

# HTML

HTML (HyperText Markup Language) veb-səhifəyə məzmun vermək üçün istifadə olunur və veb-brauzerlərə həmin məzmunu necə strukturlaşdırmaq barədə təlimat verir.

## Elementlər

HTML elementinin məzmunu elementin açılış və bağlanma teqləri arasındakı məlumatdır.

### <li> Siyahı Element

<li> siyahı elementi iki cür siyahının daxilində istifadə olunur.

- Sıralı siyahı <ol>
- Sıralanmamış siyahı <ul>

```
<ol>
  <li>Head east on Prince St</li>
  <li>Turn left on Elizabeth</li>
</ol>

<ul>
  <li>Cookies</li>
  <li>Milk</li>
</ul>
```

### <ol> Sıralı Siyahı Elementi

<ol> sıralı siyahı elementi ardıcıl qaydada elementlərin siyahısını yaradır. Hər bir siyahı elementi standart olaraq nömrələnmiş görünür.

```
<ol>
  <li>Preheat oven to 325 F 🧑🍳</li>
  <li>Drop cookie dough 🍪</li>
  <li>Bake for 15 min ⌚</li>
</ol>
```



## <ul> Sıralanmamış Siyahı Elementi

<ul> sıralanmamış siyahı elementi heç bir xüsusi sırada olmayan elementlərin siyahısını yaratmaq üçün istifadə olunur.

Hər bir fərdi siyahı elementinin standart olaraq bir marker nöqtəsi olacaq.

```
<ul>
  <li>Play more music 🎵 </li>
  <li>Read more books 📖 </li>
</ul>
```

## <div> Div Elementi

<div> <div> elementi HTML sənədini bölmələrə ayıran konteyner kimi istifadə olunur və "bölmə" üçün qısadır.

<div> elementlərində başlıqlar, paraqraflar, keçidlər, şəkillər və s. kimi məzmunlarla doldurular bilər.

```
<div>
  <h1>A section of grouped elements</h1>
  <p>Here's some text for the section</p>
</div>
<div>
  <h1>Second section of grouped elements</h1>
  <p>Here's some text</p>
</div>
```

## Bağlanma Teqi

HTML elementinin sonunu bildirmək üçün HTML bağlanma teqi istifadə olunur. Bağlama teqinin sintaksisi sol bucaq mötərizəsidir < ardınca irəli slash / sonra element adı və bağlanmaq üçün sağ bucaq mötərizəsi >.

```
<body>
...
</body>
```



digiuth

## Attribute Name and Values

HTML atributları aşağıdakı sintaksisdən istifadə edən ad və dəyərdən ibarətdir:

`name="value"` və elementin davranışını konfigurasiya etmək və ya dəyişmək üçün HTML elementinin açılış teqinə əlavə edilə bilər.

```
<elementName name="value"></elementName>
```

## <img> Image Elementi

HTML təsviri `<img>` elementləri şəkilləri sənədlərə yerləşdirir. `src` atributunda şəkil URL-i var və məcburidir. `<img>` boş elementdir, yəni onun bağlanma teqi olmamalıdır.

```

```

## <h1>-<h6> Başlıq Elementləri

HTML altı müxtəlif səviyyəli başlıq elementlərindən istifadə edə bilər. Başlıq elementləri ən yüksək səviyyədən `<h1>` ən aşağı səviyyədən `<h6>` sıralanır.

```
<h1>Breaking News</h1>
<h2>This is the 1st subheading</h2>
<h3>This is the 2nd subheading</h3>
...
<h6>This is the 5th subheading</h6>
```



## <p> Paraqraf Elementi

<p> paraqraf elementi mətn blokunu göstərir.

```
<p>This is a block of text! Lorem ipsum dolor sit amet,  
consectetur adipisicing elit.</p>
```

## Unikal ID Atributları

HTML-də onları fərqləndirmək üçün müxtəlif elementlərə xüsusi və unikal <id> atributları təyin edilə bilər.

```
<h1 id="A1">Hello World</h1>
```

Ehtiyac yarandıqda, <id> dəyəri CSS və JavaScript tərəfindən yalnız həmin element və element üzrə xüsusi təlimatları manipulyasiya etmək, formatlaşdırmaq və yerinə yetirmək üçün çağırıla bilər. Etibarlı <id> atributları hərflə başlamalı və yalnız hərflərdən (a-z), rəqəmlərdən (0-9), tire (-), alt xətt (\_) və nöqtələrdən (.) ibarət olmalıdır.

## alt Atributu

<img> elementi alt atribut vasitəsilə alternativ mətnə malik ola bilər. Əgər şəkil səhv URL səbəbindən göstərilə bilmirsə, şəkil formatı brauzer tərəfindən dəstəklənmirsə, şəkilin nümayişi bloklanıbsa və ya şəkil URL-dən alınmayıbsa, alternativ mətn göstəriləcək.

```

```

Ekrandan oxumaq üçün proqram istifadə edilərsə və veb-səhifədə təsvirin məzmunu üçün mətn deskriptoru təqdim etməklə görmə qabiliyyəti zəif olan istifadəçilərə dəstək olarsa, mətn ucadan oxunacaq.

## <body> Body Elementi

<body> elementi HTML sənədinin məzmununu təmsil edir. <body> teqlərinin içindəki məzmun veb brauzerlərdə göstərilir.

**Qeyd:** Sənəddə yalnız bir <body> elementi ola bilər.

## <a> Anchor Elementi

<a> anker elementi HTML sənədində hiperlinklər yaratmaq üçün istifadə olunur.

Hiperlinklər digər veb səhifələrə, eyni serverdəki fayllara, eyni səhifədəki yerə və ya hiperlink istinad atributu, href vasitəsilə hər hansı digər URL-yə işarə edə bilər. href lövbər elementinin göstərdiyi yeri müəyyənləşdirir.

```
<!-- Creating text links -->
<a href="http://www.codecademy.com">Visit this site</a>

<!-- Creating image links -->
<a href="http://www.codecademy.com">
  Click this image
</a>
```

## <head> Head Elementi

<head> elementi səhifənin özündə göstərilməyən HTML səhifəsi haqqında ümumi məlumatları ehtiva edir.

Bu məlumat metadata adlanır və HTML sənədinin başlığı və üslub cədvəllərinə keçidlər kimi şeyləri ehtiva edir.

```
<!DOCTYPE html>
<html>
  <head>
    <!-- Metadata is contained in this element-->
  </head>
</html>
```



## <target> Target Atributu

`<a>` lövbər elementindəki hədəf atribut hiperlinkin harada açılmalı olduğunu müəyyən edir.

`"_blank"` hədəf dəyəri brauzerə hiperlinki müasir brauzerlərdə yeni tabda və ya köhnə brauzerlərdə yeni pəncərədə açmağı və ya brauzerdə hiperlinkləri yeni pəncərədə açmaq üçün parametrləri dəyişdirdiyini bildirəcək.

```
<a href="https://www.google.com" target="_blank">This anchor element links to google and will open in a new tab or window.</a>
```

## Səhifənin Fərqli Hissəsinə keçid #

`<a>` elementi eyni HTML sənədinin müxtəlif hissələrinə hiperlinklər yarada bilər və `href` atributundan istifadə edərək istədiyiniz yeri göstərə bilər, ardınca `#` ilə əlaqələndiriləcək elementin `id`-si qeyd edilməlidir.

```
<div>
  <p id="id-of-element-to-link-to">A different part of the page!</p>
</div>

<a href="#id-of-element-to-link-to">Take me to a different part of the page</a>
```

## Şərhlər

HTML-də şərhlər açılış `<!--` və bağlanış `-->` arasında əlavə edilə bilər. Şərhlərin içindəki məzmun brauzerlər tərəfindən göstərilməyəcək və adətən kodun bir hissəsini təsvir etmək və ya digər təfərrüatları təmin etmək üçün istifadə olunur.

Şərhlər tək və ya bir neçə sətirdən ibarət ola bilər.

```
<!-- Main site content -->
<div>Content</div>

<!--
  Comments can be
  multiple lines long.
-->
```

## Boşluq

Blok səviyyəli elementlər arasında HTML sənədinə əlavə edilən sətir fasilələri kimi boşluqlar ümumiyyətlə brauzer tərəfindən nəzərə alınmayacaq və göstərilən HTML səhifəsində boşluqları artırmaq üçün əlavə edilməyəcək.

Əksinə, HTML sənədinin özünün təşkili və daha asan oxunması üçün boşluq əlavə edilir.

```
<p>Test paragraph</p>
```

```
<!-- The whitespace created by this line, and  
above/below this line is ignored by the browser-->
```

```
<p>Another test paragraph, this will sit right under the  
first paragraph, no extra space between.</p>
```

## <title> Title Elementi

<title> elementi HTML sənədinin başlığını təyin edən mətdən ibarətdir.

Başlıq brauzerin başlıq çubuğunda və ya HTML səhifəsinin göstərildiyi tabda göstərilir. <title> elementi yalnız sənədin <head> elementində ola bilər.

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title>Title of the HTML page</title>  
  </head>  
</html>
```

## Fayl Yolu

HTML-də URL yolları tam URL kimi mütləq yollar ola bilər, məsələn:

<https://developer.mozilla.org/en-US/docs/Learn> və ya eyni qovluqda və ya yerli fayla keçid verən nisbi fayl yolu eyni serverdə, məsələn: [./style.css](#).

Nisbi fayl yolları [./](#) ilə başlayır və ardınca yerli fayla gedən yol gəlir. [./](#) brauzerə cari qovluqdan fayl yolunu axtarmağı bildirir.

```
<a href="https://developer.mozilla.org/en-US/docs/Web">The URL for this anchor element is an  
absolute file path.</a>  
  
<a href="./about.html">The URL for this anchor element  
is a relative file path.</a>
```



digiuth



## Sənəd Növü Bəyannaməsi

`<!DOCTYPE html>` sənəd növü bəyannaməsi HTML sənədinin birinci sətiri kimi tələb olunur. Doctype bəyannaməsi brauzerə hansı növ sənədin gözlənilməsi və HTML-nin hansı versiyasının istifadə olunduğu barədə təlimatdır, bu halda bu HTML5-dir.

```
<!DOCTYPE html>
```

## `<form>` Elementi

HTML `<form>` elementi məlumat toplamaq və xarici mənbəyə göndərmək üçün istifadə olunur.

`<form>` müxtəlif giriş elementlərini ehtiva edə bilər. İstifadəçi formanı təqdim etdikdə, bu giriş elementlərindəki məlumat formanın fəaliyyət atributunda adlandırılan mənbəyə ötürülür.

```
<form method="post" action="http://server1">  
  Enter your name:  
  <input type="text" name="fname">  
  <br/>  
  Enter your age:  
  <input type="text" name="age">  
  <br/>  
  <input type="submit" value="Submit">  
</form>
```

## `<input>`: Nömrə Növü

HTML daxiletmə elementləri növ sayı ola bilər. Bu giriş sahələri istifadəçiyə sahənin içərisinə yalnız rəqəmlər və bir neçə xüsusi simvol daxil etməyə imkan verir.

```
<input type="number" name="balance" />
```

Nümunə kod bloku nömrə növü və balansın adı olan bir girişi göstərir. Daxiletmə sahəsi formanın bir hissəsi olduqda, forma təqdim edildikdən sonra ad: dəyər formatı ilə açar-dəyər cütü alacaq.



## <input> Element

HTML `<input>` elementi mətn sahələri, onay qutuları, düymələr və s. daxil olmaqla veb-səhifədə müxtəlif daxiletmə sahələrini göstərmək üçün istifadə olunur.

`<input>` elementi onun səhifəyə necə göstərildiyini müəyyən edən tip atributuna malikdir.

Nümunə kod bloku veb-səhifədə mətn daxiletmə sahəsi və qeyd qutusu daxiletmə sahəsi yaradacaq.

```
<label for="fname">First name:</label>
<input type="text" name="fname" id="fname"><br>

<input type="checkbox" name="vehicle" value="Bike">
I own a bike
```

## <input>: Aralıq Növü

Slayder HTML daxiletmə elementində `type="range"` atributundan istifadə etməklə yaradıla bilər. Aralıq slayderi minimum və maksimum dəyər arasında seçici rolunu oynayacaq. Bu dəyərlər müvafiq olaraq min və max atributlarından istifadə etməklə təyin edilir. Slayder addım atributundan istifadə edərək müxtəlif addımlarda və ya artımlarda hərəkət etmək üçün tənzimlənə bilər.

Aralıq kaydırıcısı nisbi mövqenin vacib olduğu, lakin dəqiq dəyərin o qədər də vacib olmadığı 2 dəyər arasında tənzimləmək üçün daha çox vizual qadçet kimi çıxış etmək üçün nəzərdə tutulub. Buna misal olaraq proqramın səs səviyyəsini tənzimləmək olar.

```
<input type="range" name="movie-rating" min="0" max="10"
step="0.1">
```



## <label> Elementi

HTML `<label>` elementi atribut üçün `<input>`'s `id` atributunun və `<label>`'s atributunun uyğun qiymətlərinə əsaslanan xüsusi `<input>` üçün identifikasiyanı təmin edir.

```
<label for="password ">Password:</label>  
<input type="text" id="password" name="password">
```

Varsayılan olaraq, `<label>` üzərinə klikləməklə əlaqədar `<input>` sahəsi diqqət mərkəzində olacaq.

Nümunə kodu, yanında "Parol:" etiket mətni olan mətn daxiletmə sahəsi yaradacaq. Səhifədə "Parol: " üzərinə klikləməklə, əlaqədar `<input>` sahəsi diqqət mərkəzində olacaq.

---

**Bu məlumatlar Digiuth layihəsi çərçivəsində hazırlanmışdır. Layihə Çexiya, Macarıstan, Polşa və Slovakiya hökumətləri tərəfindən Beynəlxalq Vişeqrad Fondunun Vişeqrad Qrantları vasitəsilə birgə maliyyələşdirilir. Fondun missiyası Mərkəzi Avropada davamlı regional əməkdaşlıq üçün ideyaları irəli sürməkdir.**



# digiuth

## <link> Link Element

`<link>` elementi HTML sənədlərini CSS faylları kimi xarici resurslarla əlaqələndirmək üçün istifadə olunur.

Tez-tez istifadə edir:

Xarici resurs üçün URL-i müəyyən etmək üçün `href` atributu.

`rel` atributu əlaqəli sənədin cari sənədlə əlaqəsini təyin etmək üçün.

Əlaqələnən məzmunun növünü müəyyən etmək üçün `type` atributu.

```
<!-- How to link an external stylesheet with href, rel,
and type attributes -->
```

```
<link href="./path/to/stylesheet/style.css"
rel="stylesheet" type="text/css">
```

## CSS'in məqsədi

CSS və ya Cascading Style Sheets, HTML elementlərinin necə görünəcəyini fərdiləşdirən HTML ilə birlikdə istifadə olunan bir dildir. CSS üslubları müəyyən edə və vərəqin tərtibatını və dizaynını dəyişə bilər.

## Ayrı Fayllarda CSS yazmaq

CSS kodu HTML kodundan ayrı saxlamaq üçün öz fayllarında yazıla bilər. CSS faylları üçün genişlənmə `.css`-dir.

Bunlar `<head>` bölməsindəki `<link>` teqindən istifadə etməklə HTML faylı ilə əlaqələndirilə bilər.

```
<head>
  <link href="style.css" type="text/css"
rel="stylesheet">
</head>
```

## HTML faylında CSS yazmaq

CSS kodu HTML faylında kodu `<style>` teqlərinə əlavə etməklə yazıla bilər. `<style>` teqləri ilə əhatə olunmuş kod CSS sintaksisi kimi şərh olunacaq.

```
<head>
  <style>
    h1 {
      color: blue;
    }
  </style>
</head>
```

## Inline Styles

CSS üslubları elementin açılış etiketindəki `<style>` atributundan istifadə etməklə birbaşa HTML elementlərinə əlavə edilə bilər. Hər üslub bəyannaməsi nöqtəli vergüllə bitir. Bu şəkildə əlavə edilən üslublar daxili üslublar kimi tanınır.

```
<h2 style="text-align: center;">Centered text</h2>

<p style="color: blue; font-size: 18px;">Blue, 18-point
text</p>
```

## Class və ID Selektorları

CSS sinifləri təkrar istifadə edilə bilər və bir çox elementə tətbiq oluna bilər. Sinif seçiciləri `.` ilə işarələnir, sonra sinif adı. CSS ID seçiciləri unikal olmalıdır və yalnız bir elementin üslubunu yaratmaq üçün istifadə edilməlidir. ID seçiciləri `#` hash işarəsi və ardınca id adı ilə işarələnir.

```
/* Selects all elements with class="column" */
.column {
}

/* Selects element with id="first-item" */
#first-item {
}
```

## CSS Selektorlarının Qrup Halı

Vergüllə ayrılmış siyahıdan istifadə edərək çoxlu seçiciləri eyni CSS qaydasına uyğunlaşdırın. Bu nümunədə həm `h1`, həm də `h2` üçün mətn qırmızı rəngə təyin edilib.

```
h1, h2 {  
    color: red;  
}
```

## CSS klass seçiciləri

CSS sinif seçicisi elementləri sinif atributunun məzmununa əsasən uyğunlaşdırır. Sinif atributunun dəyəri kimi təqvim xanası olan elementləri seçmək üçün `.` qabaqcadan yazmaq lazımdır.

```
.calendar-cell {  
    color: #fff;  
}
```

## CSS ID seçiciləri

CSS ID seçicisi `ID` atributunun məzmununa əsasən elementlərə uyğun gəlir. `ID` atributunun dəyərləri bütün DOM-da unikal olmalıdır. `ID` atributunun dəyəri kimi iş başlığına malik elementi seçmək üçün başına `#` işarəsi qoyulmalıdır.

```
#job-title {  
    font-weight: bold;  
}
```

## `console.log()`

`console.log()` metodu mesajları konsola daxil etmək və ya çap etmək üçün istifadə olunur. O, həmçinin obyektləri və digər məlumatları çap etmək üçün istifadə edilə bilər.

```
console.log('Hi there!');  
// Prints: Hi there!
```

## Rəqəmlər

Nömrələr primitiv məlumat növüdür. Onlara bütün tam ədədlər və kəsr ədədlər daxildir.

```
let amount = 6;  
let price = 4.99;
```

## Booleans

Booleanlar primitiv məlumat növüdür. Onlar doğru və ya yanlış ola bilər. (`true` or `false`)

```
let lateToWork = true;
```

## Strings

Stringlər primitiv məlumat növüdür.

Bunlar tək dırnaq və ya qoşa dırnaq işarələri ilə əhatə olunmuş hər hansı bir simvol qrupudur (hərflər, boşluqlar, rəqəmlər və ya simvollar).

```
let single = 'Wheres my bandit hat?';  
let double = "Wheres my bandit hat?";
```

# Funksiyalar

Funksiyalar JavaScript-də əsas tikinti bloklarından biridir.

Funksiya bir tapşırığı yerinə yetirmək və ya dəyəri hesablamaq üçün təkrar istifadə edilə bilən ifadələr toplusudur.

Funksiyalara bir və ya bir neçə dəyər ötürülə bilər və onların icrasının sonunda bir dəyər qaytara bilər.

Funksiyadan istifadə etmək üçün onu çağırmaq istədiyiniz sahədə müəyyən etməlisiniz.

Təqdim olunan nümunə kodda 2 dəyər qəbul edən və həmin ədədlərin cəmini qaytaran funksiya var.

```
// Defining the function:
function sum(num1, num2) {
    return num1 + num2;
}

// Calling the function:
sum(3, 6); // 9
```

## if Statement

`if` ifadəsi mötərizə dəsti ilə ifadəni qəbul edir:

İfadə doğru qiymətə qiymətləndirilsə, onun kod gövdəsindəki kod yerinə yetirilir.

İfadə saxta qiymətə qiymətləndirilsə, onun kod gövdəsi icra olunmayacaq.

```
const isMailSent = true;

if (isMailSent) {
    console.log('Mail sent to recipient');
}
```

## switch Statement

`<switch>` ifadəsini çoxlu hal bəndlərinə qarşı yoxlamaq üçün bir vasitə təmin edir.

Əgər iş uyğun gəlsə, həmin bəndin daxilindəki kod icra olunur.

Case bəndi `<break>` açar sözü ilə tamamlanmalıdır.

Heç bir halda uyğun gəlmirsə, lakin `<default>` bənd daxil edilibsə, `<default>` daxilindəki kod icra ediləcək.

**Qeyd:** İşin blokundan `<break>` buraxılıbsa, keçid ifadəsi `<break>` ilə qarşılaşana və ya axın pozulana qədər iş qiymətlərini yoxlamağa davam edəcək.

```
const food = 'salad';

switch (food) {
  case 'oyster':
    console.log('The taste of the sea 🍤');
    break;
  case 'pizza':
    console.log('A delicious pie 🍕');
    break;
  default:
    console.log('Enjoy your meal');
}

// Prints: Enjoy your meal
```