી કરવી પડશે અથવા એક અટપટો કમ્પ્યૂટર પ્રોગ્રામ લખવો પડશે. કેલ્સી તમારું કામ સરળ બનાવી દે છે. તમારે ફક્ત સેલમાં એક વિધેય લખવું પડે છે :

=NPER(7.5%,-12000,100000)

અહીં NPERનો ઉપયોગ તમે ઉછીના લીધેલા પૈસાની ચૂકવણી માટેના સમયગાળાની ગણતરી માટે થયો છે. આ વિધેયનું પહેલું ચલ (7.5%) એ અચળ વ્યાજનો દર છે. બીજું ચલ -12000 એ વાર્ષિક હપ્તાની રકમ (રૂપિયામાં) છે (તે ઋણ છે કારણ કે દરેક હપ્તો ચૂકવવામાં આવે ત્યારે કુલ ઉછીની લીધેલી રકમમાંથી આ રકમ બાદ કરવી પડે); અહીં તમારા કાકા 1000 રૂ. પ્રતિ માસ પાછા ચૂકવે છે અને વર્ષમાં કુલ 12000 રૂ. પાછા આપે; ત્રીજું ચલ એ કુલ ઉછીની લીધેલી રકમ છે. જ્યારે તમે આ સૂત્ર કેલ્સીના કોઈ સેલમાં લખો છો ત્યારે તે 13.56 વર્ષ જવાબ પાછો આપશે. આ એ જ જવાબ છે જે તમારા કાકા જાણવા ઇચ્છતા હતા. આ ઉદાહરણ આકૃતિ 7.21માં દર્શાવેલ છે.



આકૃતિ 7.21 : NPER વડે ઉછીના લીધેલાં નાણાં પાછા ચૂકવવા માટેનો સમયગાળો

NPER વિધેયની વાક્યરચના **NPER(Interest,Instalment,Loan,Future,Type)** છે, અહીં interest - વ્યાજનો દર, Instalment - હપ્તાની રકમ, Loan - ઉછીની લીધેલી રકમ, Future - છેલ્લો હપ્તો ચૂકવ્યા પછી રહેલી રકમ (સામાન્ય રીતે આપણે ઉછીની લીધેલી રકમની પૂરેપૂરી ચૂકવણી કરીએ છીએ) અને Type હપ્તાના સમયગાળાની પહેલાં ચૂકવણી કરવામાં આવે છે કે નહીં, (જો સમયગાળાની પહેલાં હપ્તો આપવાનો હોય તો તેની કિંમત (typeની કિંમત) એક હોય છે, અન્યથા તે શૂન્ય રાખવામાં આવે છે.) અહીં Future અને Type ચલ એ મરજિયાત છે અને તમે જોઈ શકશો કે આપણે ઉપરનાં ઉદાહરણમાં આ બંને ચલનો ઉપયોગ કરેલો નથી કારણ કે આપણે પૂરેપૂરી રકમની ચૂકવણી કરવા ઇચ્છીએ છીએ અને સમયગાળાની પહેલાં સામાન્ય રીતે આપણે હપ્તાની ચૂકવણી કરીએ છીએ, જેમાં નિષ્ફળ જતાં આપણે વધારે વ્યાજ ચૂકવવું પડે છે.

કેલ્સી નાણાં સંચાલનનાં કાર્ય માટે આ પ્રકારના અન્ય અનેક વિધેયો પૂરાં પાડે છે. ઉદાહરણોમાં વ્યાજની ગણતરી, ભવિષ્યની અમુક રકમની આજની કિંમત (net present value), ભવિષ્યની રકમ (Future value) અને રકમ પાકવાની તારીખ છે. તમે કેલ્સીની helpનો ઉપયોગ કરો અને તેના વિશે વધારે વિગત મેળવો.

નિર્ણયો લેવા (Making Decisions)

આપણે અત્યાર સુધી ચર્ચા કરેલાં વિધેયો ફક્ત કેટલાક ચલનો ઉપયોગ કરીને, તેના ઉપર પ્રક્રિયા કરીને કોઈ કિંમત પાછી આપતાં હતાં. તે ખરેખર આપણા માટે યોગ્ય કામ કરે છે પણ છતાં તે વિધેયો એટલાં ચાલાક નથી. જો વિધેય કેટલાક પ્રકારના નિર્ણયો લેવા સક્ષમ હોય તો તે વિધેય ચાલાક કહેવાય. આવા પ્રકારનાં કાર્ય માટે આપણી પાસે શરતી વિધેયો હોવાં જોઈએ કે જે પૂર્વનિર્ધારિત શરતોને આધીન નિર્ણય લઈ શકે. આ માટેનું આદર્શ ઉદાહરણ છે તમારી પરીક્ષાનું પરિણામ. નીચે આપેલું એક ઉદાહરણ લો જેમાં તમારા પરિણામની કક્ષા નક્કી કરવા માટે પૂર્વનિર્ધારિત શરતોની ચર્ચા કરેલી છે.

જો તમારા ગુણ >=35% અને < 48%; હોય તો તમે PASS વર્ગ મેળવો છો,

જો તમારા ગુણ >=48% અને < 55%; હોય તો તમે SECOND વર્ગ મેળવો છો,

જો તમારા ગુણ >=55% અને < 60%; હોય તો તમે FIRST વર્ગ મેળવો છો.

આ પ્રકારના અનેક સંજોગો હોય છે જેમાં આ જાતનાં વર્ણનાત્મક તર્ક અને અગાઉથી નક્કી કરેલી શરતોની જરૂર પડે છે. કેલ્સી આ પ્રકારનાં અનેક તાર્કિક વિધેયો પૂરાં પાડે છે જે તમારા કાર્યને સરળ બનાવી દે છે. ચાલો, આપણે આ પ્રકારના કેટલાક વિધેયોનો અભ્યાસ કરીએ.

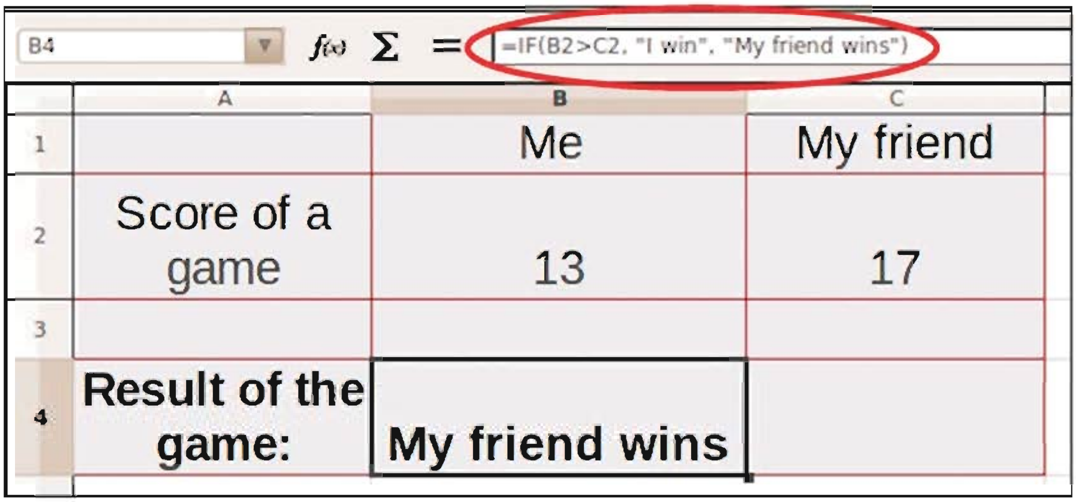
IF વિધેય વડે નિર્ણય કરવા (Making decisions with IF)

કેલ્સીમાં વપરાતા બધાં શરતી વિધેયોમાં સૌથી ધ્યાનાકર્ષક (પ્રખ્યાત) વિધેય IF છે. આ વિધેયમાં ત્રણ ચલ હોય છે : Test, True અને False. Test પદાવલિ / ચલ તમારે જે શરતની કસોટી કરવાની છે તેનો નિર્દેશ કરે છે. કેલ્સી પદાવલિની ગણતરી કરે છે અને જો પરિણામ સાચું હોય તો તે True વિભાગમાં દર્શાવેલી કિંમત પરત કરે છે અન્યથા False વિભાગમાં દર્શાવેલી કિંમત પરત કરે છે.

હવે તમે નીચે આપેલું ઉદાહરણ લો અને કેલ્સી વર્કશીટમાં પ્રયત્ન કરો :

- સેલ A2 અને B2માં અલગ અલગ કિંમત લખો.
- સેલ C2 માં જઈને નીચે આપેલું વિધેય લખો :
  =IF(A2>B2,"Value at A2 is greater", "Value at B2 is greater")
- હવે સેલ A2 અને B2ની કિંમતોમાં ફેરફાર કરો અને સેલ C2માં શું ફેરફાર થાય છે તેનો અભ્યાસ કરો.

આકૃતિ 7.22માં દર્શાવ્યા પ્રમાણે રમતમાં તમારા અને તમારા મિત્રના મેળવેલા ગુણ બે સેલમાં દાખલ કરેલા હોય તો દરેક રમતમાં કોણ વિજેતા થશે તે નિર્ણય કરવા માટે એક સૂત્ર તમે લખી શકો.



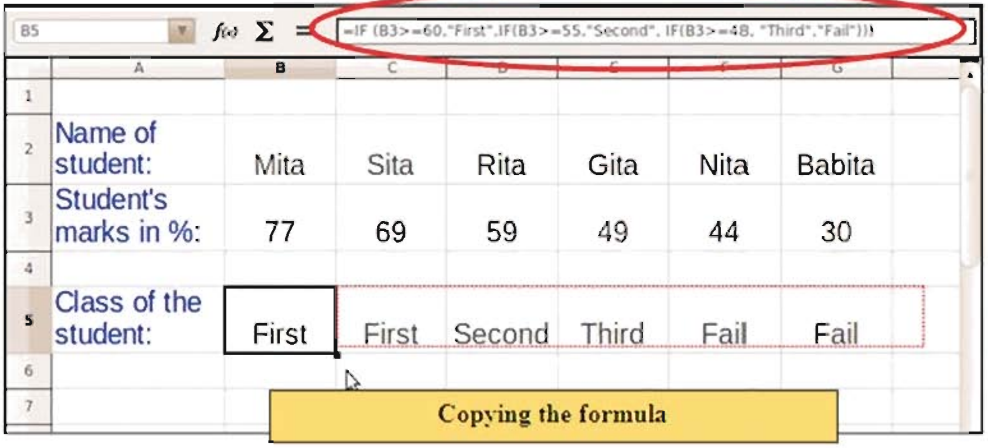
આકૃતિ 7.22 : વિજેતા કોણ છે ?

તમે ફૉર્મ્યુલા બારમાં (અંડાકારથી ચિહ્નિત) સૂત્ર જોઈ શકશો. ઉપર જણાવેલાં વિદ્યાર્થીના પરિણામના ઉદાહરણમાં એક કરતાં વધારે IF વિધેયની રચના કરવાની જરૂર પડે છે. આમાનું એક નીચે આપેલું છે.

ધારો કે સેલ A2 માં વિદ્યાર્થીના કુલ ગુણના ટકાનો સંગ્રહ કરેલો છે. આ માટે તમે IF વિધેય નીચે પ્રમાણે લખી શકો :

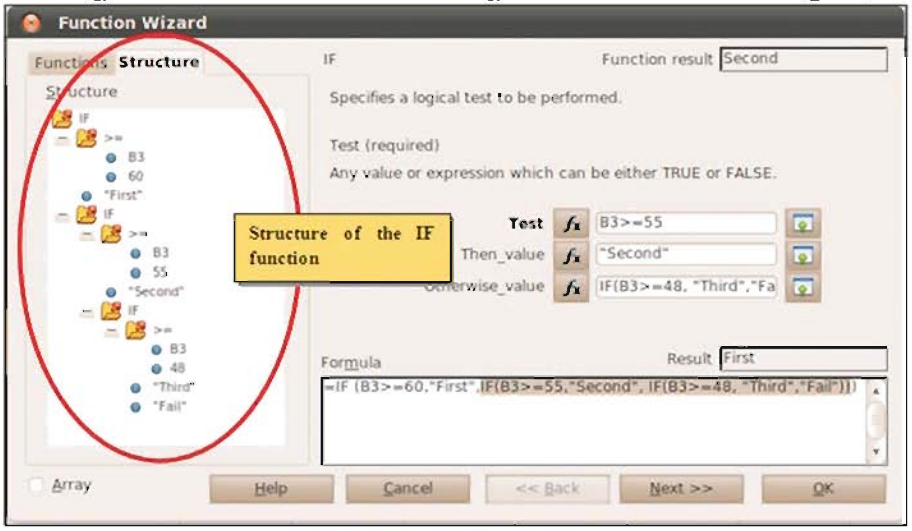
=IF(A2>=60,"First",IF(A2>=55,"Second",IF(A2>=48,"Third","Fail")))

ઉપરના વિધેયમાં, પહેલું IF વિધાન સેલ A2ની કિંમત ચકાસે છે; કે જે વિદ્યાર્થીના કુલ ગુણના ટકા છે. જો સેલ A2ની કિંમત 60 બરાબર કે વધારે હશે તો પદાવલિની કિંમત સાચી ગણાશે અને વિધેય પહેલી પદાવલિમાં જણાવેલી કિંમત પાછી આપશે, કે જે "First" છે. જો વિદ્યાર્થીનાં 60 કરતાં ઓછા ટકા હશે તો તે વિધેયની બીજી પદાવલિની કિંમત આપશે કે જે અન્ય IF વિધાન છે. આ રીતે તમે એક IFની અંદર અનેક IFનો સમાવેશ કરી શકો અને આથી તેને nested if કહેવામાં આવે છે. આકૃતિ 7.23માં આ દર્શાવેલું છે. આકૃતિમાં IF વિધાનની જુદી જુદી શક્યતાઓ દર્શાવવા માટે આપણે અનેક વિદ્યાર્થીઓનાં નામ અને તેણે મેળવેલા ટકા આપેલા છે. જો કે આપણે સૂત્ર ફક્ત એક જ વાર લખેલું છે. (જુઓ ફૉર્મ્યુલા બાર ઉપર નિશાની કરેલ છે) અને આકૃતિ 7.23માં દર્શાવ્યા પ્રમાણે તે સૂત્રને અન્ય વિદ્યાર્થીઓ માટે ડ્રેગ (નક્લ) કરેલું છે.



આકૃતિ 7.23 : IF વિધેયનું ઉદાહરણ

જ્યારે મોટા સૂત્ર લખવાનાં હોય છે ત્યારે તેના ઉપરની પકડ ઢીલી થઈ જવાની શક્યતા વધી જાય છે અને તે લખવામાં ભૂલ થઈ શકે છે. આ ટાળવા માટે તમે ફંકશન વિઝાર્ડની મદદ લઈ શકો છો. ફંકશન વિઝાર્ડ વાપરવાનો એક ફાયદો એ છે કે તમારે વિધેયનું સર્વથા ખરું નામ ટાઇપ કરવું પડતું નથી. એક વખત તમે ફંકશન વિઝાર્ડ જુઓ અને તેને વાપરી જુઓ, તમે Structure ટૅબ પસંદ કરો તેથી આકૃતિ 7.24માં દર્શાવ્યા પ્રમાણે તમે લખેલા સૂત્રનું બંધારણ જોવા મળશે કે જે વિધેય બનાવવાનું બંધારણ આબેહૂબ રીતે પ્રદર્શિત કરે છે. જો તેમાં કંઈ ભૂલો હશે તો તે યાદીમાં લાલ બિંદુ સ્વરૂપે દર્શાવે છે.



આકૃતિ 7.24 : વિધેયનું બંધારણ

સંબંધિત પ્રક્રિયકો (Relational operators)

ઉપરનાં ઉદાહરણમાં જ્યારે આપણે IF વિધાનની ચર્ચા કરતા હતા ત્યારે આપણે >= જેવાં પ્રક્રિયક (ઑપરેટર) જોયાં હતાં. આવાં પ્રક્રિયકો (ઑપરેટર્સ) રિલેશનલ ઑપરેટર કહેવાય છે. કોષ્ટક 7.1માં તમારે જરૂરી રિલેશનલ ઑપરેટરની યાદી આપેલી છે.

રિલેશનલ ઑપરેટર	ચિહ્ન	વર્ણન
Equal	=	બંને સંકાર્ય (operands) (જેમ કે A1 અને A2) સરખા છે. ઉદાહરણ =IF(A1=A2, "A1 is equal to A2", "A1 and A2 not equal")
Less than	<	પ્રથમ સંકાર્ય બીજા સંકાર્ય કરતાં નાનો છે. ઉદાહરણ =IF(A1<A2, "A1 is less than A2", "A1 is not less than A2")
Greater than	>	પ્રથમ સંકાર્ય બીજા સંકાર્ય કરતાં મોટો છે. ઉદાહરણ : =IF(A1>A2, "A1 is greater than A2", "A1 is not greater than A2")
Less than or equal to	<=	પ્રથમ સંકાર્ય બીજા સંકાર્ય કરતાં નાનો અથવા બરાબર છે. ઉદાહરણ : =IF(A1<=A2, "A1 is less than or equal to A2", "A1 is not less than or equal to A2")
Greater than or equal to	>=	પ્રથમ સંકાર્ય બીજા સંકાર્ય કરતાં મોટો અથવા બરાબર છે. ઉદાહરણ: =IF(A1>=A2, "A1 is greater than or equal to A2", "A1 is not greater than or equal to A2")
Different from	<>	બંને સંકાર્ય સરખા નથી, પ્રથમ સંકાર્ય બીજા સંકાર્ય બરાબર નથી. ઉદાહરણ : =IF(A1<>A2, "A1 is different than A2", "A1 is not different than A2")

કોષ્ટક 7.1 : વિવિધ રીલેશનલ ઑપરેટર (સંબંધિત પ્રક્રિયકો)

તાર્કિક વિધેયો (Logical functions)

કેલ્સી અન્ય કેટલાંક વિધેયો પૂરાં પાડે છે જે તાર્કિક નિર્ણયો લેવામાં મદદરૂપ થાય. આ વિધેયોમાં AND, OR અને NOTનો સમાવેશ થાય છે. આ વિધેયો અટપટાં સૂત્રો અને અન્ય વિધેયોની રચના કરવામાં ઉપયોગી બને છે. આ વિભાગમાં ઉપર જણાવેલા ત્રણ તાર્કિક વિધેયોની સમજ આપેલી છે.

TRUE

આ વિધેય આપણને જે કિંમત પાછી આપે છે, તે તાર્કિક કિંમત TRUE (ખરું) છે. આ વિધેયની વાક્યરચના **TRUE()** છે.

FALSE

આ વિધેય આપણને જે કિંમત પાછી આપે છે, તે તાર્કિક કિંમત FALSE છે. આ વિધેયની વાક્યરચના **FALSE()** છે.

NOT

આ વિધેયની વાક્યરચના **NOT (પદાવલિ કે જેનું પરિણામ તાર્કિક કિંમતમાં આવે)** છે. આ NOT વિધેય તાર્કિક કિંમતને તેનાથી વિપરીત બનાવે છે. આથી FALSE બદલાય ને TRUE બને છે અને TRUE બદલાય ને FALSE બને છે. નીચેનાં ઉદાહરણો જુઓ :

=NOT(6<>6)ની કિંમત TRUE મળે છે.

=NOT(1=1)ની કિંમત FALSE મળે છે.

આપણે NOT(TRUE) અને NOT(FALSE) લખી શકીએ ? પ્રયત્ન કરી જુઓ. આકૃતિ 7.25માં TRUE, FALSE અને NOT (FALSE) નાં કેટલાંક ઉદાહરણો આપેલાં છે.

	A	B
1		
2	Logical function	Value
3	TRUE()	TRUE
4	FALSE()	FALSE
5	NOT (TRUE)	FALSE
6	NOT (FALSE)	TRUE
7	NOT (6<>6)	TRUE
8	NOT (1=1)	FALSE
9		

આકૃતિ 7.25 : TRUE, FALSE અને NOT વિધેયોનાં ઉદાહરણો

AND વિધેય

આ વિધેય વધારેમાં વધારે 30 ચલ(arguments)નાં પરિણામોને સરખાવવા માટે વપરાય છે. જો બધી જ શરતો / ચલનું પરિણામ TRUE આવશે તો જ ANDનું પરિણામ TRUE મળશે. આ વિધેયની વાક્યરચના **AND(Cond_1,Cond_2,...,Cond_30)** છે.

અહીં AND વિધેયના બધા ચલ એ શરતો છે જેનું પરિણામ TRUE અથવા FALSE માં આવી શકે, એટલે કે બૂલિયન કિંમતમાં (Boolean values) જવાબ મળે. હવે આ ઉદાહરણ લો. જો આપણે જાણવું હોય કે સેલ A1, A2 અને A3ની કિંમતો સરખી છે કે નહીં તો આપણે એ જાણવું જોઈએ કે A1 અને A2 સરખા છે કે નહીં અને પછી આપણે જાણીએ કે A2 અને A3 સરખા છે કે નહીં. જો બંનેના પરિણામ સાચાં હોય તો આપણે નક્કી કરી શકીએ કે ત્રણેની કિંમત સરખી છે. એટલે કે, આપણે "=" રિલેશનલ ઑપરેટર વાપરીને બે શરતો બનાવવી પડે કે જે A1=A2 અને A2=A3 છે . આથી, AND વિધેય **=AND(A1=A2,A2=A3)** લખી શકાય. ઉપર જે ચર્ચા કરી તે આકૃતિ 7.26માં દર્શાવેલી છે.

C5 | f(x) Σ = | =AND(A1=A2,A2=A3)

	A	B	C
1	12		
2	12		
3	6		
4			
5	Are the above values same? :		FALSE
6			

આકૃતિ 7.26 : AND વિધેયનું ઉદાહરણ

OR વિધેય

આ વિધેયની વાક્યરચના **OR(Cond_1,Cond_2,...,Cond_30)** છે.

આ OR વિધેયમાં વધારેમાં વધારે 30 શરતોના (arguments)નાં પરિણામોને સરખાવવા માટે થાય છે. જો આપેલી શરતોમાંથી ઓછામાં ઓછી એક શરતનું પરિણામ TRUE હશે તો OR આપણને TRUE કિંમત પાછી આપશે. ઉદાહરણ તરીકે, આપણે ઉપર વર્ણવેલા તાર્કિક AND વિધેયનું સૂત્ર લઈએ. આપણે સૂત્રમાં થોડો ફેરફાર કરીએ. અગાઉ આપણે ત્રણે ચલ A1, A2 અને A3 સરખા છે તેમ ઇચ્છતા હતા. હવે આપણે ત્રણમાંથી કોઈ પણ બે ચલની કિંમત સરખી હોય તો આપણે આગળ વધવા ઇચ્છીએ છીએ. આ માટે **=OR(A1=A2,A2=A3,A1=A3)** લખી શકાય.

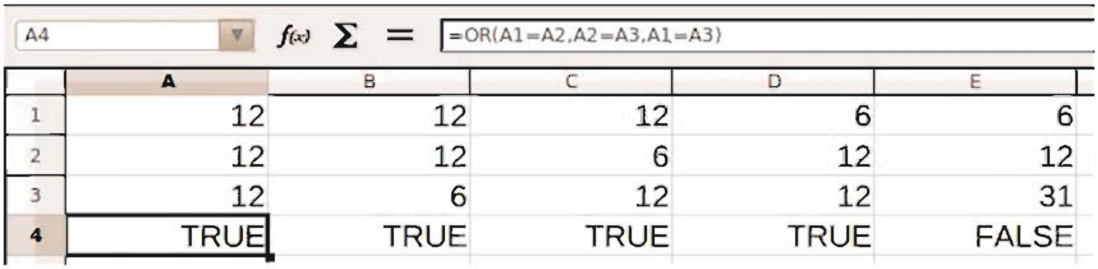
નીચેનાં સંજોગોમાં ઉપરનું વિધેય TRUE કિંમત પાછી આપશે :

- જો A1 બરાબર A2 હોય;
- જો A2 બરાબર A3 હોય;
- જો A1 બરાબર A3 હોય; અથવા
- જો A1, A2 અને A3 ની કિંમત સરખી હોય.

જો ઉપરના બધાં જ વિધાનો ખોટાં હોય તો વિધાન FALSE કિંમત પાછી આપશે. એટલે કે નીચેનાં બધાં વિધાનો સાચા હોવાના સંજોગોમાં વિધાન FALSE કિંમત આપશે.

- A1 બરાબર A2 નથી; A2 બરાબર A3 નથી; અને A1 બરાબર A3 નથી; જો આ બધા વિધાનો સાચાં હોય તો જ પરિણામ FALSE મળશે.

એટલે કે જો વિધેયમાં આપેલી બધી શરતો FALSE હોય તો જ OR વિધેય FALSE કિંમત પાછી આપશે. આકૃતિ 7.27માં OR વિધેયનાં કેટલાંક ઉદાહરણો આપેલાં છે અને કોષ્ટક 7.2માં તાર્કિક વિધેયનાં કેટલાંક ઉદાહરણો આપેલાં છે.


A4 | =OR(A1=A2,A2=A3,A1=A3)

	A	B	C	D	E
1	12	12	12	6	6
2	12	12	6	12	12
3	12	6	12	12	31
4	TRUE	TRUE	TRUE	TRUE	FALSE

આકૃતિ 7.27 : OR વિધેયનાં ઉદાહરણ

ઉદાહરણો			પદાવલિની કિંમત	પરિણામ
A1ની કિંમત	A2ની કિંમત	A3ની કિંમત		
12	12	12	A1=A2 → TRUE A2=A3 → TRUE A1=A3 → TRUE	TRUE
12	12	6	A1=A2 → TRUE A2=A3 → FALSE A1=A3 → FALSE	TRUE
12	6	12	A1=A2 → FALSE A2=A3 → FALSE A1=A3 → TRUE	TRUE
6 6	12 12	12 31	A1=A2 → FALSE A2=A3 → TRUE A1=A3 → FALSE A1=A2 → FALSE	TRUE **FALSE**
			A2=A3 → FALSE A1=A3 → FALSE	

કોષ્ટક 7.2 : તાર્કિક વિધેય OR(A1=A2;A2=A3;A1=A3)નાં વિવિધ ઉદાહરણો

સ્ટ્રિંગ ફંક્શન (String functions) – શાબ્દિક લખાણ ઉપરનાં વિધેયો

શાબ્દિક લખાણની બે સ્ટ્રિંગ જોડવી (Joining two strings of text)

CONCATENATE વિધેય શાબ્દિક લખાણની વધારેમાં વધારે 30 સ્ટ્રિંગ (string) ને જોડી શકે છે. આ વિધેયની વાક્યરચના **CONCATENATE(text1,text2,...,text30)** છે.

તમે CONCATENATE વિધેયનાં નામને બદલે '&' ચિહ્ન પણ વાપરી શકો છો. એટલે કે "ice" અને "cream" જેવી બે સ્ટ્રિંગને જોડવા માટે તમે વિધેયને નીચે મુજબ લખી શકો :

="ice"& "cream" અથવા

=CONCATENATE("ice","cream")

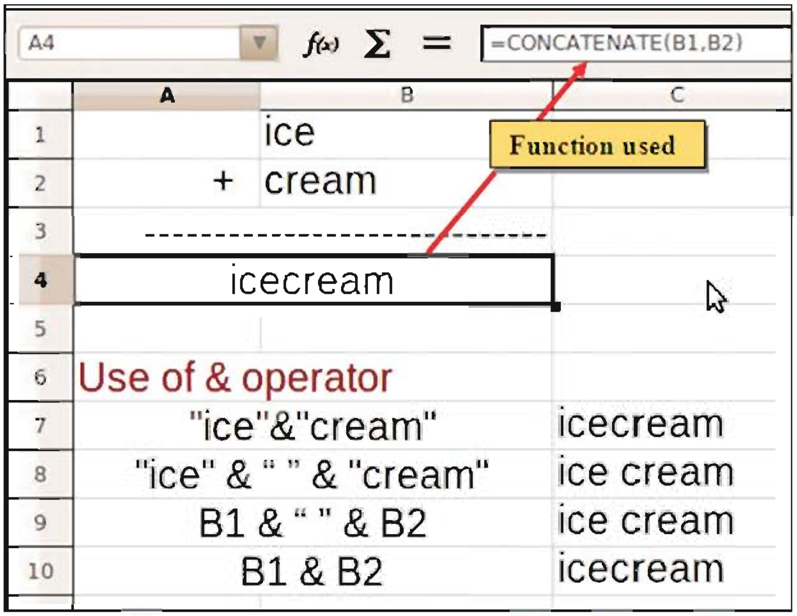
ઉપરનાં બંને વિધેયોનું પરિણામ સ્ટ્રિંગ "icecream" મળશે. જો બંને શબ્દો વચ્ચે એક ખાલી જગ્યા (space) તમારે જોઈતી હોય તો તમે બેમાંથી કોઈ પણ એક સ્ટ્રિંગમાં તમે ઉમેરી શકો જેમ કે **="ice "& "cream"** અથવા **="ice"& " cream"**

જે સ્ટ્રિંગ તમારે જોડવાની છે તેનો સંગ્રહ અલગ અલગ સેલ(ધારો કે A1 અને A2)માં કરેલો હોય તો તમે **=A1&A2** અથવા **=CONCATENATE(A1,A2)** પણ લખી શકો.

નીચે આપેલો સ્વાધ્યાય કરો :

- સેલ A1 ઉપર જઈ તમારું પ્રથમ નામ લખો.
- સેલ A2 ઉપર જઈ તમારા પિતાનું નામ લખો.
- સેલ A3 ઉપર જઈ તમારી અટક લખો.
- સેલ A4 ઉપર જઈ એક ખાલી જગ્યા " " લખો.
- & વિધેયનો ઉપયોગ કરી સેલ C3માં **=A1&A4&A2&A4&A3**

આકૃતિ 7.28માં CONCATENATION વિધેયનાં કેટલાંક ઉદાહરણો આપેલાં છે.



આકૃતિ 7.28 : CONCATENATE વિધેયનાં ઉદાહરણો

LOWER વિધેય

આ વિધેય આપેલા બધા અક્ષરોને નાના મૂળાક્ષરો(lower case)માં ફેરવે છે. આ વિધેયની વાક્યરચના **LOWER (શાબ્દિક લખાણ)** છે. ઉદાહરણ : **=LOWER("School")** આપણને જવાબમાં કિંમત "school" પરત આપે છે.

Upper વિધેય

આ વિધેય આપેલા બધા અક્ષરોને મોટા મૂળાક્ષરો(કેપિટલ અક્ષરો upper case)માં ફેરવે છે. આ વિધેયની વાક્યરચના **UPPER(text)** છે. ઉદાહરણ : **=UPPER("School")** આપણને જવાબમાં કિંમત "SCHOOL" આપે છે.

Proper વિધેય

આ વિધેય બધા શબ્દોનો પહેલો અક્ષર મોટા મૂળાક્ષરમાં અને બાકીના અક્ષરો નાના મૂળાક્ષરોમાં ફેરવે છે. આ વિધેયની વાક્યરચના **PROPER(text)** છે. ઉદાહરણ : **=PROPER ("I love my scHool")** ની કિંમત **I Love My School** પાછી આપે છે.

સામાન્ય રીતે LOWER, UPPER અને PROPER વિધેયો દસ્તાવેજને આકર્ષક સ્વરૂપ આપવા માટે (ફૉર્મેટ કરવા માટે) વપરાય છે. યોગ્ય રીતે ફૉર્મેટ કરેલા દસ્તાવેજની સુવાચ્યતા અને સ્પષ્ટતા ઘણી સારી હોય છે.

રોમન અને અરેબિક અંક (Roman and Arabic) માટેનાં વિધેય

ROMAN વિધેય આપેલી સંખ્યાને રોમન અંકમાં ફેરવે છે. તેની વાક્યરચના ROMAN(number, mode) છે.

અહીં એ નોંધ કરો કે 0થી 3999ની સંખ્યા જ રોમન અંકમાં રજૂ કરી શકાય. આથી આપણે ચલમાં આપેલી સંખ્યાની કિંમત આ હદમાં જ હોવી જોઈએ. તમે જાણો છો કે રોમન સંખ્યા ઘણી જુદી જુદી રીતે લખી શકાય છે. ધારો કે સંખ્યા 8ને તમારે રોમનમાં લખવી છે. તમે આ સંખ્યા VIIIની રીતે લખી શકો, જો કે આ જ સંખ્યાને તમે IIX તરીકે પણ લખી શકો. વિધેયમાંનું બીજું ચલ આ રીતે રજૂઆતના પ્રકાર માટે વપરાય છે. આ Mode ચલની કિંમત 0, 1, 2 અથવા 3 હોઈ શકે. TV અને ફિલ્મ ઉદ્યોગમાં ફિલ્મની પટ્ટી અને બીજી વસ્તુઓને સાંકેતિક ભાષામાં રાખવા માટે રોમન સંખ્યાનો ઉપયોગ થાય છે.

અરેબિક વિધેય રોમન વિધેય કરતાં ઊલટું છે, તે રોમન સંખ્યાને તેના ચલ તરીકે લઈને તેને અરેબિક સંખ્યામાં ફેરવે છે.

નીચે આપેલું ઉદાહરણ જુઓ :

=ROMAN (125) ની કિંમત **CXXV** પાછી મળે છે.

વિધેય **=ARABIC("MMXI")** શું કિંમત પાછી આપે છે તે શોધી કાઢો ? તે **2013** છે ?

આકૃતિ 7.29માં કેટલાંક ROMAN અને ARABIC વિધેયનાં ઉદાહરણો આપેલાં છે.

	A	B	C
1	Function	Value	
2	ROMAN(125)	CXXV	
3	ARABIC("mmxiii")	2013	
4	ARABIC(ROMAN(2013))	2013	
5			
6	Also consider following examples to see how mode works		
7	ROMAN(999)	CMXCIX	
8	ROMAN(999,0)	CMXCIX	
9	ROMAN(999,1)	LMVLIV	
10	ROMAN(999,2)	XMIX	
11	ROMAN(999,3)	VMIV	

આકૃતિ 7.29 : ROMAN અને ARABIC વિધેયનાં ઉદાહરણો

Trim વિધેય

આ વિધેય આપેલાં શાબ્દિક લખાણમાંથી (સ્ટ્રિંગમાંથી) વધારાની ખાલી જગ્યા (spaces) દૂર કરે છે. એટલે કે જો તમે **TRIM("I   Love   My   School!")** લખો તો તે તમને **I Love My School !** પાછું આપશે. આ વિધેયની વાક્યરચના **TRIM(text)** છે.

સ્ટ્રિંગની સરખામણી કરવી (Comparing Strings)

EXACT વિધેય આપેલી બે સ્ટ્રિંગની સરખામણી કરે છે અને જો તે આખી સ્ટ્રિંગ બિલકુલ સરખી હોય તો 1 કિંમત પરત કરે છે, અને જો સરખી ન હોય તો 0 કિંમત પરત કરે છે. ઉદાહરણ : **=EXACT("Blue", "Blue")** આપણને 1 કિંમત પરત કરે છે જ્યારે **=EXACT("Blu", "Blue")** આપણને 0 કિંમત પરત કરે છે.

આ વિધેયની વાક્યરચના **EXACT(s1, s2)** છે.

સ્ટ્રિંગના પેટાભાગ પાડવા (Substring)

કેલ્સીમાં ડાબી બાજુથી સ્ટ્રિંગનો પેટાભાગ કરવા માટે અને બીજુ જમણી બાજુથી સ્ટ્રિંગનો ભાગ કરવા માટે એમ બે વિધેયો ઉપલબ્ધ છે.

LEFT વિધેય શાબ્દિક લખાણમાંથી ડાબી બાજુનો પહેલો અક્ષર અથવા આપણે જણાવેલા n અક્ષરો પાછા આપે છે.

આ વિધેયની વાક્યરચના **LEFT(Text, n)** છે. આથી, આપેલી સ્ટ્રિંગની ડાબી બાજુથી 4 અક્ષરો જુદા પાડવા માટે આપણે **=LEFT("Ram Dhashrathbhai Patel", 4)** લખી શકીએ, જે **"Ram "** સ્ટ્રિંગ પાછી આપશે.

આ જ પ્રમાણે RIGHT વિધેય આપણને આપેલા શાબ્દિક લખાણની સ્ટ્રિંગમાંથી જમણી બાજુના છેડાથી n અક્ષરો છૂટા પાડી જવાબ પાછો આપે છે. એટલે કે જો આપણે **=RIGHT("Ram Dhashrathbhai Patel", 5)** લખીએ તો "Patel" જવાબ મળશે. આ વિધેયની વાક્યરચના **RIGHT(Text, n)** છે.

કેલ્સી સ્ટ્રિંગની વચ્ચેથી લખાણ જુદું કરવાની સગવડતા પણ પૂરી પાડે છે. આ વિધેયની વાક્યરચના **MID(Text, Start, n)** છે.

ઉપરનાં ઉદાહરણમાં આપેલાં નામમાંથી વચ્ચેનું નામ જુદું પાડવા માટે આપણે **=MID("Ram Dhashrathbhai Patel", 5, 13)** લખીએ તો **"Dhashrathbhai"** સ્ટ્રિંગ પાછી મળશે.

સ્ટ્રિંગની લંબાઈ(Length of a String) LEN વિધેય

LEN વિધેય આપેલી (શાબ્દિક લખાણ) સ્ટ્રિંગની લંબાઈ (કેટલા અક્ષરનો સમાવેશ છે તે) જણાવે છે. તેની વાક્યરચના **LEN(Text)** છે. આથી જો આપણે **LEN("Ram")** લખીએ તો આપણને **3** જવાબ મળશે. આકૃતિ 7.30માં સ્ટ્રિંગને લગતાં વિધેયોનાં ઉદાહરણો આપેલાં છે.

	A	B
1	Function	Value
2	TRIM("I Love My School")	I Love My School
3	EXACT("Blue", "Blue")	TRUE
4	EXACT("My School", "New School ")	FALSE
5	LEFT("Ram D Patel", 3)	Ram
6	RIGHT("Ram D Patel", 5)	Patel
7	MID("Ram D Patel", 5,1)	D
8	LEN("Ram D Patel")	11

આકૃતિ 7.30 : સ્ટ્રિંગને લગતાં વિધેયોનાં ઉદાહરણો

Time અને Date વિધેયો (Time and Date functions)

કેલ્સી આપણને સમય અને તારીખની ગણતરી માટેના વિધેય પૂરા પાડે છે. જેમાં નાનો સમયગાળો (જેમ કે કલાક, મિનિટ અથવા સેકન્ડમાં) હોય તો તમે time વિધેયનો ઉપયોગ કરી શકો અને મોટા સમયગાળા માટે (જેમ કે અઠવાડિયા, મહિનાઓ અથવા વર્ષો) તમે Date વિધેયનો ઉપયોગ કરી શકો.

આજની તારીખ જાણવા માટે (Today's date) TODAY વિધેય

TODAY વિધેય તમારા કમ્પ્યૂટરમાં સંગ્રહ કરેલી આજની તારીખ જણાવે છે. આ વિધેયમાં કોઈ ચલ નથી. ધારો કે આજની તારીખ 11મી જુલાઈ 2013 છે તો TODAY() વિધેય આપણને 11/07/2013 અથવા 11. Jul. 2013 (તમારા કમ્પ્યૂરમાં જે રીતે ફૉર્મેટ સેટ કરેલું હશે તે પ્રમાણે) જવાબ આપશે.

Date વિધેય

DATE વિધેય આપણે આપેલાં વર્ષ, તારીખ અને દિવસ પ્રમાણે માન્ય તારીખ આપે છે. એટલે કે જો તમે **=DATE(2013, 1, 1)** આપો તો તે વિધેય 1/1/13 કિંમત પાછી આપશે.

Day વિધેય

DAY વિધેય કોઈ પણ માન્ય સ્વરૂપમાં આપેલી તારીખને મહિનાના દિવસમાં ફેરવવા માટે વપરાય છે. ઉદાહરણ : જો આજે 11મી જુલાઈ 2013 હોય તો **=DAY(today())** આપણને 11 જવાબ આપશે.

ઉપર જણાવેલું વિધેય આપણે **=DAY(DATE(2013,7,11))** પણ લખી શકીએ. આ વિધેયની વાક્યરચના **DAY(Date)** છે.

DATE વિધેયમાં આપેલું ચલ તારીખ સ્વરૂપમાં પૂરું પાડવાની ખાતરી માટે આપણે અગાઉ વાપર્યું હતું તે રીતે DATE વિધેય વાપરી શકો. DATE વિધેય સૌપ્રથમ આપેલી સંખ્યાને માન્ય તારીખનાં સ્વરૂપમાં ફેરવે છે અને પછી તારીખમાંથી દિવસનો નંબર પાછો આપે છે.

WEEKDAY વિધેય

WEEKDAY વિધેય આપણે આપેલી સાચી તારીખને અઠવાડિયાના વાર (જેમ કે સોમવાર, મંગળવાર,....) માં ફેરવે છે. દિવસોને અનુક્રમ નંબર 0 (રવિવાર)થી 6 (શનિવાર) આપવામાં આવે છે. ઉદાહરણ : જો તમે **=WEEKDAY(DATE(2013,7,11))** લખો તો જવાબ 5 મળશે.

આ વિધેયની વાક્યરચના WEEKDAY(Date, n) છે. અહીં Date એ કોઈ માન્ય તારીખ છે અને n એ સંખ્યા છે જે અઠવાડિયું કયા દિવસથી શરૂ થાય છે તે જણાવે છે. તમે n માટે નીચેની કિંમતો વાપરી શકો છો.

N=1 : અઠવાડિયું રવિવારથી શરૂ થાય છે. (દિવસ 0).

N=2 : અઠવાડિયું સોમવારથી શરૂ થાય છે. (દિવસ 1).

જો તમે ચલ (argument) ન જણાવો તો તે કિંમત 1 લેશે, એટલે કે ઉપરના ઉદાહરણમાં જણાવ્યા પ્રમાણે દિવસોને અનુક્રમ નંબર 0 (રવિવાર)થી 6 (શનિવાર) ગણશે.

નીચે કેટલાંક ઉદાહરણો આપ્યાં છે :

=WEEKDAY(DATE(2013,7,11),0)નો જવાબ **3** મળશે.

=WEEKDAY(DATE(2013,7,11),1)નો જવાબ **5** (પૂર્વનિર્ધારિત) કે જે **=WEEKDAY(DATE(2013,7,11))** બરાબર થાય.

=WEEKDAY(DATE(2013,7,11),2)નો જવાબ **4** મળશે.

Month વિધેય

MONTH વિધેય કોઈ માન્ય તારીખને મહિનાના નંબરમાં ફેરવે છે. ઉદાહરણ : જો તમે =MONTH(DATE(2013,7,11)) કોઈ સેલમાં લખશો તો જવાબ 7 મળશે.

આ વિધેયની વાક્યરચના **MONTH(t)** છે.

YEAR વિધેય

YEAR વિધેય માન્ય તારીખને વર્ષમાં ફેરવે છે. ઉદાહરણ તરીકે, જો તમે **=YEAR(DATE(2013,7,11))** કોઈ સેલમાં લખશો તો જવાબ 2013 મળશે.

આ વિધેયની વાક્યરચના **YEAR(t)** છે.

વર્ષમાં દિવસોની સંખ્યા (Number of Days in a Year)

DAYSINYEAR વિધેય ચલની તારીખ આપેલ હોય તે વર્ષના કુલ દિવસોની સંખ્યા આપે છે. સ્પ્રેડશીટના સેલમાં નીચેનું વિધેય લખો :

=DAYSINYEAR(DATE(2013,7,11)).

આ વિધેય 365 જવાબ આપશે.

આ વિધેયની વાક્યરચના **DAYSINYEAR(t)** છે.

આપેલું વર્ષ લીપ યર છે કે કેમ તે જાણવા માટે કદાચ તમે અગાઉ કોઈ વિશિષ્ટ પ્રોગ્રામ કે ગૂંચવાડાભર્યું સૂત્ર લખ્યું હશે. એના બદલે તમે વર્ષમાં કેટલા દિવસ છે તે ચકાસીને જાણી શકો છો.

બે તારીખનો તફાવત (Difference between two dates) – DAYS વિધેય

DAYS વિધેય આપેલી બે તારીખ વચ્ચેના દિવસોની ગણતરી કરે છે. અહીં એ નોંધ કરો કે વિધેયનું નામ DAYને બદલે DAYS છે. અગાઉ ચર્ચા કરેલું DAY વિધેય મહિનાનો કેટલામો દિવસ છે તે જણાવે છે. DAYS વિધેય આપણે આપેલી બે માન્ય તારીખો વચ્ચેના દિવસોની સંખ્યા આપે છે.

જો આપણે **=DAYS(DATE(2012,7,11), DATE(2013,7,11))** લખીએ તો 366 જવાબ મળશે. 2012 વર્ષ લીપ યર હોવાથી તેમાં 366 દિવસો છે !

હવે આ વિધેય નીચે દર્શાવ્યા પ્રમાણે વાપરો :

=DAYS(A1, NOW())

અહીં A1 એક સેલ સ્થાનાંક છે જેમાં તમારે તમારી જન્મ તારીખ લખવાની છે. બીજો ચલ NOW() છે, જે કમ્પ્યૂટરમાં સંગ્રહ કરેલી આજની તારીખ સમય સાથે પાછી આપશે. હવે વિચારો કે ઉપર જણાવેલું વિધેય શું જવાબ આપશે ? બરાબર છે, તમે સાચા છો, તે તમારી ઉંમર દિવસોમાં જણાવશે. આ રીતે નિશ્ચિતપણે તમે શોધી શકશો કે તમારા વર્ગમાં સૌથી મોટું કોણ છે ?

NOW વિધેયની વાક્યરચના **NOW()** જ છે. આ વિધેયમાં કોઈ ચલ નથી.

અહીં નોંધ કરો કે બીજી તારીખ પહેલી તારીખમાંથી બાદ કરવામાં આવે છે, આથી ઘણા સમયે વિધેય ઋણ કિંમત પાછી આપશે. આ ટાળવા માટે તમે ABS વિધેયનો ઉપયોગ કરી શકો. આકૃતિ 7.31માં તારીખ અને સમયને લગતા કેટલાક વિધેયોનાં ઉદાહરણો આપેલાં છે.

	A	B
1	Function	Value
2	TODAY()	11. Jul. 2013
3	DATE(2013,7,11)	July 11, 2013
4	DAY(DATE(2013,7,11))	11
5	WEEKDAY(DATE(2013,7,11))	5
6	WEEKDAY(DATE(2013,7,11),0)	3
7	WEEKDAY(DATE(2013,7,11),1)	5
8	WEEKDAY(DATE(2013,7,11),2)	4
9	MONTH(DATE(2013,7,11))	7
10	YEAR(DATE(2013,7,11))	2013
11	DAYSINYEAR(DATE(2013,7,11))	365
12	DAYS(DATE(2012,7,11), DATE(2011,7,11))	366
13	WEEKS(DATE(2011,7,11), DATE(2012,7,11),0)	52
14	YEARS(DATE(2011,7,11), DATE(2013,7,11),0)	2

આકૃતિ 7.31 : તારીખ અને સમયને લગતાં વિધેયો

આપેલી બે તારીખો વચ્ચે અઠવાડિયાની સંખ્યા (Number of weeks between two dates)

આપણે જાણીએ છીએ કે DAY અને DAYS નામનાં બે વિધેયો છે; એ જ રીતે WEEKS બે તારીખો વચ્ચે અઠવાડિયાઓની સંખ્યાની ગણતરી કરે છે.

=WEEKS(DATE(2011,7,11), DATE(2012,7,11),0)નો જવાબ 52 મળશે.

ઉપર જણાવેલાં વિધેયમાં બે DATE વિધેયોનો ઉપયોગ કરેલો છે જે આપેલા ચલને માન્ય તારીખના સ્વરૂપમાં ફેરવે છે. આ ઉપરાંત, WEEKS વિધેય એક વધારે ચલ '0' નો પણ ઉપયોગ કરે છે, કે જે અઠવાડિયાના પ્રકારનો નિર્દેશ કરે છે. આ ચલની '0' કિંમત અઠવાડિયાનો સમયગાળો જણાવે છે. અહીં તમે '0' ને બદલે '1' જણાવીને કેલેન્ડર અઠવાડિયાનો ઉપયોગ કરી શકો છો.

WEEK વિધેયની વાક્યરચના **WEEKS(t1, t2, Week type)** છે. જેવી રીતે DAY અને DAYS બે અલગ અલગ વિધેયો છે એ જ રીતે WEEK અને WEEKS બે અલગ અલગ વિધેયો છે.

આપેલી બે તારીખો વચ્ચે વર્ષની સંખ્યા (Number of years between two dates)

YEARS વિધેય આપેલી બે માન્ય તારીખો વચ્ચે વર્ષની સંખ્યાની ગણતરી કરે છે કે જે વર્ષના સમયગાળામાં (ચલની કિંમત 0 જણાવીને) અથવા કેલેન્ડર વર્ષમાં (ચલની કિંમત 1 જણાવીને) હોય છે. ઉદાહરણ તરીકે, જો તમે **=YEARS(DATE(2011,7,11), DATE(2013,7,11), 0)** લખશો તો તમે **2** જવાબ મેળવશો.

આ વિધેય વડે તમે કેટલાં વર્ષનાં થયાં તે જાણી શકશો. તમારી જન્મ તારીખથી આજની તારીખ વચ્ચે કેટલાં વર્ષ છે તે ફક્ત શોધી કાઢો.

અન્ય સ્પ્રેડશીટ પૅકેજમાં ઉપલબ્ધ વિધેયો (Functions in other Spreadsheet Packages)

ઘણા ખરા સ્પ્રેડશીટ પેકેજ આ જ પ્રકારનાં વિધેય પૂરા પાડે છે. જો તમે માઇક્રોસૉફ્ટ એક્સલનું ઉદાહરણ લેશો તો તમને જણાશે કે તે પણ આ પ્રકરણમાં ચર્ચા કરી તેમાંના મોટાભાગના વિધેયો પૂરા પાડે છે.

ક્યારેક તમારે તેની વાક્યરચના (syntax)માં ગૌણ પરિવર્તન કરવા પડે છે જેમ કે તેના સ્પેલિંગમાં કોઈ નાનો ફેરફાર અથવા વિધેયના ચલ વચ્ચે ';' (અર્ધવિરામ) કે ',' (અલ્પવિરામ)નો ઉપયોગ.

ગૂગલ સ્પ્રેડશીટ પૅકેજ [doc.google.com] પણ આ જ પ્રકારનાં વિધેયો પૂરા પાડે છે. તમે જાણો છો કે તે ઇન્ટરનેટ, મોબાઇલ ફોન અને એન્ડ્રોઇડ જેવી ચાલક પદ્ધતિ વાપરતાં યંત્રોમાં ખૂબ ઉપયોગી થાય છે.

આ ઉપરાંત ગૂગલ પૅકેજ તમને નિ:શુલ્ક સ્પ્રેડશીટ ટેમ્પલેટ પણ અનેક કાર્યો કરવા - જેમ કે વિદ્યાર્થીઓનું સમયપત્રક, પ્રગતિપત્રકનો અહેવાલ, પ્રોજેક્ટનું આયોજન, વિદ્યાર્થીઓ અને કર્મચારીઓની હાજરી, અંગત આવક-જાવકનું નિયમન, લગ્નપ્રસંગ માટે મહેમાનોની યાદી, માલ-વેચાણનું ભરતિયું બનાવવું વગેરે માટે પૂરાં પાડે છે.

આ ટેમ્પલેટમાં તૈયાર માળખાં જેમ કે વિભાગો, ઊભી હરોળ અને આડી હરોળનાં શીર્ષકો, જરૂરી ગણતરીઓ કરવા માટેનાં સૂત્રો તેમજ સ્ક્રિપ્ટ હોય છે. ગૂગલ સ્પ્રેડશીટ જેવાં વિનિયોગો પ્લૅટફૉર્મથી સ્વતંત્ર અને અન્ય જગ્યાએ ફેરવવા સરળ હોવાથી ખૂબ જ પ્રચલિત બની રહ્યા છે.

સારાંશ

આપણે આ પ્રકરણમાં વિવિધ ગાણિતિક, આંકડાકીય, સ્ટ્રિંગ, તારીખ અને સમયને લગતાં તેમજ ફૉર્મેટિંગ માટેનાં વિધેયોનો અભ્યાસ કર્યો. આપણે ફંક્શન વિઝાર્ડ મારફતે કઈ રીતે વિધેય દાખલ કરી શકાય તે પણ જાણ્યું. આ ઉપરાંત, વિધેયો સાથે કાર્ય કરતા સમયે ભૂલો તરફ નિર્દેશ કરવામાં અને તેને સુધારવામાં ફંક્શન વિઝાર્ડ મદદરૂપ થાય છે તે પણ શીખ્યા.

સ્વાધ્યાય

1. ફંક્શન વિઝાર્ડનું કાર્ય ટૂંકમાં સમજાવો.
2. તમારી પસંદગીના કોઈ પણ ત્રણ ગાણિતિક વિધેય યોગ્ય ઉદાહરણ સાથે સમજાવો.
3. તમારી પસંદગીના કોઈ પણ ત્રણ આંકડાકીય વિધેય યોગ્ય ઉદાહરણ સાથે સમજાવો.
4. તમારી પસંદગીના નિર્ણય લેવામાં ઉપયોગી કોઈ પણ ત્રણ વિધેય યોગ્ય ઉદાહરણ સાથે સમજાવો.
5. તમારી પસંદગીના કોઈ પણ ત્રણ તારીખ અને સમયને લગતા વિધેય યોગ્ય ઉદાહરણ સાથે સમજાવો.
6. તમારી પસંદગીના કોઈ પણ ત્રણ શાબ્દિક લખાણ ઉપરના વિધેય (સ્ટ્રિંગ ફંક્શન) યોગ્ય ઉદાહરણ સાથે સમજાવો.
7. કેલ્સીમાં ઉપલબ્ધ IF વિધેય યોગ્ય ઉદાહરણ સાથે સમજાવો.
8. ત્રણ તાર્કિક વિધેય યોગ્ય ઉદાહરણ સાથે સમજાવો.
9. નીચેનામાંથી સાચો વિકલ્પ જણાવો :

(1) નીચેનામાંથી કઈ રીતે આપણે કેલ્સીમાં વિધેય દાખલ કરી શકીએ ?

(a) સેલમાં વિધેયનું નામ સીધેસીધું ટાઈપ કરીને

(b) ફંક્શન વિઝાર્ડ અથવા ટૂલબાર ઉપરથી પસંદ કરીને

(c) (a) અને (b) બંને

(d) વિધેય આધારિત

(2) વિધેયની શરૂઆત નીચેનામાંથી કયા વિકલ્પ સાથે થાય છે ?

(a) '=' ચિહ્ન (b) મૂળાક્ષરો (c) સંખ્યા (d) આમાંનો કોઈ પણ વિકલ્પ

(3) નીચેનામાંથી ક્યો તાર્કિક વિધેય નથી ?

(a) OR (b) AND (c) NOT (d) PROPER

(4) આપેલું વર્ષ લીપ યર છે કે નહીં તે શોધવા માટે તમે કેલ્સીના સમય અને તારીખને લગતાં વિધેયોમાંથી ક્યો વિધેય વાપરશો ?

(a) DATE (b) TIMESTAMP (c) YEARS (d) YEARDIFF

(5) કેલ્સીના સેલમાં વિધેય દાખલ કરવા માટે નીચેનામાંથી કઈ યુક્તિ શક્ય છે ?

(a) ફંક્શન વિઝાર્ડ દ્વારા (b) જાતે વિધેય દાખલ કરીને

(c) (a) અને (b) બંને (d) વિધેય આધારિત

(6) કેલ્સીનાં વિધેયમાં કેટલા ચલ વાપરી શકાય ?

(a) એક (b) બે (c) ત્રણ (d) વિધેય આધારિત

(7) વિધેયનો ચલ શું હોઈ શકે ?

(a) કિંમત (b) શાબ્દિક લખાણ (c) અન્ય વિધેય (d) આપેલા બધા

પ્રાયોગિક સ્વાધ્યાય

1. તમારા જુદા જુદા વિષયોના ગુણ લઈ એક સાદું ગુણપત્રક બનાવો અને તેમાં શાળાનું નામ, વિદ્યાર્થીનો નંબર, વિદ્યાર્થીનું નામ, ધોરણ અને ગુણનો સમાવેશ કરો. આપેલા ગુણને આધારે ટકા અને વર્ગ (પ્રથમ વર્ગ, પાસ, નાપાસ વગેરે) પણ શોધો. આ કામ માટે તમે તમારું ગયા વર્ષનું નિશાળનું રિપોર્ટ કાર્ડ પણ વાપરી શકો.
2. કેલ્સી વાપરીને તમારા ઘરનું વીજળીનું બિલ બનાવો.
3. તમારા ઘરનાં સદસ્યને કે પડોશીને લોન બાબતની માહિતી પૂછો. NPER વિધેય વાપરીને લોન પરત કરવાનો સમયગાળો શોધો.
4. ત્રણ જુદા જુદા સેલમાં ત્રણ સંખ્યાઓ લખો. આ કિંમતો એવી લખો કે જે એક માન્ય તારીખ બની રહે. હવે DATE વિધેય વાપરીને આ કિંમતોને માન્ય તારીખમાં ફેરવો.
5. કોઈ તારીખ લઈ શોધી કાઢો કે તે લીપ યર છે કે નહીં ?
6. મોબાઈલ, આઈ-ફોન અને આઈ-પેડ માટેનાં સ્પ્રેડશીટ પૅકેજ વિશે માહિતી મેળવવા પ્રયત્ન કરો. આ જાતની સગવડ ધરાવતાં પૅકેજ / સ્પ્રેડશીટનાં નામ શોધી કાઢો.

કેલ્સીમાં આલેખની રચના

ટેક્નોલૉજીના વિકાસ અને તક્નિકી વિશ્લેષણની વધતી જરૂરિયાતોને કારણે આલેખ(ચાર્ટ-chart)નો ઉપયોગ ઘણો વધ્યો છે. સ્પ્રેડશીટ પૅકેજ વ્યાપકપણે પ્રચલિત થવાનું એક કારણ આલેખની ઉપયોગિતા છે. આલેખ એ વિશાળ માહિતીને આકર્ષક રીતે રજૂ કરવાની ક્ષમતા ધરાવતી વ્યાવસાયિક સ્થિતિનું એક ઉદાહરણ છે. આલેખને આંકડાકીય માહિતી ચિત્રાત્મક સ્વરૂપે રજૂ કરવાની રીત પણ કહેવામાં આવે છે.

આલેખ તૈયાર કરવા માટે કોઈ પણ વ્યક્તિને વિવિધ આલેખોનું મૂળભૂત જ્ઞાન અને આલેખનાં સ્વરૂપો તેમજ તેના વિનિયોગોના ખ્યાલ વિશે જાણકારી હોવી જોઈએ. એટલે કે, સ્પ્રેડશીટ પૅકેજમાં માત્ર કોઈ પણ પ્રકારનો આલેખ પસંદ કરીને આલેખ તૈયાર કરી દેવાથી વ્યવસાયમાં તેનું ખાસ પ્રદાન રહેતું નથી. તમને આલેખનાં ખ્યાલોની સમજણ હોવી જ જોઈએ કે આલેખ તમારા વ્યવસાયમાં કઈ રીતે મદદરૂપ થશે. આ ઉપરાંત દરેક વ્યક્તિએ એ પણ જાણવું જોઈએ કે કોઈ ચોક્કસ પ્રકારનો આલેખ (ચાર્ટ) ક્યારે વાપરવો જોઈએ, તે ક્યા પ્રકારની માહિતી પૂરી પાડે છે, તે ક્યા પ્રકારનો આધાર પૂરો પાડે છે અને આ આલેખથી ક્યા પ્રકારના નિર્ણય લઈ શકાય છે.

આલેખના પ્રકારો (Type of charts)

વિનિયોગની પ્રકૃતિ અને આપણી જરૂરિયાત પ્રમાણે આલેખનો પ્રકાર પસંદ કરવો જોઈએ. આલેખ વિવિધ કાર્યો માટે વપરાય છે જેમ કે : ઇતિહાસ કહેવા માટે, વિકલ્પોના મૂલ્યાંકન માટે, કોઈ વલણ રજૂ કરવા માટે અને અસાધારણ કિસ્સાઓ શોધવા માટે. આનો અર્થ એ થાય કે ખોટા પ્રકારના આલેખની પસંદગી ખ્યાલોની અપૂરતી રજૂઆત કરે છે અને ક્યારેક ગેરસમજ ઉત્પન્ન કરે છે. જ્યારે બીજી બાજુએ, યોગ્ય પ્રકારના આલેખની પસંદગી સાચા અને ઝડપી નિર્ણયો લેવામાં મદદરૂપ થાય છે. આપણે શા માટે આલેખ વાપરવા જોઈએ તેનાં પ્રચલતિ કારણો નીચે મુજબ છેઃ

- સરખામણી માટે
- વિતરણની સમજૂતી આપવા માટે
- સ્થિતિ સમજવા માટે
- અમુક ચોક્કસ સમયગાળાના વલણનું વિશ્લેષણ કરવા માટે
- નિયત પરિણામમાં ફેરફારની બારીક તપાસ કરવા માટે
- વાસ્તવિક વસ્તુઓ (એન્ટિટી) વચ્ચેના સંબંધો ઓળખવા અને સમજવા માટે

કોઈપણ પરિસ્થિતિમાં આલેખ તૈયાર કરવા માટે નીચે આપેલાં પગલાંને અનુસરો :

- તમે જે કંઈ કહેવા ઇચ્છતા હોય તે બાબત સ્પષ્ટ બનો.
- માહિતી (ડેટા) ભેગી કરો અને ગોઠવો.
- અમાન્ય ડેટા કાઢી નાખો.
- સૌથી વધુ યોગ્ય આલેખનો પ્રકાર નક્કી કરો અને ખાતરી કરો કે આલેખ માટે એકઠો કરેલો ડેટા પૂરતો છે કે કેમ.
- આલેખ (ચાર્ટ) બનાવો અને
- ચાર્ટને વધારે આકર્ષક અને યોગ્ય બનાવો (ફૉર્મેટ કરો).

ઉપર જણાવ્યા પ્રમાણે ધ્યેય નક્કી કર્યા પછી આલેખ બનાવવા માટે સૌ પ્રથમ પગલું ડેટા તૈયાર કરવાનું છે. એક વાર સ્પ્રેડશીટના દસ્તાવેજમાં ડેટા દાખલ કરી દેવામાં આવે પછી તમે તેને ચિત્રાત્મક (ગ્રાફિકલ) સ્વરૂપમાં જોઈ શકો છો.

આલેખ (ચાર્ટ) ઉમેરવો (Inserting a chart)

વર્કશીટમાં આલેખ ઉમેરવા માટે નીચે પ્રમાણે કાર્ય કરો :

- સ્પ્રેડશીટમાં ડેટાનો વિસ્તાર પસંદ કરો.
- હવે પસંદ કરો : **Insert → Chart**

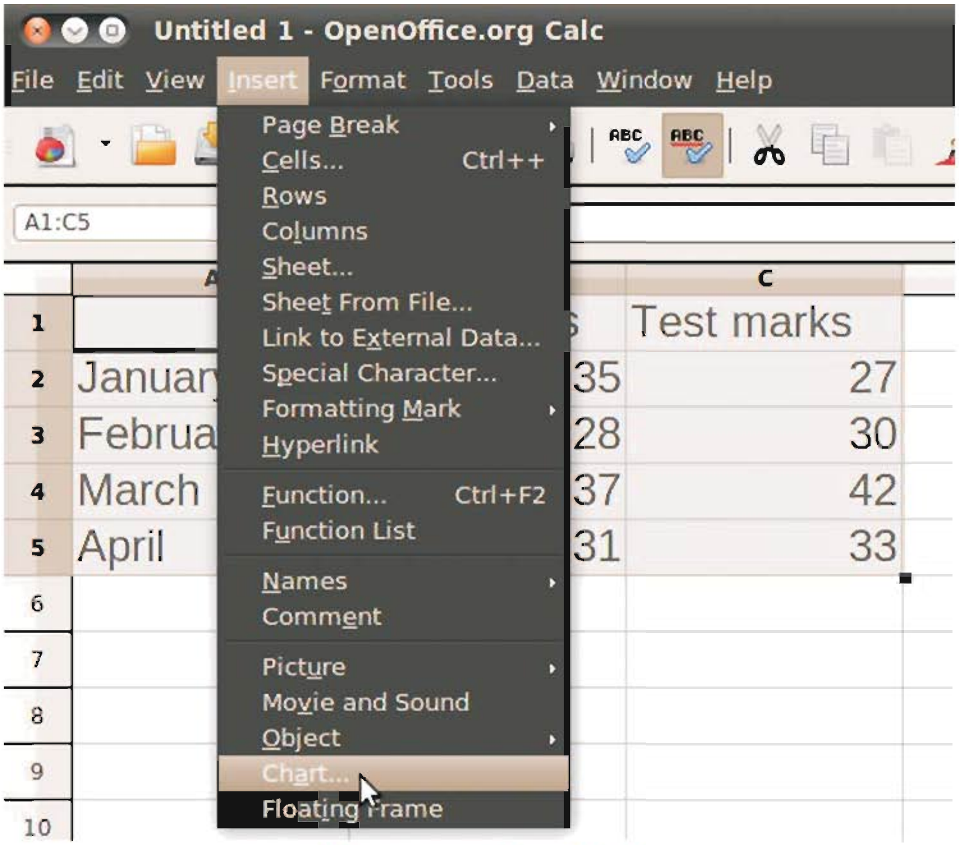
ધારો કે તમને શૈક્ષણિક વર્ષમાં જાન્યુઆરીથી એપ્રિલ દરમિયાન લીધેલી અણધારી કસોટી (surprise quizzes) અને વર્ગ પરીક્ષાઓના ગુણ કોષ્ટક 8.1માં જણાવ્યા પ્રમાણે આપ્યા છે.

	Quiz marks	Test marks
January	35	27
February	28	30
March	37	42
April	31	33

કોષ્ટક 8.1 : વિદ્યાર્થીના ગુણ

સ્પ્રેડશીટ દસ્તાવેજ ખોલો અને કોષ્ટક 8.1માં દર્શાવ્યા પ્રમાણે ડેટા દાખલ કરો. ડેટાનો ક્રમ બદલશો નહીં અન્યથા જ્યારે તમે આલેખ બનાવશો ત્યારે તે અલગ પરિણામ આપશે. ડેટાનો સંગ્રહ પણ વારંવાર કરો.

એકવાર તમે સંપૂર્ણ ડેટા દાખલ કરી દો પછી ડેટાનો વિસ્તાર પસંદ કરો. અહીં આપણા કિસ્સામાં ડેટાનો વિસ્તાર સેલ A1 થી સેલ C5 છે. ડેટાનો વિસ્તાર પસંદ કરેલો રાખીને જ હવે **Insert** વિકલ્પ આપો. એક ઊભું સબમેનૂ સ્ક્રીન ઉપર દેખાશે. તેમાંથી **Chart** વિકલ્પ પસંદ કરો. કાર્યની શ્રેણી આકૃતિ 8.1માં દર્શાવી છે.



આકૃતિ 8.1 : આલેખ ઉમેરવો