

HTML، CSS اور جاوا اسکریپٹ کے ساتھ ویب ڈویلپمنٹ

اس باب کا مطالعہ کرنے کے بعد طلبہ اس قابل ہو جائیں گے وہ:

- جاوا اسکریپٹ کا سینٹیکس اور ڈیٹا کی اقسام کو سمجھ سکیں۔
- جاوا اسکریپٹ میں ویری ایبل، آپریٹرز اور فنکشن کے ساتھ کام کر سکیں۔
- جاوا اسکریپٹ کے ساتھ ایونٹس اور صارف کے ان پٹ کو سنبھال سکیں۔
- جاوا اسکریپٹ کا استعمال کرتے ہوئے سادہ پروگرام بناسکیں۔
- HTML فارم بنائیں اور انھیں اسٹائل کر سکیں۔
- آپریٹرز، ویری ایبل اور فنکشن کے ساتھ ایونٹس کو سنبھالنے کے لیے جاوا اسکریپٹ کا استعمال کر سکیں۔
- جامد ویب صفحات تیار کریں۔
- ویب صفحات بنانے کے لیے HTML ٹیگ مناسب طریقے سے لاگو کر سکیں۔
- ایک بنیادی HTML صفحہ بناسکیں۔
- کسی صفحے پر متن، تصاویر اور لنکس شامل کر سکیں۔
- فہرستیں اور جدول بناسکیں۔
- HTML عناصر پر سٹائل لاگو کر سکیں۔
- فونٹ، رنگ، اور پس منظر کے ساتھ کام کریں۔
- پیراگراف اور فہرستوں کی شکل میں ڈیٹا ظاہر کرنے کے لیے ویب صفحات بناسکیں۔
- CSS سینٹیکس سے واقف ہو سکیں۔
- CSS کے ساتھ لے آؤٹ بناسکیں۔
- CSS کے ساتھ اینی میشن اور ٹرانزیکشن شامل کر سکیں۔
- سٹیلک ویب پیجز کو تیار، ٹیسٹ اور ڈی بگ کر سکیں۔
- تصاویر اور متن کو مؤثر طریقے سے منظم کر سکیں۔
- آپریٹرز، ویری ایبلز اور فنکشنز کا استعمال کرتے ہوئے ایونٹس کو سنبھالنے کے لیے HTML کے ساتھ جاوا اسکریپٹ کا استعمال کر سکیں۔

تعارف

اس باب میں، ہم بنیادی ویب ڈویلپمنٹ جاوا اسکرپٹ، ہائپر ٹیکسٹ مارک اپ لینگویج (HTML) اور کیسکیڈنگ اسٹائل شیٹس (CSS) کے بنیادی اصولوں کو دیکھیں گے۔ اس باب کے اختتام تک، آپ جاوا اسکرپٹ سینٹیکس اور ڈیٹا کی اقسام کو سمجھنے، ویری ایبلز، آپریٹرز اور فنکشنز کے ساتھ کام کرنے، ایونٹس اور صارف کے ان پٹ کو سنبھالنے، سادہ پروگرام بنانے، HTML اور CSS کا استعمال کرتے ہوئے سٹینک ویب صفحات تیار کرنے کے قابل ہوں گے۔

8.1 ویب ڈویلپمنٹ

ویب ڈویلپمنٹ ویب سائٹس اور ویب ایپلی کیشنز بنانے کا عمل ہے۔ اس میں ویب سائٹس کو ڈیزائن، تعمیر اور اس کی دیکھ بھال کرنے کے لیے مختلف پروگرامنگ کی زبانوں اور آلات کا استعمال شامل ہے۔

8.1.1 ویب ڈویلپمنٹ کیوں سیکھیں؟

ویب ڈویلپمنٹ کئی وجوہات کی بنا پر ایک قابل قدر مہارت ہے:

- (i) ڈیجیٹل خواندگی: جب آپ ویب ڈویلپمنٹ سیکھتے ہیں تو، آپ کو پتہ چلتا ہے کہ ویب سائٹس کیسے بنائی جاتی ہیں۔ آپ HTML کے بارے میں سیکھتے ہیں، جو ایک ویب پیج کے ڈھانچے کی طرح ہے، CSS، جو ویب پیج کو اچھا بناتا ہے، اور جاوا اسکرپٹ، جو ویب پیج کو پُرکشش بناتا ہے۔ اس سے آپ کو یہ سمجھنے میں مدد ملتی ہے کہ انٹرنیٹ کیسے کام کرتا ہے۔
- (ii) کیریئر کے مواقع: بڑھتی ہوئی آئی ٹی صنعت میں ملازمت کے امکانات کی ایک وسیع رینج کھولتا ہے۔ ویب ڈویلپرز بہت سے مختلف قسم کی ملازمتیں حاصل کر سکتے ہیں۔ آپ ایک ویب ڈویلپر، ویب ڈیزائنر اور بہت کچھ بن سکتے ہیں۔ بہت سی کمپنیوں کو اپنی ویب سائٹس بنانے اور ان کو برقرار رکھنے کے لیے ویب ڈویلپرز کی ضرورت ہوتی ہے۔ اس کا مطلب ہے کہ آپ کو بہت ساری جگہوں پر اچھی نوکریاں مل سکتی ہیں۔

- (iii) مسئلہ حل کرنا: جب آپ ایک ویب سائٹ بناتے ہیں تو آپ بہت سے مسائل حل کرتے ہیں۔ مثال کے طور پر، اگر کوئی ویب سائٹ سست ہے تو، آپ یہ معلوم کرتے ہیں کہ یہ سست کیوں ہے پھر اسے ٹھیک کرتے ہیں۔ اس سے آپ کو منطقی طور پر سوچنے اور مسائل کو بہتر طریقے سے حل کرنے میں مدد ملتی ہے۔

- (iv) تخلیقی صلاحیت: یہ آپ کو بصری طور پر پُرکشش اور انٹرایکٹو ویب سائٹس بنانے کی اجازت دیتا ہے۔ ویب ڈویلپمنٹ آپ کو تخلیقی بناتی ہے۔ آپ بہترین ترتیب، رنگ، اور پُرکشش خصوصیات کے ساتھ ویب سائٹس ڈیزائن کر سکتے ہیں۔ مثال کے طور پر آپ اپنے فن پارے کو دکھانے کے لیے ایک ذاتی بلاگ یا پورٹ فولیو بنا سکتے ہیں اور اپنی منفرد ویب سائٹ بنا سکتے ہیں۔

- (v) انٹرپرائیو ریشپ: ویب ڈویلپمنٹ کی مہارت کے ساتھ، آپ اپنا آن لائن کاروبار شروع کر سکتے ہیں۔ مثال کے طور پر اگر آپ دستکاری بناتے ہیں تو، آپ انھیں فروخت کرنے کے لیے ایک ویب سائٹ بنا سکتے ہیں۔ یا آپ ایک تفریحی ایپ کی طرح ایک نئی ویب سروس تشکیل دے سکتے ہیں اور اسے دنیا کے ساتھ اشتراک کر سکتے ہیں۔

کیا آپ جانتے ہیں؟

ہاٹ میل پہلی ویب پر مبنی ای میل سروس ہے جسے دو طلبہ ”سپر بھائی اور جیک اسمتھ“ نے 1996ء میں بنایا۔ اس وقت وہ اسٹیفن روڈ یونیورسٹی میں تھے۔ بعد میں اسے مائیکروسافٹ نے 400 ملین ڈالر میں خرید لیا۔

8.2 ویب ڈویلپمنٹ کے بنیادی اجزاء

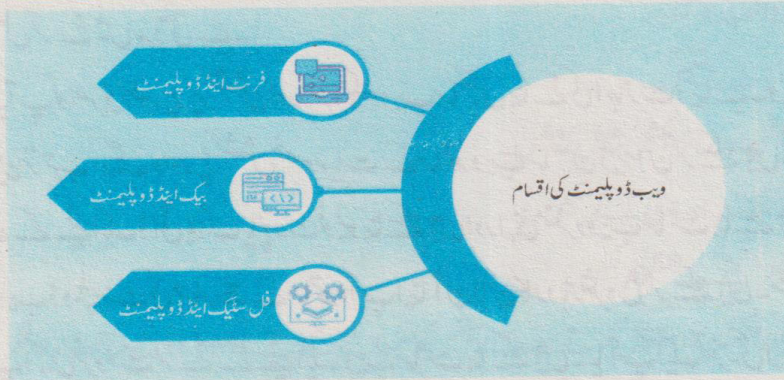
ویب ڈویلپمنٹ میں ویب سائٹس اور ویب ایپلی کیشنز بنانا شامل ہے۔ اس کے تین اہم اجزاء ہیں:

- (i) فرنٹ اینڈ ڈویلپمنٹ: یہ اس بات پر توجہ مرکوز کرتا ہے کہ صارفین کسی ویب سائٹ پر کیا دیکھتے ہیں اور اس کے ساتھ بات چیت کرتے ہیں۔ مندرجہ ذیل، انٹرایکٹو فرنٹ اینڈز کو ڈیزائن کرنے کے لیے بنیادی باتیں استعمال کی جاتی ہیں:
 - HTML ویب صفحات پر مواد کو تشکیل دیتا ہے، جیسے عنوانات، پیراگراف، تصاویر، اور لنکس۔
 - CSS ویب صفحات پر مواد کو اسٹائل کرتا ہے، رنگ، فونٹ اور ترتیب کو تبدیل کرتا ہے تاکہ وہ خوبصورت نظر آئیں۔
 - جاوا اسکریپٹ ویب صفحات میں انٹرایکٹیویٹی شامل کرتا ہے، جس سے وہ ڈائنامک بنتی ہے اور ان کو استعمال کرنا اچھا لگتا ہے۔ یہ فارم، اینی میشن اور کھیل جیسی خصوصیات کی اجازت دیتا ہے۔
- (ii) بیک اینڈ ڈویلپمنٹ: یہ ویب سائٹ کے پس پردہ حصے کا انتظام کرتا ہے، جیسے سرورز، ڈیٹابیس اور ایپلی کیشن منطق وغیرہ۔ اہم بیک اینڈ ٹیکنالوجیز یہ ہیں:

- ویب سرورز: وہ کمپیوٹرز ہیں جو ویب پیجز کو اسٹور کرتے ہیں اور جب صارف یو۔ آر۔ ایل لکھتے ہیں تو ان کو ویب پیج فراہم کرتے ہیں۔
- ڈیٹابیس ڈیٹا کو اسٹور اور منظم کرتے ہیں، جیسے صارف کی معلومات، مصنوعات کی تفصیلات، اور ویب سائٹ کا مواد وغیرہ۔
- بیک اینڈ پروگرامنگ لینگویجز، جیسے پی ایچ پی، پائٹھن، اور روبی جو فارم پروسیسنگ، اور صارف لاگ ان کا انتظام جیسے کاموں کو سنبھالتے ہیں۔

کیا آپ جانتے ہیں؟

پہلی ویب سائٹ ٹم برنرز لی نے 1991 میں بنائی تھی اور یہ اب بھی <http://info.cern.ch> پر قابل رسائی ہے۔ یہ ایک سادہ صفحہ تھا۔ ورلڈ وائڈ ویب پروجیکٹ کے بارے میں معلومات کے لنکس کے ساتھ۔

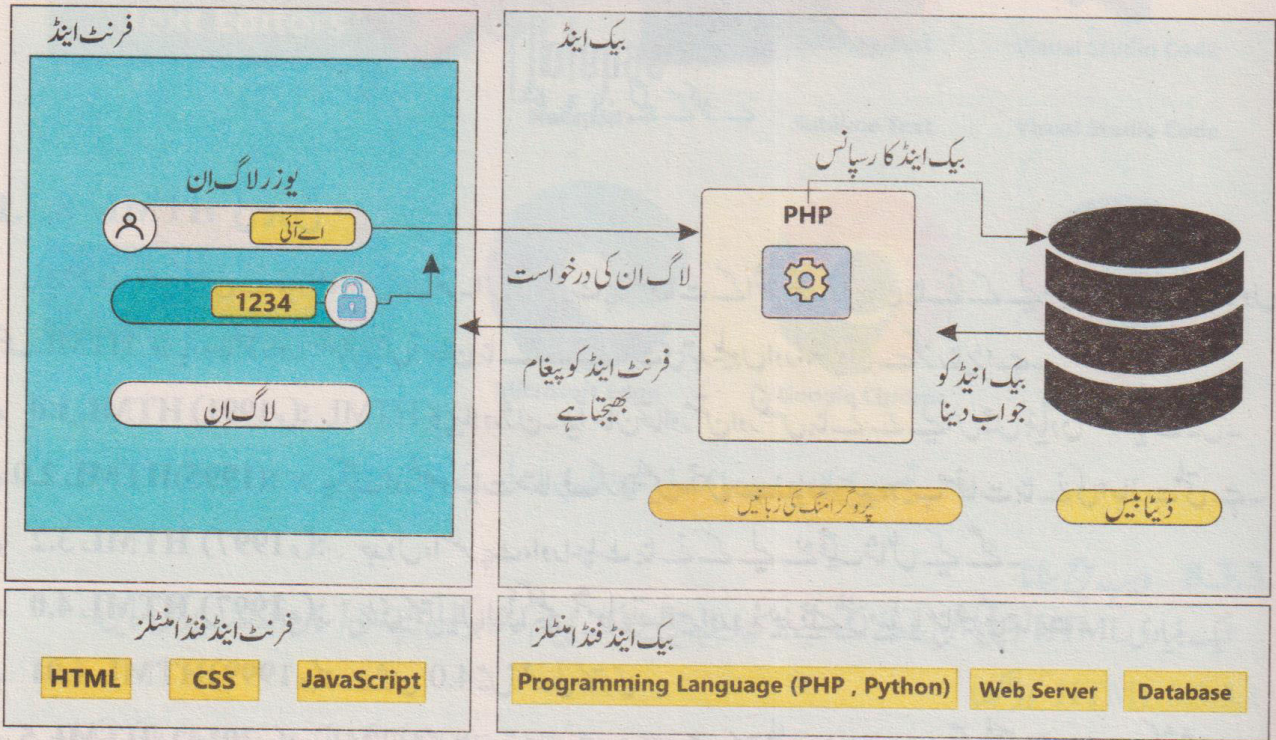


شکل 8.1: ویب ڈویلپمنٹ کی اقسام

مثال: لاگ ان سسٹم

لاگ ان سسٹم ویب ڈویلپمنٹ میں ایک عام خصوصیت ہے، جس سے صارفین کو ویب سائٹ پر اپنے اکاؤنٹس تک رسائی حاصل کرنے کی اجازت ملتی ہے۔ یہ مثال فرنٹ اینڈ اور بیک اینڈ ڈویلپمنٹ کے کردار کے ساتھ ساتھ فل اسٹیک ڈویلپمنٹ کے تصور کی وضاحت کرنے میں مدد کرے گی۔

(iii) فل اسٹیک ڈویلپمنٹ: ہمارے لاگ ان سسٹم کی صورت میں، ایک فل اسٹیک ڈویلپر یو آئی (Front End) بنائے گا اور صارف کی توثیق اور ڈیٹا بیس تعامل (Back End) کو سنبھالے گا۔



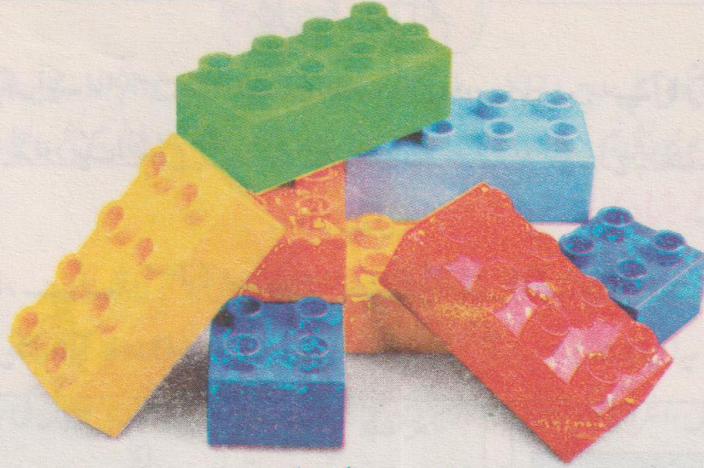
شکل 8.2: لاگ ان سسٹم کا تصویری خلاصہ

کیا آپ جانتے ہیں؟

فل اسٹیک ڈویلپر کی زیادہ اہمیت ہے کیونکہ وہ کر سکتے ہیں۔ ویب ایپلیکیشن کے تمام پہلوؤں کو منظم اور تیار کرنا، بنانا وہ ٹیک انڈسٹری میں ورٹائل اور قیمتی ہیں۔

8.3 HTML کے ساتھ شروع کرنا

HTML ویب صفحات بنانے کے لیے استعمال ہونے والی معیاری زبان ہے۔ HTML کو ایک ویب سائٹ کے بلڈنگ بلاکس کے طور پر سوچیں۔ بالکل اسی طرح جیسے لیگو کے ٹکڑے (جیسا کہ شکل 8.3 میں دکھایا گیا ہے) ایک ڈھانچے کی تعمیر کے لیے ایک ساتھ آتے ہیں، HTML ٹیگ ایک ویب پیج بنانے کے لیے اکٹھے ہوتے ہیں۔



شکل 8.3: لگو کے ٹکڑے

8.3.1 HTML کی تاریخ

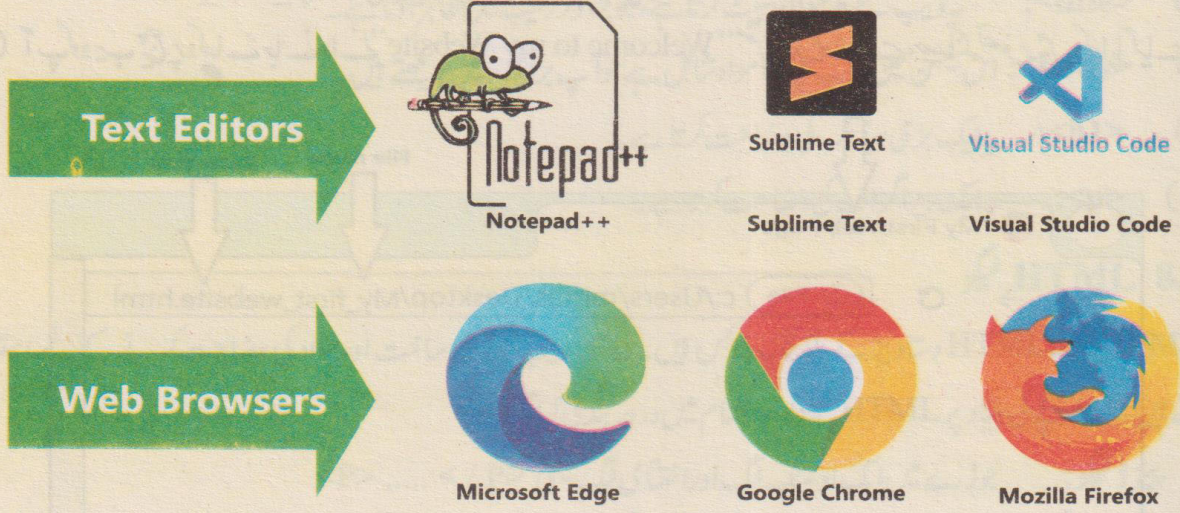
- HTML کوٹم برنرزی نے 1991ء میں بنایا تھا۔ اس کو انٹرنیٹ پر معلومات کے اشتراک کو آسان بنانے کے لیے ڈیزائن کیا گیا تھا۔ سالوں تک HTML کو زیادہ طاقتور اور استعمال میں آسان بنانے کے لیے بہت سی تبدیلیوں اور بہتریوں سے گزرنا پڑا ہے۔
- HTML 1.0 (1991ء): HTML کا پہلا ورژن۔ یہ آسان تھا اور متن اور لنکس بنانے کے لیے اس میں بنیادی خصوصیات تھیں۔
- HTML 2.0 (1995): مزید ٹیگ اور خصوصیات متعارف کروائیں، جس سے زیادہ پیچیدہ ویب صفحات بنانے کی اجازت ملتی ہے۔
- HTML 3.2 (1997ء): جدول، اسکرپٹ، اور ایپلٹ بنانے کے لیے نئے ٹیگ شامل کیے گئے۔
- HTML 4.0 (1997ء): بڑی بہتریاں لائی گئیں بشمول تصاویر اور ویڈیوز جیسے ملٹی میڈیا عناصر کی حمایت۔
- HTML 4.01 (1999ء): ورژن 4.0 میں معمولی بہتری
- HTML 5 (2014ء): HTML کا تازہ ترین ورژن۔ اس میں بہتر ملٹی میڈیا سپورٹ، گرافکس اور زیادہ پُرکشش ویب صفحات کے لیے نئے عناصر شامل ہیں۔

8.3.2 ڈویلپمنٹ انوائرنمنٹ کا قیام

- ویب سائٹس بنانا شروع کرنے کے لیے، آپ کو کچھ بنیادی ٹولز اور انوائرنمنٹ کی ضرورت ہے:
- (i) ٹیکسٹ ایڈیٹر: یہ وہ جگہ ہے جہاں آپ اپنا HTML کوڈ لکھتے ہیں۔ مقبول ٹیکسٹ ایڈیٹرز میں Sublime، NotePad++ اور Text Visual Studio Code شامل ہیں۔
- (ii) ویب براؤزر: آپ اسے اپنی HTML فائلوں کو دیکھنے اور جانچنے کے لیے استعمال کریں گے۔ عام ویب براؤزرز Google Chrome، Mozilla Firefox اور Microsoft Edge ہیں۔

کیا آپ جانتے ہیں؟

ایک سادہ ٹیکسٹ ایڈیٹر اور ویب براؤزر سے شروع کریں۔ ایک بار جب آپ HTML کے ساتھ آرام دہ اور پرسکون، آپ مزید جدید ٹولز کو تلاش کر سکتے ہیں۔



تصویر 8.4: ویب سائٹ ڈویلپمنٹ پر ماحول کے بنیادی عناصر

8.3.3 ویب پیج بنانا

ایک بنیادی HTML ایپلی کیشن بنانے کے لیے جو کسی ویب پیج پر ایک پیغام کو ظاہر کرتی ہے، ان آسان اقدامات پر عمل کریں:

8.3.3.1 HTML کوڈ لکھنا

- اپنا ٹیکسٹ ایڈیٹر کھولیں۔ آپ Sublime Text، NotePad++، NotePad یا کوئی اور ٹیکسٹ ایڈیٹر استعمال کر سکتے ہیں۔
- اپنے ٹیکسٹ ایڈیٹر میں درج ذیل HTML کوڈ لکھیں۔
- اپنی فائل کو HTML ایکسٹینشن کے ساتھ محفوظ کریں، مثال کے طور پر، My_first_website.html،

HTML کوڈ کی مثال

```
<!DOCTYPE html>
<html>
<head>
<title>My First Web Page </title>
</head>
<body>
<h1> Welcome to My Website </h1>
```

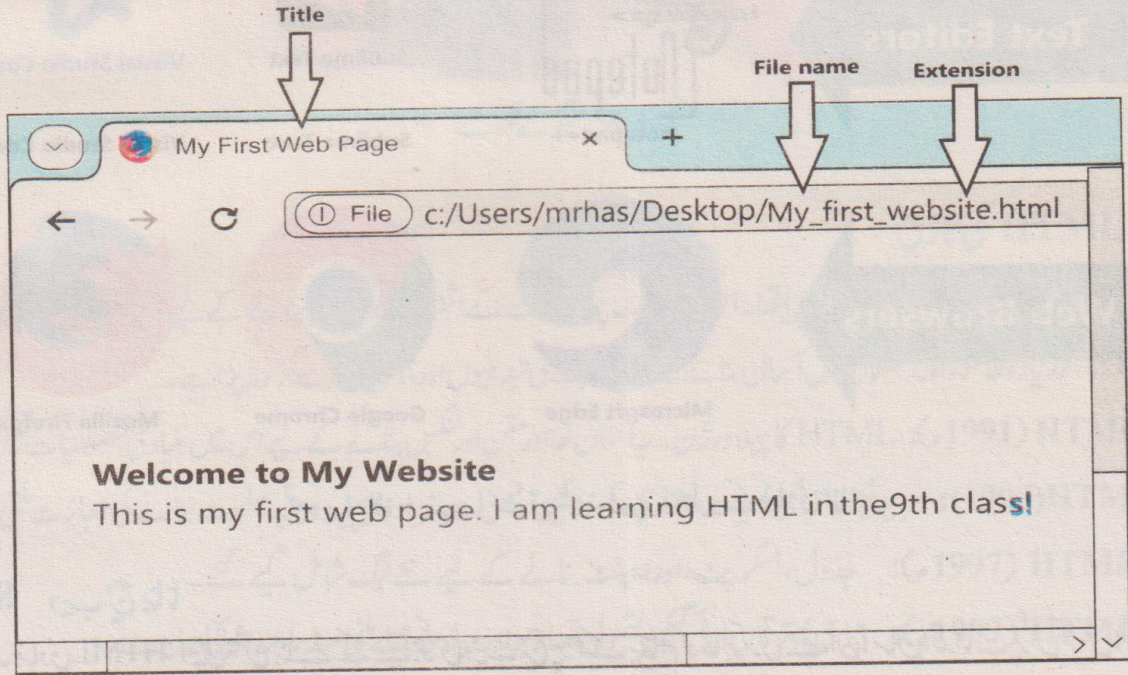

<p> This is my first web page. I am learning HTML in 9th class! </p>

</body>

</html>

HTML 8.3.3.2 فائل کو دیکھنا

- (i) اپنا ویب براؤزر کھولیں (گوگل کروم، موزیلا فائر فاکس، وغیرہ)۔
- (ii) My_first_website.html نامی اپنی فائل پر ڈبل کلک کریں۔
- (iii) آپ کو ویب پیج پر دکھائے جانے والے "Welcome to my Website" متن کو دیکھنا چاہیے، جیسا کہ تصویر میں دکھایا گیا ہے۔



شکل 8.5: ویب براؤزر میں HTML

دلچسپ معلومات:

اپنی HTML فائلوں کو محفوظ کرتے وقت، ہمیشہ اپنی فائل کے نام کے آخر میں .html ایکسٹینشن شامل کرنا یاد رکھیں۔ اس سے آپ کے کمپیوٹر کو اسے ویب پیج کے طور پر پہچاننے میں مدد ملتی ہے۔ اگر آپ نوٹ پیڈ استعمال کر رہے ہیں تو "ٹائپ کے طور پر محفوظ کریں" ڈراپ ڈاؤن میں "All Files" کو منتخب کریں۔ پھر .html کے بعد اپنی فائل کا نام ٹائپ کریں (مثال کے طور پر، _hello_world.html)۔

کیا آپ جانتے ہیں؟

اگر آپ HTML فائل میں کوئی تبدیلی کرتے ہیں تو، تازہ ترین مواد دیکھنے کے لیے اپنے براؤزر میں ویب پیج کو ریفرش کریں۔

HTML 8.4 بنیادی ڈھانچہ

ایک منظم HTML دستاویز پڑھنے اور سمجھنے کے لیے آسان ہے۔ مناسب طریقے سے نیسٹڈ اور اچھی طرح سے منظم عناصر ڈویلپر ز اور براؤزرز کو مواد کی صحیح تشریح کرنے میں مدد کرتے ہیں۔ یہ تنظیم اس بات کو یقینی بناتی ہے کہ ویب پیج ارادے کے مطابق ظاہر ہو۔

ہر HTML دستاویز میں ایک بنیادی ڈھانچہ ہوتا ہے جہاں:

(i) `<!DOCTYPE html>` : یہ لائن براؤزر کو بتاتی ہے کہ یہ ایک HTML 5 کی دستاویز ہے۔

(ii) `<html>` : یہ HTML صفحے کا بنیادی یا روت عنصر ہے۔

(iii) `<head>` : اس سیکشن میں عنوان کی طرح HTML دستاویز کے بارے میں مینا معلومات شامل ہیں۔ جیسا کہ ٹائٹل

(iv) `<title>` : یہ ویب صفحے کا عنوان سیٹ کرتا ہے، جو براؤزر ٹیب میں ظاہر ہوتا ہے۔

(v) `<body>` : اس سیکشن میں ویب پیج کا مواد شامل ہے جو آپ براؤزر میں دیکھتے ہیں۔

(vi) `<h1>` : یہ ایک بڑی ہیڈنگ کی وضاحت کرتا ہے۔

(vii) `<p>` : یہ ایک پیراگراف کی وضاحت کرتا ہے۔

HTML 8.4.1 ٹیگر

HTML ٹیگ HTML دستاویز کے بلڈنگ بلاکس ہیں۔ وہ کسی ویب پیج کی ساخت اور مواد کی وضاحت کرنے کے لیے استعمال ہوتے ہیں۔ ڈھانچے کی بنیاد پر HTML ٹیگز کو دو اقسام میں درجہ بندی کیا گیا ہے

(i) پیئرڈ ٹیگز: جوڑے میں ایک اوپننگ ٹیگ اور اختتامی ٹیگ یعنی `<P> ..... </P>`

(ii) آن پیئرڈ ٹیگز: ٹیگ بند کرنے کی ضرورت نہیں ہے۔ انھیں خود کو بند کرنے والے کے طور پر بھی جانا جاتا ہے۔ یہ ٹیگز `<br>`، `<img>` ہیں۔

HTML 8.5 کے ساتھ مواد بنانا

HTML میں مواد کسی ویب پیج پر اہم معلومات ہے جسے صارفین پڑھتے ہیں اور ان کے ساتھ بات چیت کرتے ہیں۔ اس میں متن، تصاویر، ویڈیوز، لنکس، اور دیگر عناصر شامل ہیں جو صفحے کے مقصد اور پیغام کو پہنچاتے ہیں۔ اس سے لوگوں کو آپ کی سائٹ تلاش کرنا آسان ہو جاتا ہے۔

8.5.1 ہیڈنگز

HTML میں ہیڈنگز `<h1>` سے لے کر `<h6>` تک ہیں۔ یہ ویب پیج پر مواد کی ساخت اور درجہ بندی کی وضاحت کرنے کے لیے استعمال ہوتے ہیں۔ یہاں وہ مندرجہ ذیل وجوہات کی وجہ سے اہم ہیں:

ہیڈنگز کی اہمیت

(i) مواد کے عنوانات کو منظم کرنے سے مواد کو حصوں اور ذیلی حصوں میں منظم کرنے میں مدد ملتی ہے، جس سے صارفین کے لیے پڑھنا اور سمجھنا آسان ہو جاتا ہے۔ `<h1>` عام طور پر صفحے کے مرکزی ہیڈنگ کے لیے استعمال کیا جاتا ہے، جب کہ `<h1>` سے `<h6>` کو اہمیت کے کم ہوتے ہوئے ترتیب میں ذیلی ہیڈنگز کے لیے استعمال کیا جاتا ہے۔

(ii) سرچ انجن آپٹیمائزیشن (SEO): سرچ انجن کسی ویب پیج کی ساخت اور اہم موضوعات کو سمجھنے کے لیے عنوانات کا استعمال کرتے ہیں۔

عنوانات کا مناسب استعمال صفحے کے SEO کو بہتر بنا سکتا ہے، جس سے اسے تلاش کے نتائج میں اعلیٰ درجہ حاصل کرنے میں مدد ملتی ہے۔
(iii) مستقل فارمیٹنگ: معیاری عنوان ٹیگ کا استعمال مختلف براؤزرز اور آلات میں مستقل فارمیٹنگ کو یقینی بناتا ہے۔

8.5.1.1 مثال

درجہ بندی کے ڈھانچے میں مواد کو منظم کرنے کے لیے مختلف ہیڈنگز کی سطحوں کو کس طرح استعمال کیا جاسکتا ہے اس کی ایک مثال یہ ہے:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Importance of Headings in HTML</title>
```

```
</head>
```

```
<body>
```

```
<h1>Main Title (h1)</h1>
```

```
<p>This is the main title of the page.</p>
```

```
<h2>Subheading 1 (h2)</h2>
```

```
<p>This is a subheading under the main title.</p>
```

```
<h3>Subheading 2 (h3)</h3>
```

```
<p>This is a subheading under Subheading 1.</p>
```

```
</body>
```

```
</html>
```

آؤٹ پٹ

Main Title (h1)

This is the main title of the page.

Subheading 1 (h2)

This is a subheading under the main title.

Subheading 2 (h3)

This is a subheading under Subheading 1.

Subheading 3 (h4)

This is a subheading under Subheading 2.

8.5.2 پیراگراف

HTML میں پیراگراف متن کو پڑھنے کے قابل حصوں میں منظم اور الگ کرنے کے لیے استعمال ہوتے ہیں۔ ہر پیراگراف اس کے اوپر اور نیچے جگہ کے ساتھ متن کا ایک بلاک بناتا ہے، جس سے مواد کو پڑھنا آسان ہو جاتا ہے۔

پیراگراف `<p>` ٹیگ سے شروع ہوتا ہے اور یہ `</p>` کے ساتھ ختم ہوتا ہے۔

8.5.3 لنکس

HTML میں لنکس ایک ویب پیج کو دوسرے سے مربوط کرنے کے لیے استعمال ہوتے ہیں۔ وہ آپ کو ایک ہی صفحے کے مختلف حصوں یا انٹرنیٹ پر دوسرے صفحات پر جانے کے لیے الفاظ یا تصاویر پر کلک کرنے دیتے ہیں۔

لنکس کو `<a>` ٹیگ کی مدد سے بنایا جاتا ہے۔

```
<a href="https://www.example.com">Visit Example.com</a>
```

```
<a href="mailto:example@example.com">Send Email</a>
```

بعض اوقات HTML میں لنکس آپ کو ای میل بھیجنے کے لیے کلک کرنے دے سکتے ہیں۔ یہ خصوصی لنکس `mailto` سے شروع ہوتے ہیں اور جب آپ ان پر کلک کرتے ہیں تو، وہ آپ کا ای میل پروگرام کھولتے ہیں تاکہ آپ لنک میں ای میل ایڈریس پر پیغام بھیج سکیں۔

8.5.4 تصاویر

تصاویر HTML میں اہم ہیں کیونکہ وہ ویب صفحات کو زیادہ پرکشش بناتی ہیں۔ مزید برآں، تصاویر کا استعمال برانڈنگ میں مدد کرتا ہے، کیونکہ لوگوں اور مخصوص مناظر صارفین کے لیے برانڈ کو پہچاننا آسان بناتے ہیں۔ آخر میں، تصاویر کے لیے `Alt` ٹیکسٹ شامل کرنا اس بات کو یقینی بناتا ہے کہ بصارت سے محروم صارفین یہ سمجھ سکیں کہ تصاویر کیا نمائندگی کرتی ہیں۔

تصاویر `<img>` ٹیگ کا استعمال کرتے ہوئے شامل کی جاتی ہیں۔

```

```

```
<p>“A picture speaks louder than words.”</p>
```

8.5.5 لسٹیں

لسٹیں پیچیدہ خیالات کو آسان حصوں میں توڑ کر پڑھنے کی صلاحیت کو بہتر بناتی ہیں، جس سے صارفین آسانی سے تفصیلات اسکین کر سکتے ہیں۔ مجموعی طور پر، لسٹیں مواد کو ہر ایک کے لیے زیادہ منظم اور قابل رسائی بناتی ہیں۔

آپ آرڈرڈ (ترتیب وار) اور ان آرڈرڈ (غیر ترتیب) فہرستیں تشکیل دے سکتے ہیں۔

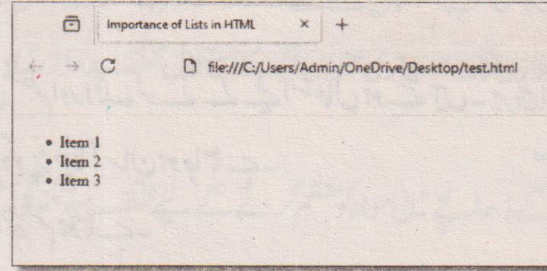
8.5.5.1 ان آرڈر فہرست

Item 1

Item 2

Item 3

Result



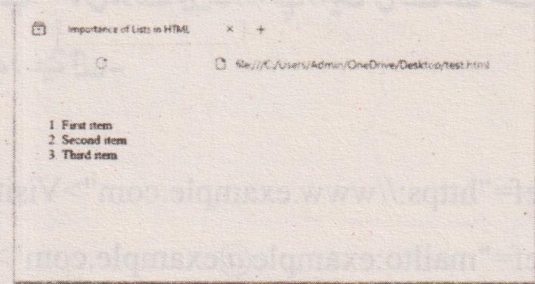
8.5.5.2 آرڈر فہرست

First item

Second item

3rd item

Result



8.5.6 HTML میں ٹیبل بنانا

HTML میں ٹیبلز کو ایک منظم شکل میں ڈیٹا کو ظاہر کرنے کے لیے استعمال کیا جاتا ہے جس سے معلومات کا آسان موازنہ اور تنظیم کی اجازت ملتی ہے۔ ایک ٹیبل <Table> ٹیگ کا استعمال کرتے ہوئے بنایا گیا ہے، جس میں <tr> (table row) ٹیگ کے ذریعہ وضاحت کردہ قطاریں شامل ہیں، اور ہر قطار <td> (table data) ٹیگ کے ذریعے نمائندگی کرنے والے سیلز (Cells) پر مشتمل ہے۔ مزید برآں، اعداد و شمار کے لیے سیاق و سباق فراہم کرنے کے لیے <th> (Table Header) ٹیگ کا استعمال کرتے ہوئے ٹیبل کے عنوانات شامل کیے جاسکتے ہیں۔

<table>

<tr>

<th>Name</th>

<th>Age</th>

</tr>

<tr>

<td>Alice</td>

<td>14</td>

</tr>

<tr>

آؤٹ پٹ

Name	Age
Alice	14
Bob	15


```



```

HTML 8.5.7 کمٹس

HTML میں کمٹس مندرجہ ذیل وجوہات کی بنا پر مفید ہو سکتے ہیں:

- (i) کوڈ کے ایک مخصوص سیکشن کے مقصد کی وضاحت
- (ii) مستقبل کی ترامیم کے لیے یاد دہانیاں چھوڑنا
- (iii) ٹیسٹنگ مقاصد کے لیے کوڈ کو عارضی طور پر غیر فعال کرنا

HTML کمٹس کے سینٹیکس

HTML کمٹس <!--...!> سے شروع ہوتے ہیں اور <!--...!> سے ختم ہوتے ہیں۔ ان مارکرز کے اندر رکھے گئے کسی بھی متن کو کمٹس کے طور پر سمجھا جائے گا اور براؤزر میں دکھایا جائے گا۔ <!--...!> This is a comment..!>

CSS 8.6 کے ساتھ اسٹائلنگ

ویب صفحات کی بصری ظاہری شکل کو بڑھانے اور صارف کے تجربے کو بہتر بنانے کے لیے Cascading Style Sheet (CSS) کے ساتھ اسٹائل کرنا ضروری ہے۔ CSS ڈویلپرز کو HTML عناصر کی ترتیب، رنگ، فونٹ اور مجموعی ڈیزائن کو کنٹرول کرنے کی اجازت دیتا ہے، جو مواد کو پریزنٹیشن سے الگ کرتا ہے۔ CSS مخصوص عناصر پر اسٹائل لاگو کرنے کے لیے مختلف خصوصیات اور سلیکٹرز پیش کرتا ہے، جو رسپانسیو ڈیزائن کو ممکن بناتے ہوئے مختلف اسکرین سائز اور آلات کے مطابق ایڈجسٹ ہوتا ہے۔

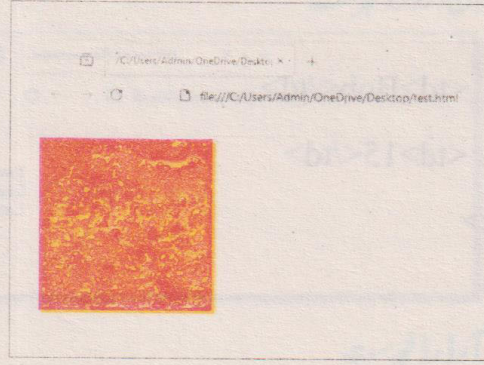
CSS 8.6.1 کی بنیادی ڈھانچہ

CSS کی بنیادی ساخت HTML عناصر پر اسٹائل کو مؤثر طریقے سے لاگو کرنے کے لیے ضروری ہے۔ CSS قواعد پر مشتمل ہے جو سلیکٹرز اور ڈیکلریشن پر مشتمل ہے۔ سلیکٹرز اس بات کی وضاحت کرتے ہیں کہ اسٹائل کس HTML عناصر پر لاگو ہوں گے، جبکہ ڈیکلریشن خصوصیات اور اقدار سمیت لاگو ہونے والے مخصوص اسٹائل کی وضاحت کرتے ہیں۔ ایک عام CSS اصول اس فارمیٹ کی پیروی کرتا ہے:

selector { property: value; }

مثال کے طور پر، ایک سادہ CSS اصول ویب پیج پر تمام عنوانات کا رنگ اور سائز تبدیل کر سکتا ہے:


```
h1 {
    color: red;
    font-size: 24px; }
```



اس مثال میں CSS تمام `<h1>` عناصر کو نشانہ بناتا ہے، ان کے متن کا رنگ نیلے اور فونٹ کا سائز 24 پکسلز پر سیٹ کرتا ہے۔

8.6.2 HTML میں CSS کو منسلک کرنا

CSS کو HTML کے ساتھ ایچ کرنا ویب صفحات کو اسٹائل کرنے کے لیے ضروری ہے اور یہ تین بنیادی طریقوں یعنی (i) ان لائن اسٹائل، (ii) انٹرئل اسٹائل اور (iii) ایکسٹرنل اسٹائل سے کیا جاسکتا ہے۔

(i) ان لائن اسٹائلز: اس طریقہ کار میں اسٹائل کی خصوصیت کا استعمال کرتے ہوئے CSS کو براہ راست انفرادی HTML عناصر میں شامل کرنا ہے۔ مثال کے طور پر:

```
<h1 style="color: blue;">Hello World</h1>
```

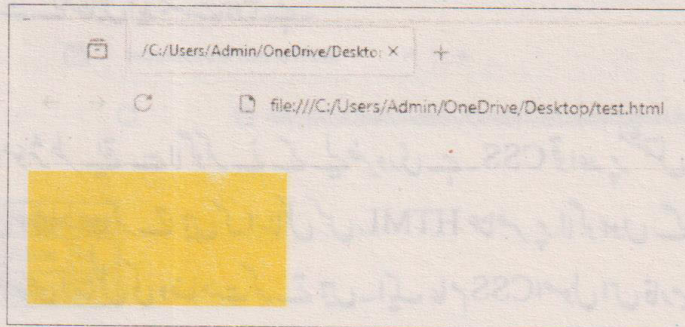
ہیڈنگ کا رنگ نیلے رنگ میں تبدیل کرتا ہے۔ اگرچہ فوری تبدیلیوں کے لیے آسان ہے لیکن ان لائن اسٹائل کو ڈکو بے ترتیب اور کم برقرار رکھنے کے قابل بنا سکتے ہیں۔

(ii) انٹرئل اسٹائلز: CSS کو `<style>` ٹیگ کا استعمال کرتے ہوئے HTML ڈاکیومنٹ کے `<head>` سیکشن میں بھی شامل کیا جاسکتا ہے۔ یہ طریقہ آپ کو دوسرے صفحات کو متاثر کیے بغیر پورے صفحے کے لیے اسٹائل کی وضاحت کرنے کی اجازت دیتا ہے۔ مثال کے طور پر:

```
<style>
```

```
h1 {
    color: yellow;
}
```

```
</style>
```



(iii) ایکسٹرنل اسٹائلز: بڑے پراجیکٹس کے لیے سب سے مؤثر طریقہ ایکسٹرنل CSS فائل کا استعمال کرنا ہے، جو `<head>` میں `<link>` ٹیگ کے ساتھ HTML دستاویز سے منسلک ہے۔ یہ HTML کو صاف رکھتا ہے اور متعدد صفحات میں آسان اپ ڈیٹس کی اجازت دیتا ہے۔ مثال کے طور پر:

<link rel="stylesheet" href="styles.css">

ان طریقوں سے CSS کو ضم کر کے، ڈویلپر زبصری طور پر پرکشش اور اچھی طرح سے منظم ویب صفحات تشکیل دے سکتے ہیں جو صارف کے تجربے کو بہتر بناتے ہیں۔

8.6.3 فونٹس، کلرز، پس منظر اسٹائلنگ فونٹس کے ساتھ HTML عناصر کو اسٹائل کرنا

آپ CSS کا استعمال کرتے ہوئے اپنے ویب پیج پر متن کی ظاہری شکل کو تبدیل کر سکتے ہیں۔ اس میں فونٹ، سائز اور اسٹائل کو تبدیل کرنا شامل ہے۔

کیا آپ جانتے ہیں؟

آپ سی ایس ایس میں مختلف فونٹس کے نام بتا کر استعمال کر سکتے ہیں۔ کے لیے مثال کے طور پر، آپ فونٹ کو Times New Roman یا Arial پریسٹ کر سکتے ہیں۔

اسٹائلنگ فونٹس کی مثال

دیکھیے آپ پیراگراف کے فونٹ کو کس طرح اسٹائل کر سکتے ہیں:

P {

font-family: Arial, sans-serif;

font-size: 16px;

font-weight: bold;

font-style: italic;

}

اس مثال میں:

(i) فونٹ فیملی : Arial, SansSerif فونٹ کو Arial پریسٹ کرتا ہے۔ اگر Arial دستیاب نہیں ہے تو یہ ایک عام SansSerif فونٹ استعمال کرے گا۔

(ii) فونٹ کا سائز : 16px؛ فونٹ کا سائز 16 پیکسلز پریسٹ کرتا ہے۔

(iii) فونٹ وزن : Bold (بولڈ)؛ متن کو بولڈ بناتا ہے۔

(iv) فونٹ اسٹائل : Italic (ترچھا)؛ متن کو ترچھا بناتا ہے۔

8.6.3.1 بیک گراؤنڈ کو اسٹائل کرنا

CSS ایک ایسی زبان ہے جو ویب پیجز کو اسٹائل کرنے کے لیے استعمال ہوتی ہے۔ CSS کے ساتھ آپ جو تفریحی کام کر سکتے ہیں ان میں سے ایک یہ ہے کہ آپ اپنے ویب پیج کا بیک گراؤنڈ یا اس پر موجود عناصر کو تبدیل کریں۔ یہاں کچھ طریقے ہیں جن سے آپ CSS

کے ساتھ بیک گراؤنڈ کو اسٹائل کر سکتے ہیں:

- (i) بیک گراؤنڈ کا رنگ: آپ پس منظر کے رنگ کی پراپرٹی کا استعمال کرتے ہوئے اپنے ویب پیج یا کسی بھی HTML عنصر کے بیک گراؤنڈ کا رنگ تبدیل کر سکتے ہیں۔ مثال کے طور پر، اگر آپ نیلے رنگ کا بیک گراؤنڈ چاہتے ہیں تو، آپ لکھ سکتے ہیں:

```
body { background-color: blue; }
```

- (ii) بیک گراؤنڈ کی تصویر: آپ بیک گراؤنڈ۔ امیج پراپرٹی کا استعمال کرتے ہوئے بیک گراؤنڈ کے طور پر تصویر سیٹ کر سکتے ہیں۔ مثال کے طور پر، کسی تصویر کو بیک گراؤنڈ کے طور پر سیٹ کرنے کے لیے، آپ لکھ سکتے ہیں:

```
body { background-image: url("your-image.jpg"); }
```

- (iii) بیک گراؤنڈ کا اعادہ: بعض اوقات، بیک گراؤنڈ کی تصویر چھوٹی ہوتی ہے، اور آپ چاہتے ہیں کہ اسے پورے صفحے پر دہرایا جائے۔ آپ ایسا کرنے کے لیے بیک گراؤنڈ دہرانے والی پراپرٹی کا استعمال کر سکتے ہیں۔ مثال کے طور پر:

```
body { background-image: url("your-image.jpg"); background-repeat: repeat; }
```

- (iv) بیک گراؤنڈ کی پوزیشن: بیک گراؤنڈ کی پوزیشن پراپرٹی کا استعمال کرتے ہوئے آپ بیک گراؤنڈ کی تصویر کو بالکل وہیں رکھ سکتے ہیں جہاں آپ اسے چاہتے ہیں۔ مثال کے طور پر:

```
body { background-image: url("your-image.jpg"); background-position: center; }
```

- (v) بیک گراؤنڈ کا سائز: آپ بیک گراؤنڈ کے سائز کی پراپرٹی کا استعمال کرتے ہوئے بیک گراؤنڈ کی تصویر کے سائز کو کنٹرول کر سکتے ہیں۔ مثال کے طور پر، بیک گراؤنڈ کی تصویر کے ساتھ پورے صفحے کا احاطہ کرنے کے لیے، آپ لکھ سکتے ہیں:

```
body { background-image: url("your-image.jpg"); background-size: cover; }
```

ان خصوصیات کا استعمال کرتے ہوئے، آپ اپنے ویب صفحات کے لیے رنگین اور دلچسپ پس منظر بنا سکتے ہیں۔

8.6.4 ترتیب بنانا اور مواد کو منظم کرنا

کسی ویب پیج پر ترتیب بنانا اور مواد کو منظم کرنا ویب ڈیزائن کا ایک اہم حصہ ہے۔ CSS آپ کو اپنے ویب پیج کے مختلف حصوں کو منظم طریقے سے ترتیب دینے میں مدد کرتا ہے۔ ترتیب بنانے اور مواد کو منظم کرنے کے لیے یہاں کچھ بنیادی طریقے ہیں:

- (i) Divs اور Sections: HTML عناصر جیسے <div> اور <section> مواد کو ایک ساتھ گروپ کرنے کے لیے استعمال ہوتے ہیں۔ اس کے بعد آپ ان گروپوں کو اسٹائل اور پوزیشن کرنے کے لیے CSS کا استعمال کر سکتے ہیں۔ مثال کے طور پر:

```
<div class = "container">
```

```
<section class="header">This is the header</section>
```

```
<section class="content">This is the main content</section>
```

```
<section class="footer">This is the footer</section> </div>
```


(ii) CSS گرڈ: CSS گرڈ لے آؤٹ پیچیدہ ترتیب بنانے کے لیے ایک طاقتور آلہ ہے۔ یہ آپ کو قطاروں اور کالموں میں آئٹمز کو ترتیب دینے کی اجازت دیتا ہے۔ مثال کے طور پر:

```
.container {  
    display: grid;  
    grid-template-columns: auto auto;  
    grid-gap: 10px;  
}
```

```
.item {  
    padding: 20px;  
    background-color: lightgrey;  
}
```

(iii) CSS فلیکس باکس: فلیکس باکس ایک اور لے آؤٹ ٹول ہے جو اشیا کو لچکدار اور رسپانسو طریقے سے ترتیب دینے میں مدد کرتا ہے۔ یہ کسی قطار یا کالم میں آئٹمز کو ترتیب دینے کے لیے مفید ہے۔ مثال کے طور پر:

```
.container {  
    display: flex;  
    justify-content: space-between;  
}
```

```
.item {  
    padding: 20px;  
    background-color: lightgrey;  
}
```

(iv) پوزیشننگ: CSS پوزیشننگ خصوصیات جیسے پوزیشن، اوپر، بائیں، دائیں، اور نیچے آپ کو ویب پیج پر عناصر کو بالکل وہیں رکھنے کی اجازت دیتا ہے جہاں آپ انہیں چاہتے ہیں۔ مثال کے طور پر:

```
.box {  
    position: absolute;  
    top: 50px;
```



```

left: 100px;
width: 200px;
height: 100px;
background-color: lightblue;

```

}

(v) مارجن اور پیڈنگ: عناصر کے ارد گرد اور اندر جگہ بنانے کے لیے مارجن اور پیڈنگ کا استعمال کیا جاتا ہے۔ مارجن عنصر کے باہر جگہ بناتے ہیں، جبکہ پیڈنگ عنصر کے اندر جگہ پیدا کرتی ہے۔ مثال کے طور پر:

```

.box { margin: 20px; padding: 10px; background-color: lightgrey; }

```

کیا آپ جانتے ہیں؟

اپنے ویب پیج کو ہمیشہ مختلف براؤزرز میں جانچیں تاکہ یہ یقینی بنایا جاسکے کہ یہ نظر آتا ہے اور ہر جگہ اسی طرح کام کرتا ہے۔ یہ آپ کو کسی بھی براؤزر سے متعلق مخصوص کوپکڑنے میں مدد کرتا ہے۔ ابتدائی مسائل۔

سرگرمی

اس سرگرمی میں آپ HTML اور CSS کا استعمال کرتے ہوئے ایک بنیادی ویب پیج ترتیب بنانے کی مشق کریں گے۔ ان اقدامات پر عمل کریں:

- 1۔ ایک نئی HTML فائل بنائیں اور اسے index.html نام دیں۔
- 2۔ HTML دستاویز کی بنیادی ساخت شامل کریں۔
- 3۔ <body> ٹیگ کے اندر، کلاس نام "کنٹینر" کے ساتھ <div> بنائیں۔
- 4۔ <div> کے اندر، کلاس کے نام "head"، "text"، اور "footer" کے ساتھ تین حصے شامل کریں۔
- 5۔ <head> سیکشن میں <link> ٹیگ کا استعمال کرتے ہوئے CSS فائل کو اپنی HTML فائل سے لنک کریں۔
- 6۔ گرڈ ترتیب کے ساتھ کنٹینر کلاس کو اسٹائل کرنے کے لیے CSS کا استعمال کریں، اور، ہیڈر، مواد، اور فٹر حصوں پر پس منظر کے رنگ لاگو کریں۔
- 7۔ اپنی ترتیب دیکھنے کے لیے براؤزر میں اپنے ویب پیج کی جانچ کریں۔

8.6.5 CSS کا استعمال کرتے ہوئے اپنی میٹن اور ٹرانزیکشن شامل کرنا

CSS اپنی میٹن اور ٹرانزیکشن آپ کے ویب صفحات کو حرکت اور اثرات شامل کر کے زیادہ دلچسپ بنا سکتی ہیں۔ آئیے ان کا استعمال

کرنے کا طریقہ سیکھیں!

8.6.5.1 اپنی میشن کا اضافہ

CSS آپ کو اپنے ویب پیج کو زیادہ انٹرایکٹو بنانے کے لیے اپنی میشن شامل کرنے کی اجازت دیتا ہے۔ اپنی میشن وقت کے ساتھ عناصر کو دیکھنے یا حرکت کرنے کے طریقے کو تبدیل کر سکتے ہیں۔ CSS کے ساتھ اپنی میشن بنانے کے لیے یہاں کچھ بنیادی اقدامات ہیں:

(i) کی فریمز کی وضاحت کریں: کی فریمز اپنی میشن کے آغاز اور اختتام کے نکات کے ساتھ کسی بھی درمیانی اقدامات کی وضاحت کرنے کے لیے استعمال ہوتے ہیں۔ مثال کے طور پر:

@keyframes example {

from {background-color: red;}

to {background-color: yellow;}

}

یہ کی فریم اپنی میشن کے بیک گراؤنڈ کا رنگ سرخ سے پیلے رنگ میں تبدیل کرتی ہے۔

(ii) اپنی میشن کا اطلاق کریں: کسی عنصر پر اپنی میشن کا اطلاق کرنے کے لیے، اپنی میشن کی پراپرٹی کا استعمال کریں۔ مثال کے طور پر:

.animated-box {

width: 100px; height:

100px; background-color: red;

animation-name: example; animation-duration: 4s;

}

اس سے باکس کا بیک گراؤنڈ کا رنگ 4 سیکنڈ کے لیے سرخ سے پیلے رنگ میں تبدیل ہو جائے گا۔

(iii) لوپ اور ٹائمنگ: آپ یہ بھی سیٹ کر سکتے ہیں کہ اپنی میشن کو کتنی بار دہرانا چاہئے اور اس کا ٹائمنگ فنکشن۔ مثال کے طور پر:

.animated-box {

animation-iteration-count: infinite; /* Animation will repeat forever */

animation timing-function: linear; /* Animation will progress at a constant speed

}

8.6.5.2 ٹرانزیشن کا اضافہ

CSS اسٹائل کو ہموار اور بصری طور پر پرکشش بنانے کے لیے اپنے ویب پیج میں ٹرانزیشن شامل کرنے کی اجازت دیتا ہے۔ ٹرانزیشن فوری طور پر کرنے کے بجائے آہستہ آہستہ رنگ، سائز، یا پوزیشن جیسی خصوصیات کو تبدیل کر سکتی ہے۔ CSS کے ساتھ ٹرانزیشن منتقلی بنانے کے لیے یہاں کچھ بنیادی اقدامات ہیں:

(i) ابتدائی اسٹائل سیٹ کریں: سب سے پہلے، اس عنصر کے لیے ابتدائی اسٹائل کی وضاحت کریں جسے آپ متحرک کرنا چاہتے ہیں۔

مثال کے طور پر:

```
.Box {
    width:100px;
    height: 100px;
    background-color:red;
    transition:background-color 2s, width 2s;
}
```

یہ باکس کے ابتدائی سائز اور رنگ کو سیٹ کرتا ہے، اور واضح کرتا ہے کہ بیک گراؤنڈ کے رنگ اور چوڑائی میں تبدیلی 2 سیکنڈ میں منتقل ہونی چاہیے۔

(ii) Hover اسٹیٹ کی وضاحت: آگے، عنصر کے اسٹائل کی وضاحت کریں جب یہ گھوم رہا ہو۔ مثال کے طور پر:

```
.box:hover{
    background-color: yellow; width: 200px;
}
```

یہ پس منظر کا رنگ پیلے رنگ میں تبدیل کر دے گا اور جب ماؤس باکس کے اوپر آئے گا تو چوڑائی دوگنی ہو جائے گی۔

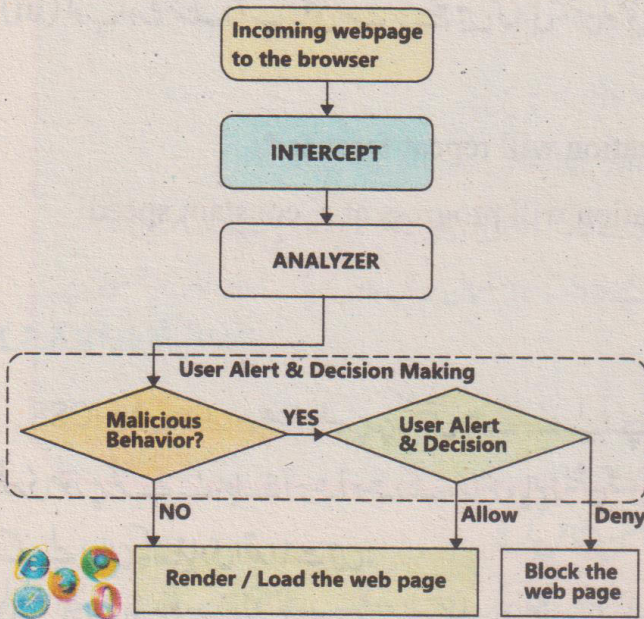
8.7 جاوا اسکریپٹ کا تعارف

جاوا اسکریپٹ ایک پروگرامنگ لینگویج ہے جو ویب سائٹس کو پُرکشش اور تفریحی بنانے کے لیے استعمال ہوتی ہے۔ یہ ڈویلپرز کو اپنی میٹھن، گیمز اور رسپانسیو خصوصیات جیسی چیزیں بنانے کی اجازت دیتا ہے، جب آپ بٹن پر کلک کرتے ہیں یا اپنے ماؤس کو منتقل کرتے ہیں تو رد عمل ظاہر کرتے ہیں۔

مثال کے طور پر، جب آپ کسی ویب پیج پر "Popup" پیغام دیکھتے ہیں یا جب آپ کسی تصویر کے اوپر ماؤس گھماتے ہیں تو کوئی تصویر تبدیل ہو جاتی ہے۔

فلو چارٹ کی شکل میں جاوا اسکریپٹ کا نفاذ شکل 8.7 میں دکھایا

گیا ہے۔



شکل 8.7: جاوا اسکریپٹ ایگزیکوشن

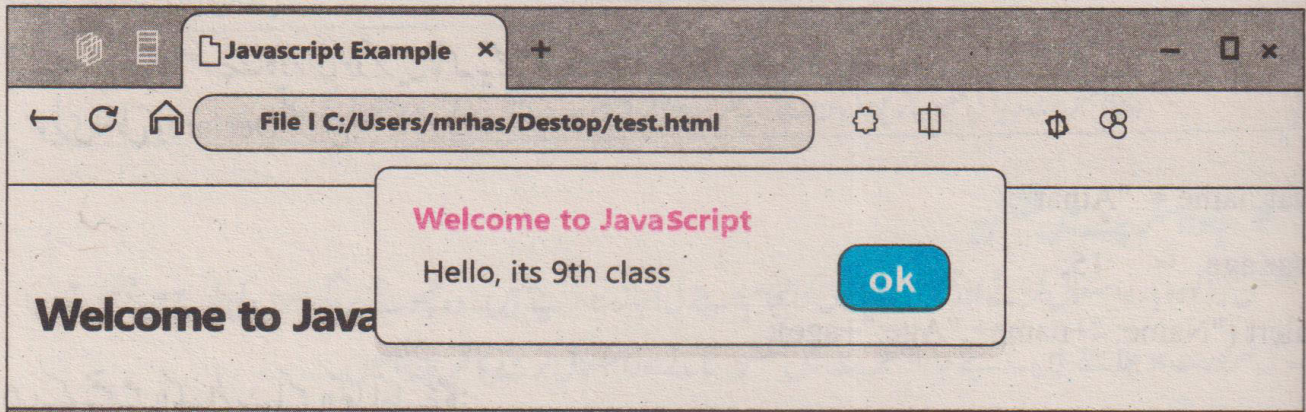
کیا آپ جانتے ہیں؟

جاوا اسکریپٹ کو صرف 10 دنوں میں برینڈن ایچی نے 1995 میں بنایا تھا۔ ابتدائی طور پر موجا، پھر LiveScript، اور آخر میں JavaScript کہا جاتا تھا۔

8.7.1 بنیادی سینٹیکس اور مثالیں

جاوا اسکریپٹ کا استعمال کرتے ہوئے الرٹ پیغام ظاہر کرنے کے لیے یہاں ایک آسان مثال ہے:

```
<!DOCTYPE html>
<html>
  <head>
    <title>JavaScript Example</title>
  </head>
<body>
  <h1>Welcome to JavaScript</h1>
  <script>
    alert("Hello, 9th Class Students!");
  </script>
</body>
</html>
```



شکل 8.8: الرٹ پیغام کی مثال

سرگرمی

سرگرمی: الرٹ دکھانا

- (i) ٹیکسٹ ایڈیٹر کھولیں (مثال کے طور پر، نوٹ پیڈ)۔
- (ii) HTML اور جاوا اسکریپٹ کوڈ لکھیں جیسا کہ اوپر دکھایا گیا ہے۔
- (iii) فائل کو index.html کے طور پر محفوظ کریں۔
- (iv) الرٹ پیغام دیکھنے کے لیے فائل کو ویب براؤزر میں کھولیں۔

8.7.1 ویری ایبل اور ڈیٹا کی اقسام

جاوا اسکریپٹ میں، آپ ویری ایبلز کا استعمال کرتے ہوئے ڈیٹا سٹور کر سکتے ہیں۔ ایک ویری ایبل ایک کنٹینر کی طرح ہے جس کے اندر معلومات رکھی ہوتی ہیں جنہیں آپ کے کوڈ میں استعمال کیا جاتا ہے۔

8.7.1.1 ویری ایبل کا اعلان

جاوا اسکریپٹ میں ویری ایبل کا اعلان کرنے کے لیے، آپ var، let، const استعمال کرتے ہیں۔ یہاں var کا استعمال کرتے ہوئے ایک مثال ہے:

```
<script>
    var name = "Athar";
    var age = 15;
    alert ("Name: "+name+", Age: "+age);
</script>
```

8.7.1.2 ڈیٹا کی رن کی مثال

آئیے مندرجہ بالا اسکریپٹ کا ڈیٹا کی رن کوڈ کریں تاکہ یہ سمجھ سکیں کہ یہ قدم بہ قدم کیسے کام کرتا ہے: ویری ایبل کو Declare کریں:

```
var name = "Athar";
var age = 15;
alert ("Name: "+name+", Age: "+age);
```

اس کے نتیجے میں ایک الرٹ باکس دکھایا جائے گا:

```
var name = "Athar";
var age = 15;
```


8.7.1.3 ڈیٹا کی اقسام

ویری ایبل مختلف قسم کے ڈیٹا کو سٹور کر سکتے ہیں۔ جاوا اسکرپٹ میں کچھ عام ڈیٹا اقسام یہ ہیں:

سٹرنگ (String): حروف کی ایک ترتیب، جو متن کے لیے استعمال ہوتی ہے۔

```
var name = "Athar"; //String
```

نمبر (Number): مکمل عدد اور فلوٹنگ پوائنٹ نمبرز دونوں کی نمائندگی کرتا ہے۔

```
var age = 15; //Number
```

بولین (Boolean): درست یا غلط ویلیوز کی نمائندگی کرتا ہے۔

```
var isStudent = true; //Boolean
```

ایرے (Array): ایک ہی ویری ایبل میں زیادہ قیمتوں کا سٹور کرنا۔

```
var scores = [90, 85, 88]; //Array
```

دلچسپ معلومات:

سٹرنگ کے لیے ہمیشہ کوٹیشن مارک کا استعمال کریں جبکہ نمبروں کے لیے کوٹیشن مارک استعمال نہ کریں۔

کیا آپ جانتے ہیں؟

ویری ایبل کو ڈیکلیئر کرنے کے بعد بھی اس کی ویلیو تبدیل کی جاسکتی ہے۔

سرگرمی

مختلف ڈیٹا ٹائپس کے لیے اپنے متغیرات بنائیں۔ مثال کے طور پر، اپنے پسندیدہ رنگ کے لیے ایک متغیر بنائیں، اپنی موجودہ گریڈ کے لیے یا یہ کہ آپ کو کوڈنگ پسند ہے یا نہیں۔

کیا آپ جانتے ہیں؟

جاوا اسکرپٹ دنیا کی مقبول ترین زبان ہے۔ یہ تقریباً 95.2% ویب سائٹس کے لیے استعمال کی جاتی ہے۔

8.7.2 جاوا اسکرپٹ میں فنکشن

فنکشن کوڈ کو دوبارہ استعمال کرنے اور مخصوص کاموں کو انجام دینے کی اجازت دیتے ہیں۔ وہ چھوٹے پروگراموں کی طرح ہیں جنہیں آپ جب بھی ضرورت ہو چلا سکتے ہیں۔ آئیے جاوا اسکرپٹ میں فنکشن بنانے اور استعمال کرنے کا طریقہ سیکھیں۔

8.7.2.1 سادہ فنکشن

یہ ایک سادہ فنکشن کی مثال ہے جو greet کا پیغام ظاہر کرتا ہے:


```
<script>
function greet() {
alert("Hello, Student!");
}
greet(); // This calls the function to execute
</script>
```

اس مثال میں:

- فنکشن greet() ایک فنکشن کو declare کرتا ہے جسے greet کا نام دیا گیا ہے۔
- alert ("Hellow, Student"); وہ کوڈ ہے جو اس وقت چلتا ہے جب فنکشن کو کال کیا جاتا ہے۔
- greet(); الٹ پیغام کو ظاہر کرنے کے لیے فنکشن کو کال کریں۔

8.7.2.2 پیرامیٹرز کے ساتھ فنکشن

بعض اوقات، آپ چاہتے ہیں کہ آپ کا فنکشن ان پٹ قیمتوں کے ساتھ کچھ کرے۔ آپ پیرامیٹرز کا استعمال کر کے ایسا کر سکتے ہیں۔

8.7.2.3 متعدد پیرامیٹرز کے ساتھ فنکشن

آپ ایسے فنکشن بھی تشکیل دے سکتے ہیں جو متعدد پیرامیٹرز لیتے ہیں۔ یہاں ایک مثال ہے:

```
<script>
function addNumbers(a, b) {
var sum = a + b;
alert("The sum is: " + sum);
}
addNumbers(5, 3); // This calls the function with the parameters 5 and 3
</script>
```

AddNumbers (3,5)، یہ فنکشن کو پیرامیٹرز 3 اور 5 کے ساتھ کال کرتا ہے۔

پیرامیٹرز ان قیمتوں کے لیے Place holder ہیں جو آپ فنکشن میں پاس کرتے ہیں۔ آپ ان اقدار کو اپنے فنکشن کے اندر استعمال کر سکتے ہیں۔

دلچسپ معلومات:

پیرامیٹرز وہ جگہیں ہیں جہاں آپ وہ اقدار رکھتے ہیں جو آپ فنکشن کو دیتے ہیں۔ آپ ان اقدار کو اپنے فنکشن کے اندر استعمال کر سکتے ہیں۔

اس مثال میں:

(1) فنکشن addNumbers (a,b) ایک فنکشن کو کال کرتا ہے جو دو پیرامیٹرز لیتا ہے a اور b۔

(2) var sum = a+b اور b کو آپس میں جمع کرتا ہے۔

(3) alert("The Sum is: "+sum)، جمع ہونے والے ویری ایبلز کا رزلٹ دکھاتا ہے۔

(4) addNumbers (3,5) فنکشن کو، آرگیومنٹ 3، 5 کے ساتھ کال کرتا ہے اور الٹ میسج میں نتیجہ 8 دکھاتا ہے۔

کیا آپ جانتے ہیں؟

فنکشن کسی بھی تعداد میں پیرامیٹرز لے سکتے ہیں۔ آپ ہر بار جب فنکشن کو کال کرتے ہیں تو مختلف اقدار پاس کر سکتے ہیں۔

سرگرمی

ایک فنکشن بنانا ہے کہ:

- 1۔ ایک فنکشن لکھیں جو نام کو ایک پیرامیٹر کے طور پر لیتا ہے اور ایک ذاتی نوعیت کا سلام ظاہر کرتا ہے۔ اپنے نام کے ساتھ فنکشن کو کال کریں تاکہ آپ سلام دیکھ سکیں۔
- 2۔ ایک فنکشن لکھیں جو کسی مستطیل کا رقبہ اس کی لمبائی اور چوڑائی کے دیے گئے اعداد و شمار کے ساتھ حساب کرے۔

8.7.3 ہینڈلنگ ایونٹس اور یوزر ان پٹ

جاوا اسکریپٹ آپ کو ایونٹس اور یوزر کے ان پٹ کو سنبھال کر اپنے ویب پیج کو پُرکشش بنانے کی اجازت دیتا ہے۔ ایک ایونٹ ایک عمل ہے جو اس وقت ہوتا ہے جب صارف کسی ویب پیج کے ساتھ تعامل کرتا ہے، جیسے بٹن پر کلک کرنا یا ایک کی دبانا۔

HTML کے ایونٹس

HTML ایونٹس وہ اقدامات ہیں جو براؤزر میں ہوتے ہیں اور اکثر صارف کے عمل سے شروع ہوتے ہیں۔ جب کوئی مخصوص واقعہ پیش آتا ہے تو جاوا اسکریپٹ کوڈ کو نافذ کر کے ویب پیج کو پُرکشش بنانے کے لیے ایونٹس کا استعمال کیا جاسکتا ہے۔

عام HTML ایونٹس

یہاں کچھ عام ایونٹس ہیں جن کا آپ سامنا کر سکتے ہیں:

- (i) آن کلک (OnClick): جب کسی عنصر پر کلک کیا جاتا ہے تو متحرک ہوتا ہے۔
- (ii) آن لوڈ (OnLoad): جب کسی صفحے یا تصویر کی لوڈنگ ختم ہو جاتی ہے تو متحرک ہوتا ہے۔
- (iii) آن ماؤس اوور (OnMouseOver): جب ماؤس پوائنٹر کسی عنصر کے اوپر حرکت کرتا ہے تو متحرک ہوتا ہے۔
- (iv) آن ماؤس آؤٹ (OnMouseOut): جب ماؤس پوائنٹر کسی عنصر سے باہر نکل جاتا ہے تو متحرک ہوتا ہے۔
- (v) آن کی اپ (OnKeyUp): جب کی بورڈ پر ایک کی کو چھوڑا جاتا ہے تو متحرک ہوتا ہے۔

8.7.3.1 جاوا اسکریپٹ کے ساتھ واقعات اور صارف کے تعامل کا انتظام کرنا

آئیے مرحلہ وار ایونٹ اور یوزر کے عمل کا انتظام کرنے کا طریقہ سیکھتے ہیں۔

ایک ایونٹ ہینڈلر ایک فنکشن ہے جو اس وقت چلتا ہے جب کوئی مخصوص واقعہ پیش آتا ہے۔ آپ ایونٹ ہینڈلرز کو HTML عناصر میں شامل کر سکتے ہیں تاکہ وہ صارف کے اقدامات کا جواب دے سکیں۔

مثال: بٹن کلک ایونٹ (ButtonClickEvent)
بٹن کلک ایونٹ کو سنبھالنے کے طریقے کی ایک مثال یہ ہے:

```
<!DOCTYPE html>
```

```
<html >
```

```
<head>
```

```
<title>Button Click Event</title>
```

```
<script>
```

```
function showMessage() {  
    alert("Button was clicked!");
```

```
}
```

```
</script>
```

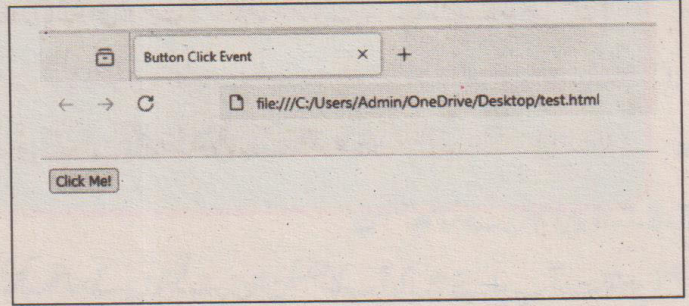
```
</head>
```

```
<body>
```

```
<button onclick="showMessage()">Click Me!</button>
```

```
</body>
```

```
</html>
```



دلچسپ معلومات:

آپ کے فنکشنز کے متعلق نام ان کے کام کو سمجھنے میں آسانی پیدا کرتے ہیں۔

8.7.4 انٹرایکٹو عناصر بنانا

اس سیکشن میں، ہم سیکھیں گے کہ سادہ پروگراموں اور فارمز کو تیار کر کے اپنے ویب پیجز کو پُرکشش کیسے بنایا جائے۔ ہم یہ بھی سیکھیں گے کہ انٹرایکٹو فعالیت کو شامل کرنے کے لیے جاوا اسکریپٹ کو HTML کے ساتھ کیسے ضم کیا جائے۔

8.7.4.1 سادہ پروگرام اور فارم تیار کرنا

فارم صارفین کو ڈیٹا ان پٹ کرنے کی اجازت دیتے ہیں، جسے جاوا اسکریپٹ کا استعمال کرتے ہوئے پروسیس کیا جاسکتا ہے۔ یہاں سادہ فارم کی ایک مثال ہے جو صارف کا نام لیتی ہے اور greet کا پیغام ظاہر کرتی ہے۔

مثال: سادہ شکل

درج ذیل مواد کے ساتھ ایک index.html فائل بنائیں:

182


```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Change Background Color</title>
```

```
<script src="script.js"></script>
```

```
</head>
```

```
<body>
```

```
<h1>Welcome to the Color Changer!</h1>
```

```
<button onclick="changeColor()">Change Background Color</button>
```

```
</body>
```

```
</html>
```

Add the following content to your script.js file:

```
function changeColor() {
```

```
document.body.style.backgroundColor = "lightblue";
```

اپنی script.js فائل میں درج ذیل مواد شامل کریں:

```
function changeColor() {
```

```
document.body.style.backgroundColor = "lightblue";
```

```
}
```

کیا آپ جانتے ہیں؟

آپ جاوا اسکریپٹ کا استعمال کر کے کسی بھی HTML میں CSS کی خاصیت کو تبدیل کر سکتے ہیں، جیسے رنگ، فونٹ سائز، یا بصیرت۔

سرگرمی

ایک بٹن بنائیں جس پر کلک کرنے سے پیج کے تمام چیزیں غائب ہو جائیں۔ HTML اور JavaScript کے مختلف ایلیمنٹ کے ساتھ تجربات کریں۔

جاوا اسکریپٹ کو HTML کے ساتھ ملا کر، آپ انٹرایکٹو اور متحرک ویب صفحات تشکیل دے سکتے ہیں جو صارف کے ان پٹ اور اقدامات کا جواب دیتے ہیں۔ اپنے ویب صفحات کو دلچسپ اور انٹرایکٹو بنانے کے مزید طریقے دریافت کرنے کی مشق جاری رکھیں۔

8.8 ڈویلپنگ اور ڈیبگنگ

ٹیسٹنگ اور ڈیبگنگ ویب ڈویلپمنٹ میں اہم اقدامات ہیں۔ وہ آپ کے کوڈ میں غلطیوں کو تلاش کرنے اور ٹھیک کرنے میں آپ کی مدد کرتے ہیں تاکہ یہ یقینی بنایا جاسکے کہ آپ کے ویب صفحات صحیح طریقے سے کام کرتے ہیں۔

8.8.1 ڈیبگنگ تکنیک

ڈیبگنگ آپ کے کوڈ میں مسائل کو تلاش کرنے اور ٹھیک کرنے کا عمل ہے۔ یہاں کچھ عام ڈیبگنگ تکنیک ہیں:

(i) براؤزر ڈویلپر ٹولز کا استعمال

زیادہ تر ویب براؤزرز میں بلٹ ان ڈویلپر ٹولز ہوتے ہیں جو آپ کو اپنے کوڈ کو ڈی بگ کرنے میں مدد کرتے ہیں۔ مثال کے طور پر، آپ غلطی کے پیغامات دیکھنے کے لیے ”کنسول“ کا استعمال کر سکتے ہیں اور اپنے کوڈ کو روکنے اور اس کے طرز عمل کی جانچ پڑتال کرنے کے لیے بریک پوائنٹس کا استعمال کر سکتے ہیں۔

```
<script>
console.log("This is a debug message");
var x = 10;
console.log("The value of x is: " + x);
</script>
```

(ii) ایرر کے پیغامات پڑھنا

جب کچھ غلط ہو جاتا ہے تو، آپ کا براؤزر عام طور پر ایرر کا پیغام دکھائے گا۔ ان پیغامات کو احتیاط سے پڑھنے سے آپ کو یہ سمجھنے میں مدد مل سکتی ہے کہ کیا غلط ہوا ہے اور اسے کیسے ٹھیک کرنا ہے۔

(iii) اپنا کوڈ چیک کرنا

اپنے کوڈ کو لائن بائی لائن چیک کریں اور کامن ایررز جیسا کہ سیکی کولن، کرلی بریسز، ویری ایبلز کا نام وغیرہ کا نہ ہونا یا غلط ہونا چیک کریں۔

8.8.2 عام مسائل کی نشاندہی اور ان کا حل

ویب ڈویلپمنٹ میں کچھ عام مسائل اور انھیں ٹھیک کرنے کے طریقے:

- 1۔ غیر مکمل لنکس: یقینی بنائیں کہ آپ کے تمام لنکس صحیح یو آر ایل کی طرف اشارہ کرتے ہیں۔ اپنی فائلوں کے راستوں کو ڈبل چیک کریں۔
- 2۔ HTML کا غلط سٹرکچر: اس بات کو یقینی بنائیں کہ آپ کے HTML ٹیگ مناسب طریقے سے نیسٹڈ اور بند ہیں۔
- 3۔ CSS کے مسائل: تصدیق کریں کہ آپ کے CSS سلیکٹرز درست ہیں اور یہ کہ آپ کے اسٹائلز میں کوئی ٹائپنگ کی غلطیاں نہیں ہیں۔

8.8.3 ٹیسٹنگ اور ڈپلوائنگ

اپنے ویب پیج کو تیار کرنے کے بعد، یہ یقینی بنانے کے لیے مختلف براؤزرز اور آلات پر اس کی جانچ کرنا ضروری ہے کہ یہ تمام صارفین کے لیے کام کرتا ہے۔

8.8.3.1 ویب صفحات کی ٹیسٹنگ کے لیے حکمت عملی

- (i) کراس براؤزر ٹیسٹنگ: ویب صفحات مختلف براؤزرز میں مختلف نظر آ سکتے ہیں۔ مستقل مزاجی کو یقینی بنانے کے لیے کروم، فائر فاکس اور ایج جیسے متعدد براؤزرز میں اپنے ویب پیج کی جانچ کریں۔

(ii) رسپانسو ڈیزائن ٹیسٹنگ: اس بات کو یقینی بنائیں کہ آپ کا ویب صفحہ ڈیسک ٹاپ، ٹیبلٹ اور اسمارٹ فونز سمیت تمام آلات پر اچھا نظر آتا ہے۔ آپ کا صفحہ مختلف اسکرین سائز پر کس طرح نظر آتا ہے اس کی جانچ کرنے کے لیے براؤزر کے رسپانسو ڈیزائن موڈ جیسے ٹولز کا استعمال کریں۔

(iii) صارف کی جانچ: اپنے دوستوں یا کنبہ کے ممبروں کو اپنے ویب پیج کا استعمال کرنے اور رائے فراہم کرنے کے لیے کہیں۔ انہیں ایسے مسائل مل سکتے ہیں جو آپ بھول گئے ہیں۔

کیا آپ جانتے ہیں؟

آپ براؤزر کے انسپکٹر ٹول کا استعمال کر کے دیکھ سکتے ہیں کہ آپ کی CSS کیسے عناصر پر لاگو ہوتی ہے۔

خلاصہ

- ویب ڈویلپمنٹ ویب سائٹس اور ویب ایپلی کیشنز بنانے کا عمل ہے۔ اس میں ویب سائٹس کو ڈیزائن، تعمیر اور برقرار رکھنے کے لیے مختلف پروگرامنگ زبانوں اور ٹولز کا استعمال شامل ہے۔
- فرنٹ اینڈ ڈیولپمنٹ اس بات پر توجہ مرکوز کرتا ہے کہ صارفین کسی ویب سائٹ پر کیا دیکھتے ہیں اور اس کے ساتھ بات چیت کرتے ہیں۔
- بیک اینڈ ڈیولپمنٹ ویب سائٹ کے پس پردہ حصے کا انتظام کرتا ہے، جیسے سرورز، ڈیٹابیس اور ایپلی کیشن منطق۔
- HTML سے مراد ہائپر ٹیکسٹ مارک اپ لینگویج ہے۔ یہ معیاری زبان ہے جو ویب صفحات بنانے کے لیے استعمال ہوتی ہے۔
- HTML کو ایک ویب سائٹ کے بلڈنگ بلاکس کے طور پر دیکھا جاسکتا ہے۔
- HTML میں مواد کسی ویب پیج پر اہم معلومات ہے جسے صارفین پڑھتے ہیں اور ان کے ساتھ بات چیت کرتے ہیں۔
- HTML میں ہیڈنگز h1 سے h6 تک ویب پیج پر مواد کی ساخت اور درجہ بندی کی وضاحت کرنے کے لیے استعمال ہوتے ہیں۔
- HTML میں پیراگراف متن کو پڑھنے کے قابل حصوں میں منظم اور الگ کرنے کے لیے استعمال ہوتے ہیں۔
- HTML میں لنکس ایک ویب پیج کو دوسرے سے مربوط کرنے کے لیے استعمال ہوتے ہیں۔ وہ آپ کو ایک ہی صفحے کے مختلف حصوں یا انٹرنیٹ پر دوسرے صفحات پر جانے کے لیے الفاظ یا تصاویر پر کلک کرنے دیتے ہیں۔
- HTML میں کمینٹس کوڈ کے اندر نوٹ یا وضاحتیں شامل کرنے کے لیے استعمال ہوتے ہیں۔
- ویب صفحات کی بصری ظاہری شکل کو بڑھانے اور صارف کے تجربے کو بہتر بنانے کے لیے CSS کے ساتھ اسٹائل کرنا ضروری ہے۔
- جاوا اسکریپٹ ایک پروگرامنگ لینگویج ہے جو ویب سائٹوں کو پُرکشش اور تفریحی بنانے کے لیے استعمال ہوتی ہے۔
- ڈیبلنگ آپ کے کوڈ میں غلطیوں کو تلاش کرنے اور ٹھیک کرنے کا عمل ہے۔

مشق

1۔ درست جواب کا انتخاب کریں:

(i) HTML کا کون سا ٹیگ درست نہیں ہے؟

(الف) <div> (ب) (ج) <head> (د) <footer>

(ii) CSS مخفف ہے:

(الف) Cascading Style Sheet (ب) Computer Style Sheet
(ج) Creative Style Sheet (د) Colorful Style Sheet

(iii) ہائپر لنک بنانے کے لیے ٹیگ استعمال ہوتا ہے:

(الف) <link> (ب) <href> (ج) <nav> (د) <a>

(iv) CSS میں بیک گراؤ تبدیل کرنے کے لیے استعمال ہوتی ہے:

(الف) Color (ب) background-color (ج) bgcolor (د) background

(v) HTML میں ان لائن ٹیگ کے لیے attribute استعمال ہوتا ہے:

(الف) else (ب) style (ج) font (د) value

(vi) CSS رول کے لیے درست syntax ہے:

(الف) selector {property:value;} (ب) selector {property=value;}
(ج) selector: {property:value;} (د) selector: {property=value;}

(vii) جاوا سکرپٹ میں کمینٹس کے لیے مارک اپ استعمال ہوتا ہے:

(الف) /* */ (ب) // (ج) <---> (د) <>

(viii) HTML میں جاوا سکرپٹ کو شامل کرنے کے لیے کوڈ استعمال کرتے ہیں:

(الف) <script src="script.js"></script> (ب) <java src="script.js"></java>
(ج) <js src="script.js"></js> (د) <code src="script.js"></code>

(ix) ان آرڈرڈ لسٹ بنانے کے لیے tag استعمال ہوتا ہے:

(الف) (ب) (ج) (د) <list>

(x) HTML میں افقی لائن لگانے کے لیے tag استعمال ہوتا ہے: +

(الف)
 (ب) <hr> (ج) <line> (د) <hline>

2۔ مختصر جواب دیں۔

- (i) HTML میں <head> ٹیگ کا مقصد کیا ہے؟
- (ii) HTML میں آرڈر شدہ فہرست اور غیر آرڈر شدہ فہرست کے درمیان فرق کی وضاحت کریں۔
- (iii) آپ CSS میں کمنٹس کیسے شامل کرتے ہیں؟
- (iv) HTML دستاویز پر CSS لاگو کرنے کے مختلف طریقے کیا ہیں؟
- (v) آپ HTML فائل میں جاوا اسکریپٹ کو کیسے شامل کر سکتے ہیں؟
- (vi) HTML میں ہائپر لنک بنانے کے لیے غوی کی وضاحت کریں۔
- (vii) HTML میں <div> کا کام کیا ہے؟
- (viii) آپ ایکسٹرنل CSS فائل کو HTML دستاویز سے کیسے جوڑتے ہیں؟
- (ix) HTML میں <table> ٹیگ کا کیا استعمال ہے؟
- (x) CSS میں باکس ماڈل کی وضاحت کریں۔

3۔ تفصیلی سوالات۔

- (i) ویب ڈویلپمنٹ کے تناظر میں HTML، CSS، اور جاوا اسکریپٹ کے مابین بنیادی اختلافات پر تبادلہ خیال کریں۔
- (ii) ویب ڈویلپمنٹ کے لیے ترقیاتی ماحول قائم کرنے کے عمل کی وضاحت کریں۔ ضروری سافٹ ویئر اور ٹولز کو الگ کر کے۔
- (iii) ایک بنیادی HTML صفحہ بنائیں جس میں ہیڈر، ایک پیراگراف، ایک تصویر، اور ایک ہائپر لنک شامل ہے۔
- (iv) آپ CSS کا استعمال کرتے ہوئے ٹیبل کو کس طرح اسٹائل کرتے ہیں؟ ایک نمونہ جدول بنائیں اور اس پر اسٹائل لاگو کریں۔
- (v) مختلف CSS سلیکٹرز کی وضاحت کریں اور ہر ایک کی مثالیں فراہم کریں۔
- (vi) مثالوں اور وضاحتوں کی مدد سے CSS کا استعمال کرتے ہوئے ایک ذمہ دار ویب صفحہ بنانے کے عمل کی وضاحت کریں۔
- (vii) ایک جاوا اسکریپٹ فنکشن لکھیں جو کسی بٹن پر کلک کرنے پر ویب صفحے کے پس منظر کا رنگ تبدیل کرتا ہے۔ مکمل کوڈ فراہم کریں اور وضاحت کریں کہ یہ کیسے کام کرتا ہے۔
- (viii) آپ CSS کا استعمال کرتے ہوئے اپنی میٹشز اور ٹرانزیشنز کو کیسے شامل کرتے ہیں؟ مثالیں فراہم کریں اور اس میں شامل خصوصیات کی وضاحت کریں۔