

# به نام خدا

برنامه نویسی مبتنی بر وب



فصل اول

HTML 5

# زبان های نشانه گذاری

- زبان های نشانه گذاری (Markup Language) نوعی از زبان های **توصیفی** هستند، که متن و اطلاعاتی در مورد نحوه نحوه تفسیر متن را با هم ادغام می کنند.

✓ **SGML** (Standard Generalization Markup Language)

✓ **XML** (Extensible Markup Language)

✓ **HTML** (Hyper Text Markup Language)



# HTML چیست؟

- **HTML** مخفف کلمات **H**yper **T**ext **M**arkup **L**anguage به معنای زبان نشانه گذاری ابر متن است. در این زبان مجموعه ای **برچسب** ها (Tag) مشخص می کنند محتوای صفحه در چه قالبی نمایش داده شود.



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
  <meta name="TITLE" content="...">
  <meta name="KEYWORDS" content="...">
  <meta name="DESCRIPTION" content="...">
  <link rel="stylesheet" href="...">
  <script language="javascript" src="...">
</head>
<body bgcolor="#ffffff">
```

# ورژن های HTML

Version	Year
HTML	1991
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML5	2014

# تاریخچه HTML

Year	Version
1989	Tim Berners-Lee invented www
1991	Tim Berners-Lee invented HTML
1993	Dave Raggett drafted HTML+
1995	HTML Working Group defined HTML 2.0
1997	W3C Recommendation: HTML 3.2
1999	W3C Recommendation: HTML 4.01
2000	W3C Recommendation: XHTML 1.0
2008	WHATWG HTML5 First Public Draft
2012	WHATWG HTML5 Living Standard
2014	W3C Recommendation: HTML5
2016	W3C Candidate Recommendation: HTML 5.1

# HTML 5.2

- Release Candidate
- Proposed Recommendation
- W3C Recommendation

## HTML 5.2

W3C Recommendation, 14 December 2017



# HTML 5.2

## Abstract

- This specification defines the 5th major version, second minor revision of the core language of the World Wide Web: the Hypertext Markup Language (HTML). In this version, new features continue to be introduced to help Web application authors, new elements continue to be introduced based on research into prevailing authoring practices, and special attention continues to be given to defining clear conformance criteria for user agents in an effort to improve interoperability.

این نسخه ی خاص، پنجمین نسخه ی اصلی و دومین بازبینی جزئی زبان پایه گستره وب جهانی، HTML، است. در این نسخه، ویژگی های جدیدی برای کمک به برنامه نویس ها اضافه شد. علاوه بر این، عنصرهای جدیدی هم بر اساس تحقیق و بررسی در تجربه های رایج برنامه نویس ها، معرفی شدند.

# Timeline

## Timeline [\[ edit \]](#)

The combined timelines for HTML 5.0, HTML 5.1 and HTML 5.2:

	2012	2013	2014	2015	2016	2017	2018
HTML 5.0	Candidate Rec	Call for Review	Recommendation				
HTML 5.1	1st Working Draft		Last Call	Candidate Recommendation	Recommendation		
HTML 5.2 <sup>[33]</sup>				1st Working Draft		Candidate Recommendation	Recommendation
HTML 5.3 <sup>[34]</sup>						Editor's Draft	



# HTML 5

- HTML 5 آخرین نسخه از این زبان نشانه گذاری است که **حاوی امکانات جدیدی** است، برخی از امکاناتی که در نسخه قبلی موجود بود در این نسخه بهبود یافته و همچنین **API های اسکریپت نویسی** جدیدی نیز به آن اضافه شده است. علاوه بر آن، در طراحی این نسخه اصول پایه ای مد نظر قرار گرفته تا اطمینان حاصل شود که این نسخه بر روی تمامی **پلت فرم ها** به خوبی کار خواهد کرد و خطاهای برنامه نویسی را به خوبی و بدون مشکل مدیریت می کند.

# HTML 5

- مستندات HTML 5 توسط دو گروه مختلف تدوین می شود و در دو نسخه ارائه می شود.

## W3C ❖

✓ کنسرسیوم جهانی وب

✓ مستندات تهیه شده توسط این گروه را می توانید در آدرس <http://dev.w3.org/html5/spec/> بیابید.

✓ 28 October 2014 انتشار یافت.

## WHATWG ❖

✓ Web Hypertext Application Technology Working Group

✓ کارگروه فناوری های ابر متن تحت وب

✓ توسط افرادی از شرکت های Mozilla, Apple, Opera تشکیل شده است.

✓ مستندات تهیه شده توسط این گروه را می توانید در آدرس <http://www.whatwg.org/html/> بیابید.

✓ به این سند، استانداردهای زنده هم گفته می شود، زیرا به طور دائم در حال توسعه و تغییر است.

✓ در سال ۲۰۱۲ منتشر شده و به طور پیوسته ادامه دارد.

# تفاوت بین مستندات WHATWG و W3C

- به طور خلاصه مستندات WHATWG کمی محاوره ای تر و کاربردی تر است و البته بعضی می گویند کمی آینده نگرانه تر است. ولی در مجموع این دو سند بسیار شبیه به هم می باشند. و می توان هر کدام از این دو سند را به عنوان منبعی برای یادگیری عناصر HTML5 و فناوری های مرتبط با آن به کار گرفت. درست است که سیاست های پشت پرده وجود دارد که ما به عنوان طراح و برنامه نویس، کنترلی بر آنها نداریم ولی وجود این دو سند متفاوت نمی تواند باعث نگرانی شود.

# WHATWG & W3C



<https://whatwg.org/>

The **Web Hypertext Application Technology Working Group** is a community of people interested in evolving HTML and related technologies.

→ [Read the HTML Living Standard](#)



<https://www.w3.org/>

The **World Wide Web Consortium** is the main international standards organization for the World Wide Web.

# HTML Tags

- برای نشان دادن هر المان در صفحه از **تگ** استفاده می شود. هر تگ دارای قسمت شروع (**opening tag**) و قسمت پایانی (**closing tag**) است.

**<tagname>** محتوای برچسپ **</tagname>**

# HTML Tags

- تگ ها را می توان به دو گروه تقسیم کرد:

```
<h1 style="font-size:60px;">Heading 1</h1>
```

---

Container Tag ❖

✓ تگ ها به صورت جفت (باز و بسته) می باشند.

Standalone Tag ❖

✓ تگ هایی که بصورت تکی بکار می روند.

```

```

# ساختار سند HTML

```
<html>
```

```
<head>
```

```
<title>Page title</title>
```

```
</head>
```

```
<body>
```

```
<h1>This is a heading</h1>
```

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

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

```
</body>
```

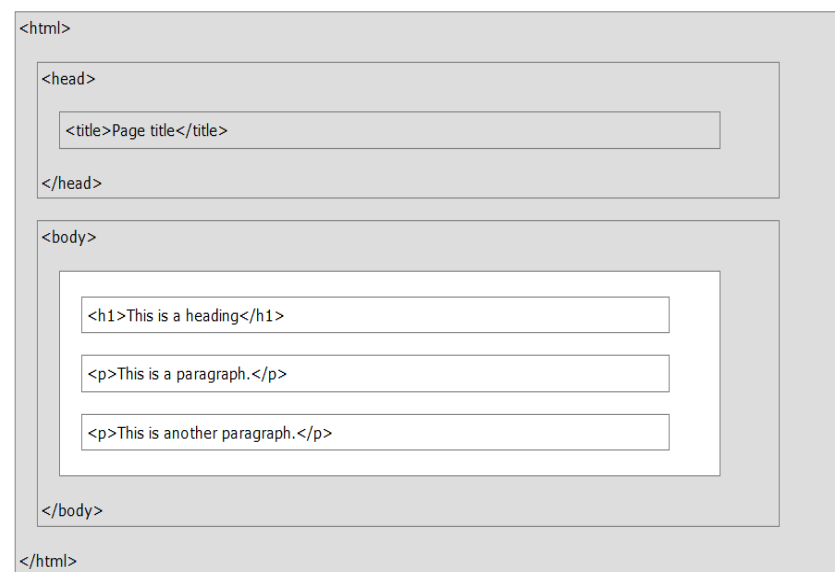
```
</html>
```

# ساختار سند HTML

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

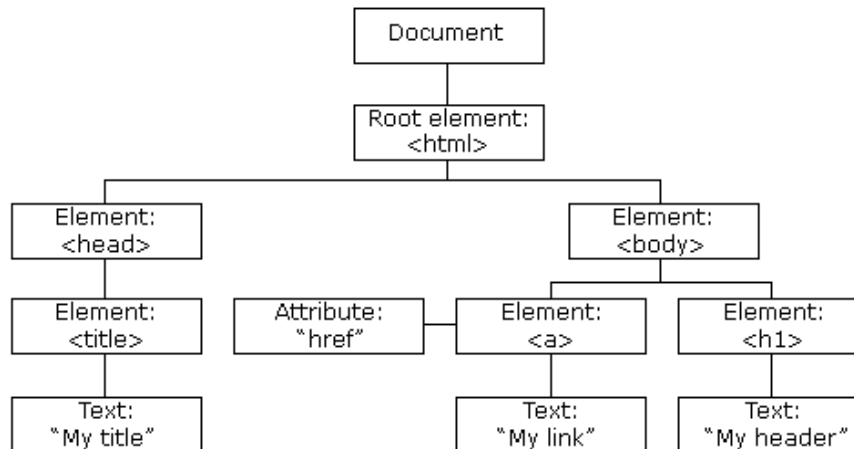
</body>
</html>
```



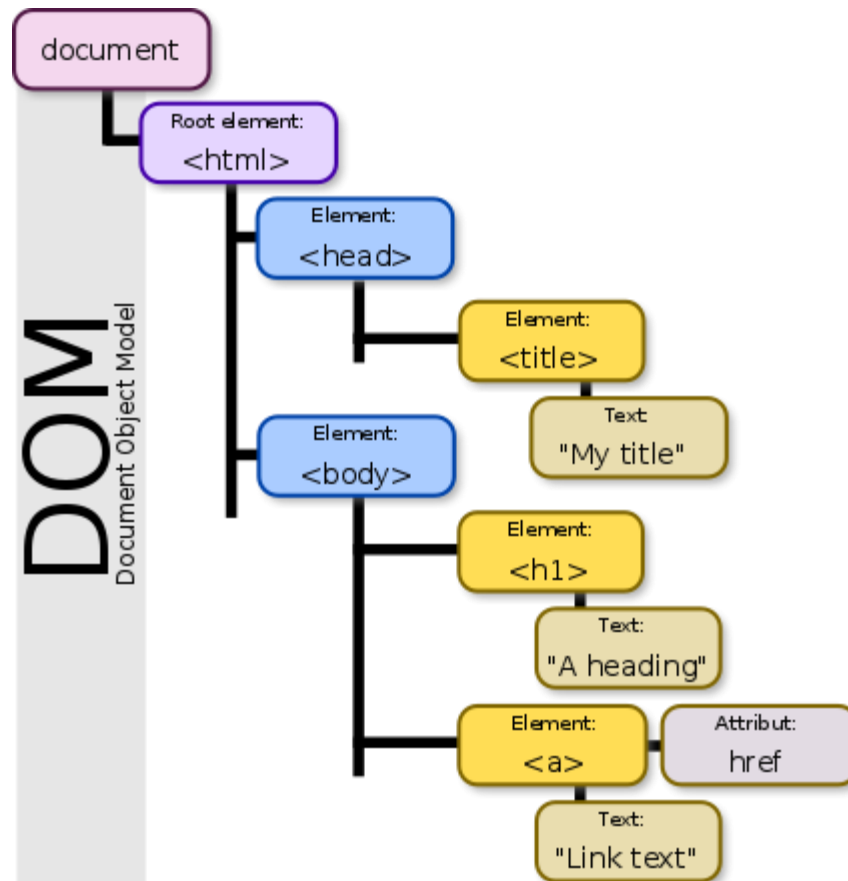


# DOM (Document Object Model)

- زمانی که یک صفحه ی وب بارگذاری می شود، مرورگر از آن صفحه، یک مدل DOM ایجاد می کند. مدل DOM یک ساختار درختی از اشیاء موجود در صفحه HTML می سازد. این مدل ویژگی ها و متدهایی را برای دستیابی به نود ها از طریق جاوا اسکریپت فراهم می کند. در واقع JavaScript با استفاده از مدل DOM، می تواند به تمام عناصر یک صفحه HTML دسترسی داشته باشد.



# (Document Object Model) DOM



## تگ <!DOCTYPE>

- این تگ نوع سند html را مشخص می کند و به مرورگر می گوید که کدها با چه ورژنی از html نوشته شده است. نحوه اعلان نوع سند در html 5 به صورت زیر است.

```
<!DOCTYPE html>
```

# ابزارهای نوشتن کدهای HTML

## • ویرایشگرهای متنی (Text Editor)

✓ در این نرم افزار ها کدهای html به صورت دستی نوشته می شود.

✓ این ویرایشگرها حجم کمی اشغال می کنند.

✓ امکاناتی چون AutoComplete را ندارند.

✓ Notepad, Notepad++, Sublime

## • نرم افزارهای WYSIWIG

✓ می توان از طریق واسط های گرافیکی کدهای html را ایجاد کرد.

✓ NetBeans, PHP Storm, Adobe DreamWeaver

# نوشتن تگ ها به صورت حروف کوچک

- HTML نسبت به حروف کوچک و بزرگ حساس نیست. اما طبق نظر W3C توصیه می شود در هنگام نوشتن تگ ها از حروف کوچک استفاده شود.

# Use Lower Case Element Names

- We recommend using lowercase element names because:
  - ✓ Mixing uppercase and lowercase names is **bad**
  - ✓ Developers normally use lowercase names (as in XHTML)
  - ✓ Lowercase look **cleaner**
  - ✓ Lowercase are easier to write

Bad:

```
<SECTION>
  <p>This is a paragraph.</p>
</SECTION>
```

Very Bad:

```
<Section>
  <p>This is a paragraph.</p>
</SECTION>
```

Good:

```
<section>
  <p>This is a paragraph.</p>
</section>
```

## عناصر Empty

- بعضی عناصر در html محتوایی ندارند. این عناصر تگ پایانی را ندارند. و در تگ ابتدایی بسته می شوند. مانند تگ `<br />` که برای شکستن خط استفاده می شود.

# تگ های تیتر Headings

- این تگ ها دارای شش سطح بوده که باعث درشت تر دیده شدن تیترها شده و استفاده بهینه از مفهوم آنها باعث می شود میزان **SEO** افزایش یابد.

```
<!DOCTYPE html>  
<html>  
<body>
```

```
<h1>This is heading 1</h1>  
<h2>This is heading 2</h2>  
<h3>This is heading 3</h3>  
<h4>This is heading 4</h4>  
<h5>This is heading 5</h5>  
<h6>This is heading 6</h6>
```

```
</body>  
</html>
```

**This is heading 1**

**This is heading 2**

**This is heading 3**

**This is heading 4**

**This is heading 5**

**This is heading 6**



code1.txt



# ایجاد پاراگراف Paragraphs

- برای ایجاد پاراگراف از تگ `<p>` استفاده می شود. این تگ دارای سطح بلاک می باشد که باعث می شود پاراگراف ها زیر هم قرار بگیرند.

```
<!DOCTYPE html>  
<html>  
<body>
```

This is a paragraph.

```
<p>This is a paragraph.</p>  
<p>This is another paragraph.</p>
```

This is another paragraph.

```
</body>  
</html>
```



code2.txt

# ایجاد پاراگراف Paragraphs

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p>
```

This paragraph  
contains a lot of lines  
in the source code,  
but the browser  
ignores it.

```
</p>
```

```
<p>
```

This paragraph  
contains a lot of spaces  
in the source code,  
but the browser  
ignores it.

```
</p>
```

```
<p>
```

The number of lines in a paragraph depends on the size of the browser window.  
If you resize the browser window, the number of lines in this paragraph will  
change.

```
</p>
```

```
</body>
```

This paragraph contains a lot of lines in the source code, but the browser ignores it.

This paragraph contains a lot of spaces in the source code, but the browser ignores it.

The number of lines in a paragraph depends on the size of the browser window. If you resize the  
browser window, the number of lines in this paragraph will change.



code3.txt

# فرمت بندی متن (Text Formatting)

- `<b>` - Bold text
- `<strong>` - Important text
- `<i>` - Italic text
- `<em>` - Emphasized text
- `<mark>` - Marked text
- `<small>` - Small text
- `<del>` - Deleted text
- `<ins>` - Inserted text
- `<sub>` - Subscript text
- `<sup>` - Superscript text

# فرمت بندی متن (Text Formatting)

```
<!DOCTYPE html>
<html>
<body>

<p>This text is normal.</p>
<p><b>This text is bold.</b></p>
<strong>This text is strong</strong>
<br/>
<i>This text is italic</i>
<br/>
<em>This text is emphasized</em>
<br/>
<small>This text is Small</small>
<br/>
<mark>This text isMarked</mark>
<br/>
<p>My favorite color is <del>blue</del> red.</p>
<p>My favorite <ins>color</ins> is red.</p>
<p>This is <sub>subscripted</sub> text.</p>
<p>This is <sup>superscripted</sup> text.</p>

</body>
</html>
```

This text is normal.

**This text is bold.**

**This text is strong**

*This text is italic*

*This text is emphasized*

This text is Small

**This text isMarked**

My favorite color is ~~blue~~ red.

My favorite color is red.

This is subscripted text.

This is superscripted text.



code4.txt

# تگ های نقل قول (Quotation)

- از تگ `<q>` برای نقل قول گذاری استفاده می شود و متن مورد نظر در داخل این تگ قرار می گیرد. این تگ برای نقل قول های کوتاه بکار می رود. عملکرد این تگ به صورت inline است.

```
<!DOCTYPE html>
<html>
<body>
```

```
<p>Browsers usually insert quotation marks around the q element.</p>
```

```
<p>WWF's goal is to: <q>Build a future where people live in harmony with nature.
</q></p>
```

```
</body>
</html>
```



code5.txt

Browsers usually insert quotation marks around the q element.

WWF's goal is to: "Build a future where people live in harmony with nature."

# نقل قول <blockquote>

- این تگ نیز برای نقل قول گذاری بکار می رود و کارکرد آن به صورت بلاک می باشد که یک بخشی را برای نقل قول از منبعی دیگر تعیین می کند.

```
<!DOCTYPE html>
<html>
<body>

<p>Browsers usually indent blockquote elements.</p>

<blockquote cite="http://www.worldwildlife.org/who/index.html">
For 50 years, WWF has been protecting the future of nature.
The world's leading conservation organization,
WWF works in 100 countries and is supported by
1.2 million members in the United States and
close to 5 million globally.
</blockquote>

</body>
</html>
```



code6.txt

Browsers usually indent blockquote elements.

For 50 years, WWF has been protecting the future of nature. The world's leading conservation organization, WWF works in 100 countries and is supported by 1.2 million members in the United States and close to 5 million globally.

# تگ اختصار Abbreviations

- برای اختصار سازی از تگ `<abbr>` استفاده می شود.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p>The <abbr title="World Health Organization">WHO</abbr> was founded in  
1948.</p>
```

```
<p>Marking up abbreviations can give useful information to browsers, translation  
systems and search-engines.</p>
```

```
</body>
```

```
</html>
```



code7.txt

The WHO was founded in 1948.

Mark World Health Organization give useful information to browsers, translation systems and search-engines.

# توضیحات (Comments)

- به منظور اضافه کردن توضیحات برای افزایش خوانایی کدها و مستند سازی می توان از تگ زیر استفاده کرد.

- `<!-- Write your comments here -->`

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<!-- This is a comment -->
```

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

```
<!-- Comments are not displayed in the browser -->
```

```
</body>
```

```
</html>
```

This is a paragraph.



code&txt



# توضیحات شرطی (Conditional Comments)

- توضیحات شرطی فقط توسط مرورگر [Internet Explorer](#) اجرا می شود.

```
<!--[if IE 9]>
```

```
.... some HTML here ....
```

```
<![endif]-->
```

# لینک در HTML (Links)

- برای ایجاد پیوند (Hyperlink) در صفحات وب می توان از تگ `a` (مخفف عبارت anchor) استفاده کرد و خصوصیت آن را تعیین نمود.

```
<!DOCTYPE html>
<html>
<body>

<p><a href="https://www.w3schools.com/html/">Visit our HTML tutorial</a></p>

</body>
</html>
```



code9.txt

[Visit our HTML tutorial](https://www.w3schools.com/html/)

**Note:** A link does not have to be text. It can be an image or any other HTML element.

# لینک گذاری روی عکس

```
<!DOCTYPE html>
<html>
<body>

<p>The image is a link. You can click on it.</p>

<a href="default.asp">
  
</a>

<p>We have added "border:0" to prevent IE9 (and earlier) from displaying a border
around the image.</p>

</body>
</html>
```

The image is a link. You can click on it.



We have added "border:0" to prevent IE9 (and earlier) from displaying a border around the image.

# رنگ لینک ها

- به صورت پیش فرض رنگ لینک ها در حالات مختلف بصورت زیر است. که می توان این رنگ بندی را به کمک خصوصیت style تغییر داد.

- An **unvisited** link is **underlined** and **blue**
- A **visited** link is **underlined** and **purple**
- An **active** link is **underlined** and **red**

[Visit our HTML tutorial](#)

[Visit our HTML tutorial](#)

# رنگ لینک ها

```
<head>
<style>
a:link {
    color: green;
    background-color: transparent;
    text-decoration: none;
}
a:visited {
    color: pink;
    background-color: transparent;
    text-decoration: none;
}
a:hover {
    color: red;
    background-color: transparent;
    text-decoration: underline;
}
a:active {
    color: yellow;
    background-color: transparent;
    text-decoration: underline;
}
</style>
</head>
```

You can change the default colors of links

HTML Images



code9-1.txt

```
<body>
```

```
<p>You can change the default colors of links</p>
```

```
<a href="html_images.asp" target="_blank">HTML Images</a>
```

```
</body>
```

# لینک محلی ( Local Links )

- برای پیوند به صفحه ای در وب سایت جاری می توان از ساختار زیر استفاده کرد.

## Example

```
<a href="html_images.asp">HTML Images</a>
```

# خصوصیت target در تگ <a>

- این خصوصیت برای نحوه بازشدن مرورگر در هنگام کلیک روی لینک مورد نظر بکار می رود.

## Example

```
<a href="https://www.w3schools.com/" target="_blank">Visit W3Schools!</a>
```

- \_blank - Opens the linked document in a new window or tab
- \_self - Opens the linked document in the same window/tab as it was clicked (this is default)
- \_parent - Opens the linked document in the parent frame
- \_top - Opens the linked document in the full body of the window
- filename - Opens the linked document in a named frame

# نحوه قرار دادن عکس در صفحات وب

- برای افزودن تصاویر به صفحه وب از برچسب `<img>` می توان استفاده کرد.  
این تگ دارای مشخصه ای برای تعیین منبع تصویر می باشد.

```
<!DOCTYPE html>  
<html>  
<body>  
  
  
  
</body>  
</html>
```



code10.txt





## مشخصه های تصویر

- **src** : نام و نشانی محل قرارگیری تصویر را مشخص می کند. src از عبارت Source گرفته شده است.
- **alt**: چنانچه به هر دلیل مرورگر قادر به نمایش تصویر نباشد، متنی را که در این مشخصه درج شده برای بازدید کننده نمایش می دهد. alt از عبارت alternative گرفته شده است. بهتر است عبارتی که در این مشخصه نوشته می شود، متناسب با تصویر باشد.

**Note:** The alt attribute is required. A web page will not validate correctly without it.

# توجه

- بهتر است برای تنظیم اندازه تصویر خصوصیت های **Width** و **Height** را بکار ببرید و برای تنظیم از خصوصیت **style attribute** این ویژگی تنظیم شود.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
img {  
    width:100%;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```

```

```

```

```
</body>
```

```
</html>
```

# Animated Images

- Html اجازه استفاده از عکس های gif. را نیز می دهد.

```
<!DOCTYPE html>
<html>
<body>

<p>The GIF standard allows moving images.</p>



</body>
</html>
```

The GIF standard allows moving images.



## تگ <picture>

- این تگ یکی از تگ های جدید HTML5 است . با کمک این تگ می توانیم چندین منبع عکس مشخص کنیم و برای هر کدام شرایط تعیین کنیم که کدام عکس چه موقع نمایش یابد . یعنی به شما امکان می دهد یک تصویر به کلی متفاوت را بر اساس یکی از شرایط زیر لود کنید.

❖ بر اساس مدیا کوئری ها مثل ارتفاع مرورگر , عرض مرورگر و جهت صفحه (افقی یا عمودی)

❖ بر اساس تراکم پیکسلی

- همه مرورگرهای جدید از این تگ پشتیبانی می کنند.

```
<picture>
<source media="(min-width: 650px)" srcset="img_pink_flowers.jpg">
<source media="(min-width: 465px)" srcset="img_white_flower.jpg">

</picture>
```

# <picture> ٹگ

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>
<body>

<picture>
  <source media="(min-width: 650px)" srcset="img_pink_flowers.jpg">
  <source media="(min-width: 465px)" srcset="img_white_flower.jpg">
  
</picture>

</body>
</html>
```



code11.txt



**Note:** Always specify an <img> element as the last child element of the <picture> element. The <img> element is used by browsers that do not support the <picture> element, or if none of the <source> tags matched.

# ایجاد جدول

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94
John	Doe	80



code12.txt

# ایجاد لیست

- برای ایجاد لیست سه نوع تگ مختلف وجود دارد که هر یک کاربرد خاصی را دارا می باشد.

## ❖ لیست غیر ترتیبی

✓ از تگ `<ul>` استفاده می شود.

## ❖ لیست ترتیبی

✓ از تگ `<ol>` استفاده می شود.

## ❖ لیست توضیحی

✓ از تگ `<dl>` استفاده می شود.

### An Unordered List:

- Item
- Item
- Item
- Item

### An Ordered List:

1. First item
2. Second item
3. Third item
4. Fourth item

# ایجاد لیست

```
<!DOCTYPE html>
<html>
<body>

<h2>An Unordered HTML List</h2>

<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>

<h2>An Ordered HTML List</h2>

<ol>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>

</body>
</html>
```

## An Unordered HTML List

- Coffee
- Tea
- Milk

## An Ordered HTML List

1. Coffee
2. Tea
3. Milk



code13.txt



# انتخاب نشانه گر آیتم هاس لیست

## Unordered HTML List - Choose List Item Marker

Value	Description
disc	Sets the list item marker to a bullet (default)
circle	Sets the list item marker to a circle
square	Sets the list item marker to a square
none	The list items will not be marked

```
<ul style="list-style-type:disc">  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Milk</li>  
</ul>
```

## Ordered HTML List - The Type Attribute

Type	Description
type="1"	The list items will be numbered with numbers (default)
type="A"	The list items will be numbered with uppercase letters
type="a"	The list items will be numbered with lowercase letters
type="I"	The list items will be numbered with uppercase roman numbers
type="i"	The list items will be numbered with lowercase roman numbers

```
<ol type="1">  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Milk</li>  
</ol>
```

# لیست توضیحی

```
<!DOCTYPE html>
<html>
<body>

<h2>A Description List</h2>

<dl>
  <dt>Coffee</dt>
  <dd>- black hot drink</dd>
  <dt>Milk</dt>
  <dd>- white cold drink</dd>
</dl>

</body>
</html>
```

## A Description List

Coffee

- black hot drink

Milk

- white cold drink



code14.txt

## نکته

- آیتم های لیست می تواند عناصر دیگر html از قبیل تصاویر، لینک و ... باشد.

# لیست افقی (Horizontal Lists)

- لیست های ایجاد شده می توانند به کمک `style` به صورت افقی نیز قرار بگیرند.  
یکی از کاربردهای این مورد ایجاد منو در وب سایت می باشد.

# مثال لیست افقی

```
<body>

<ul>
  <li><a href="#home">Home</a></li>
  <li><a href="#news">News</a></li>
  <li><a href="#contact">Contact</a></li>
  <li><a href="#about">About</a></li>
</ul>

</body>
```

Home   News   Contact   About



code15.txt

```
<style>
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
  background-color: #333333;
}

li {
  float: left;
}

li a {
  display: block;
  color: white;
  text-align: center;
  padding: 16px;
  text-decoration: none;
}

li a:hover {
  background-color: #111111;
}
</style>
```

# Block and Inline Elements

- تگ های html از لحاظ حالت نمایش بصورت **Inline** یا **Block** می باشد. عناصر Inline بصورت پیش فرض باعث شکسته شدن خط نمی شوند ولی عناصر block باعث شکسته شدن خط می شوند.

# Block and Inline Elements

Block level elements in HTML:

<address>	<article>	<aside>	<blockquote>	<canvas>	<dd>	<div>
<dl>	<dt>	<fieldset>	<figcaption>	<figure>	<footer>	<form>
<h1>-<h6>	<header>	<hr>	<li>	<main>	<nav>	<noscript>
<ol>	<output>	<p>	<pre>	<section>	<table>	<tfoot>
<ul>	<video>					

Inline elements in HTML:

<a>	<abbr>	<acronym>	<b>	<bdo>	<big>	 
<button>	<cite>	<code>	<dfn>	<em>	<i>	<img>
<input>	<kbd>	<label>	<map>	<object>	<q>	<samp>
<script>	<select>	<small>	<span>	<strong>	<sub>	<sup>
<textarea>	<time>	<tt>	<var>			

# تگ <div>

- عنصر div اغلب به عنوان یک نگه دارنده (container) برای عناصر دیگر بکار می

رود.

```
<!DOCTYPE html>  
<html>  
<body>
```

```
<div style="background-color:black;color:white;padding:20px;">  
  <h2>London</h2>  
  <p>London is the capital city of England. It is the most populous city in the  
United Kingdom, with a metropolitan area of over 13 million inhabitants.</p>  
  <p>Standing on the River Thames, London has been a major settlement for two  
millennia, its history going back to its founding by the Romans, who named it  
Londinium.</p>  
</div>
```

```
</body>  
</html>
```



code16.txt

## London

London is the capital city of England. It is the most populous city in the United Kingdom, with a metropolitan area of over 13 million inhabitants.

Standing on the River Thames, London has been a major settlement for two millennia, its history going back to its founding by the Romans, who named it Londinium.



## عنصر <span>

- این عنصر اغلب به عنوان یک نگهدارنده برای متن بکار می رود. این المان همانند یک جعبه برای متن‌هاست. در واقع این المان یک بخش از متن مورد نظر را زیر پرچم خود می‌گیرد و استایل دلخواه شما را بدان می‌دهد.

```
<!DOCTYPE html>
<html>
<body>

<h1>My <span style="color:red">Important</span> Heading</h1>

</body>
</html>
```



code17.txt

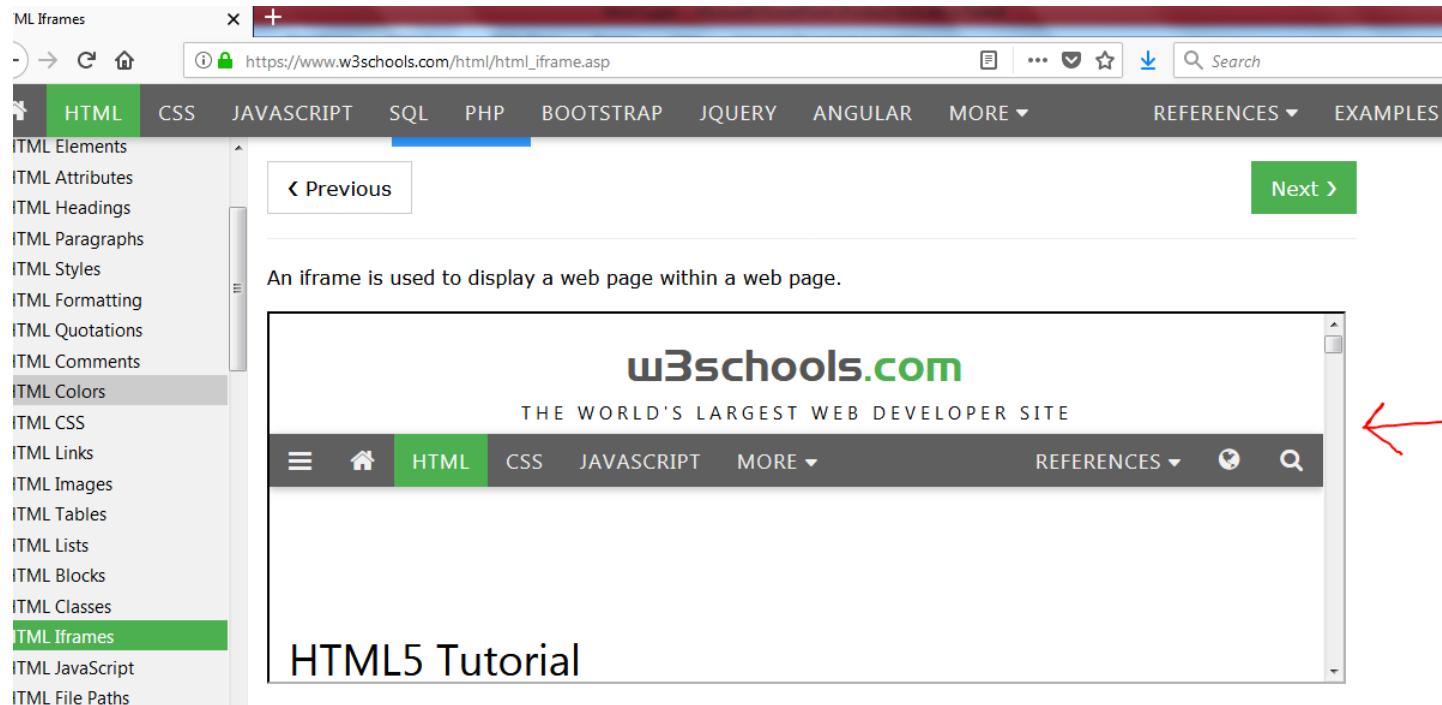
**My Important Heading**

# Span & Div

Tag	Description
<u>&lt;div&gt;</u>	Defines a section in a document (block-level)
<u>&lt;span&gt;</u>	Defines a section in a document (inline)

# iframes

- این عنصر به منظور نمایش یک صفحه وب در داخل صفحه وبی دیگر بکار می رود.



# نحوه استفاده از iframe

```
<!DOCTYPE html>  
<html>  
<body>  
  
<iframe src="demo_iframe.htm" height="200" width="300"></iframe>  
  
</body>  
</html>
```



code1&txt

**This page is  
displayed in an  
iframe**

# iframe - Target for a Link

An iframe can be used as the target frame for a link.

The **target** attribute of the link must refer to the **name** attribute of the iframe:

## Example

```
<iframe src="demo_iframe.htm" name="iframe_a"></iframe>  
  
<p><a href="https://www.w3schools.com" target="iframe_a">W3Schools.com</a></p>
```

# iframe - Target for a Link

```
<!DOCTYPE html>
<html>
<body>

<iframe height="300px" width="100%" src="demo_iframe.htm" name="iframe_a">
</iframe>

<p><a href="https://www.w3schools.com" target="iframe_a">W3Schools.com</a></p>

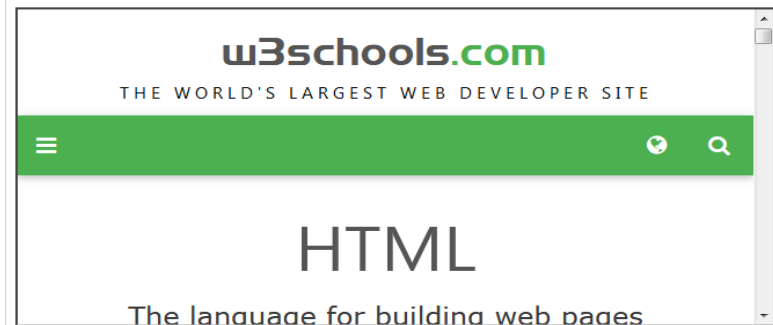
<p>When the target of a link matches the name of an iframe, the link will open in
the iframe.</p>

</body>
</html>
```



code19.txt

This page is displayed in an iframe



# مسیر فایل ها HTML File Paths

Path	Description
<code>&lt;img src="picture.jpg"&gt;</code>	picture.jpg is located in the same folder as the current page
<code>&lt;img src="images/picture.jpg"&gt;</code>	picture.jpg is located in the images folder in the current folder
<code>&lt;img src="/images/picture.jpg"&gt;</code>	picture.jpg is located in the images folder at the root of the current web
<code>&lt;img src="../../picture.jpg"&gt;</code>	picture.jpg is located in the folder one level up from the current folder

# مسیر فایل ها در HTML

- مسیر فایل محل قرارگیری یک فایل را که در وب سایت استفاده شده است را مشخص می کند. مسیر فایل برای ایجاد ارتباط با فایل های خارجی زیر بکار می رود.

Web pages ✓

Images ✓

Style sheets ✓

JavaScripts ✓



# Absolute File Paths آدرس دهی مطلق

- An absolute file path is the full URL to an internet file:

## Example

```

```

# آدرس دهی نسبی Relative File Paths

- این نوع آدرس دهی به این صورت عمل می کند که صفحه html جاری را به عنوان ریشه در نظر می گیرد. چنانچه بخواهیم به صفحات با فایلی که در یک محل (پوشه) قرار دارند اشاره کنیم کافی است تنها به نام آن صفحه یا فایل را ذکر کنیم. اما اگر صفحه یا فایل مورد درخواست در یک دایرکتوری در سطح پایین تر قرار داشته باشد، باید نام دایرکتوری را نیز ذکر کنیم. اگر صفحه یا فایل در سطح بالاتری قرار داشته باشد، باید با دو نقطه .. به سطح بالاتر اشاره نمود.

# آدرس دهی نسبی Relative File Paths

In this example the file path points to a file in the images folder located at the root of the current web:

## Example

```

```

In this example the file path points to a file in the images folder located in the current folder:

## Example

```

```

In this example the file path points to a file in the images folder located in the folder one level above the current folder:

## Example

```

```

# بهترین روش آدرس دهی Best Practice

- بهترین و اصولی ترین روش در آدرس دهی، استفاده از روش آدرس دهی نسبی می باشد.

- When using relative file paths, your web pages will not be bound to your current base URL. All links will work on your own computer (localhost) as well as on your current public domain and your future public domains.

# The HTML <head> Element

- تگ head به عنوان یک نگهدارنده ( container ) برای متادیتا ( metadata ) می باشد. متادیتا ها داده هایی در مورد سند html ایجاد شده هستند. البته این متادیتاها توسط مروگرها نمایش داده نمی شوند.

- **metadata** (data about data)

- تگ های ایجاد کننده متادیتا عبارتند از:

<title> ❖

<style> ❖

<meta> ❖

<link> ❖

<script> ❖

<base> ❖

# HTML head Elements

| Tag                   | Description   |
|-----------------------|---|
| <u>&lt;head&gt;</u>   | Defines information about the document                                |
| <u>&lt;title&gt;</u>  | Defines the title of a document                                       |
| <u>&lt;base&gt;</u>   | Defines a default address or a default target for all links on a page |
| <u>&lt;link&gt;</u>   | Defines the relationship between a document and an external resource  |
| <u>&lt;meta&gt;</u>   | Defines metadata about an HTML document                               |
| <u>&lt;script&gt;</u> | Defines a client-side script  |
| <u>&lt;style&gt;</u>  | Defines style information for a document                              |

## تگ meta

- موتورهای جست و نیز برای بهینه سازی و سرعت بخشیدن به عمل جست و، ی سری استانداردهایی را برای سازندگان وب سایت ها تعریف کرده اند تا با پیروی از این استانداردها، موتورهای جست و راحت تر بتوانند مطالب وب سایت ها را جست و و امتیاز بندی کنند. یکی از مواردی که باعث بهبود وضعیت ی سایت در نتایج جست و می شود، استفاده از تگ **<meta>** است.

## تگ meta

- این تگ، اطلاعاتی در مورد صفحه html مهیا می کند که موتورهای جست و از این اطلاعات برای دسته بندی صفحات وب سایت ها استفاده می کنند. شما می توانید در این تگ، کلمات کلیدی و یا توضیحاتی در مورد وب سایت خود قرار دهید تا موتورهای جست و با استفاده از آن بتوانند صفحه شما را به نحو بهتری فهرست کنند.



# عناصر تگ meta

Define the character set used:

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

Define a description of your web page:

```
<meta name="description" content="Free Web tutorials">
```

Define keywords for search engines:

```
<meta name="keywords" content="HTML, CSS, XML, JavaScript">
```

Define the author of a page:

```
<meta name="author" content="John Doe">
```

# عناصر طرح بندی صفحات Layout Elements

- وب سایت ها محتوایشان را اغلب در چند ستون نمایش می دهند (همانند مجله یا نشریه). **HTML 5** عناصر معنایی جدیدی را که قسمت های یک صفحه وب را تعریف می کنند معرفی کرده است.



- **<header>** - Defines a header for a document or a section
- **<nav>** - Defines a container for navigation links
- **<section>** - Defines a section in a document
- **<article>** - Defines an independent self-contained article
- **<aside>** - Defines content aside from the content (like a sidebar)
- **<footer>** - Defines a footer for a document or a section
- **<details>** - Defines additional details
- **<summary>** - Defines a heading for the **<details>** element

# فرم ها HTML Forms

- تگ `<form>` یک فرم را تعریف می کند که از آن برای جمع آوری ورودی های کاربر می توان استفاده کرد. عناصر فرم موارد مختلفی مانند text fields, checkboxes, radio buttons, submit buttons و غیره هستند.

`<form>`

.

*form elements*

.

`</form>`

## تگ <input>

- تگ `<input>` مهمترین عنصر یک فرم است. این عنصر می تواند به حالت های مختلفی نمایش داده شود.

| Type                                     | Description  |
|--|--|
| <code>&lt;input type="text"&gt;</code>   | Defines a one-line text input field                        |
| <code>&lt;input type="radio"&gt;</code>  | Defines a radio button (for selecting one of many choices) |
| <code>&lt;input type="submit"&gt;</code> | Defines a submit button (for submitting the form)          |

# ورودی متن Text Input

- کد `<input type="text">` یک ورودی تک خطی برای متن ایجاد می کند.

## Example

```
<form>
  First name:<br>
  <input type="text" name="firstname"><br>
  Last name:<br>
  <input type="text" name="lastname">
</form>
```

First name:

Last name:

Note that the form itself is not visible.

Also note that the default width of a text input field is 20 characters.



code20.txt

**Note:** The form itself is not visible. Also note that the default width of a text field is 20 characters.

# Radio Button Input

- کد `<input type="radio">` یک radio button را تعریف می کند. radio button ها امکان انتخاب یک گزینه از تعدادی محدود گزینه را فراهم می آورد.

## Example

```
<form>
  <input type="radio" name="gender" value="male" checked> Male<br>
  <input type="radio" name="gender" value="female"> Female<br>
  <input type="radio" name="gender" value="other"> Other
</form>
```

☒ Male  
☐ Female  
☐ Other



code21.txt

# لیست انتخابی

- با عنصر `<select>` می توان drop-down list ایجاد کرد.

```
<form action="/action_page.php">
  <select name="cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat">Fiat</option>
    <option value="audi">Audi</option>
  </select>
  <br><br>
  <input type="submit">
</form>
```

Volvo ▼

Submit Query

# لیست انتخابی

By default, the first item in the drop-down list is selected.

To define a pre-selected option, add the **selected** attribute to the option:

## Example

```
<option value="fiat" selected>Fiat</option>
```



# لیست انتخابی

```
<form action="/action_page.php">
  <select name="cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat" selected>Fiat</option>
    <option value="audi">Audi</option>
  </select>
  <br><br>
  <input type="submit">
</form>
```

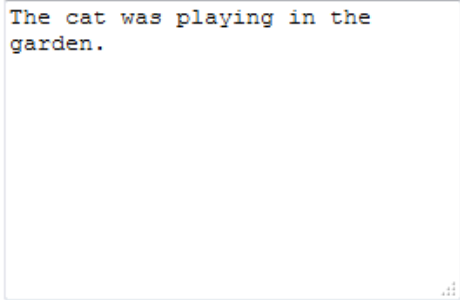
Fiat ▼

Submit Query

## <textarea> عنصر

### Example

```
<textarea name="message" rows="10" cols="30">  
The cat was playing in the garden.  
</textarea>
```

A visual representation of a text area. It is a rectangular box with a thin border. Inside the box, the text "The cat was playing in the garden." is written in a monospaced font. The text is aligned to the left and occupies the first line of the box. The box is white with a light gray border.

The **rows** attribute specifies the visible number of lines in a text area.

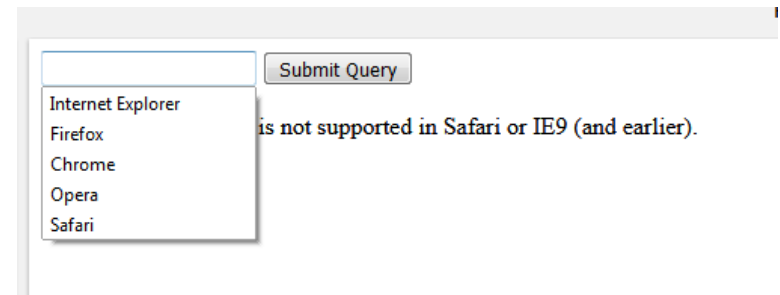
The **cols** attribute specifies the visible width of a text area.

# عنصر <datalist>

- تگ `<datalist>` یک لیست از گزینه های از قبل تعریف شده برای یک عنصر `<input>` مشخص می کند.

## Example

```
<form action="/action_page.php">
  <input list="browsers">
  <datalist id="browsers">
    <option value="Internet Explorer">
    <option value="Firefox">
    <option value="Chrome">
    <option value="Opera">
    <option value="Safari">
  </datalist>
</form>
```



The screenshot shows a web form with a text input field and a 'Submit Query' button. The input field has a dropdown menu open, displaying a list of browsers: Internet Explorer, Firefox, Chrome, Opera, and Safari. To the right of the input field, there is a message: 'is not supported in Safari or IE9 (and earlier)'.



code21-2.txt

# <button> **عنصر**

The **<button>** element defines a clickable **button**:

## Example

```
<button type="button" onclick="alert('Hello World!')">Click Me!</button>
```

Hello World!

OK

Click Me!

**Note:** Always specify the **type** attribute for the button element. Different browsers may use different default types for the button element.

# Submit Button

- کد `<input type="submit">` یک دکمه برای تأیید داده های فرم به form-handler ایجاد می کند.

## Example

```
<form action="/action_page.php">  
  First name:<br>  
  <input type="text" name="firstname" value="Mickey"><br>  
  Last name:<br>  
  <input type="text" name="lastname" value="Mouse"><br><br>  
  <input type="submit" value="Submit">  
</form>
```

First name:

Mickey

Last name:

Mouse

Submit



code22.txt

## Submitted Form Data

Your input was received as:

firstname=Mickey&lastname=Mouse

The server has processed your input and returned this answer.

# خصوصیت Action در تگ Form

- آدرسی که در خصوصیت **action** قرار می گیرد، باعث می شود که اطلاعات فرم پس از فشردن دکمه submit به آن آدرس فرستاده شود.

```
<form action="/action_page.php">
```

- **نکته:** اگر خصوصیت action نوشته نشود، اطلاعات به صفحه جاری فرستاده می شود.

# خصوصیت Method در تگ Form

- این خصوصیت تعیین کننده متد HTTP (GET یا POST) مورد استفاده در ارسال داده ها بعد از کلیک دکمه submit است.

## Example

```
<form action="/action_page.php" method="get">
```

## Example

```
<form action="/action_page.php" method="post">
```



Code23.txt

# متد GET

- متد پیش فرض وقتی فرم submit می شود، GET می باشد. در این متد داده ها قابل مشاهده در فیلد آدرس هستند.

First name:

Last name:

```
<form action="/action_page.php" method="get">
```

```
/action_page.php?firstname=Mickey&lastname=Mouse
```



# نکات مورد توجه در متد GET

## Notes on GET:

- Appends form-data into the URL in name/value pairs
- The length of a URL is limited (about 3000 characters)
- Never use GET to send sensitive data! (will be visible in the URL)
- Useful for form submissions where a user want to bookmark the result
- GET is better for non-secure data, like query strings in Google

# متد POST

- از متد **POST** زمانی استفاده می شود که داد های ارسالی فرم مهم و اطلاعات شخصی می باشند. این متد، داده های submit شده را هنگام ارسال نشان نمی دهد.

# نکات مورد توجه در متد POST

## Notes on POST:

- POST has no size limitations, and can be used to send large amounts of data.
- Form submissions with POST cannot be bookmarked

# خصوصیت Name

- هر فیلد ورودی در فرم، باید یک خصوصیت **نام** داشته باشد. اگر این خصوصیت نادیده گرفته شود، داده های آن فیلد ورودی ارسال نخواهد شد.

## Example

```
<form action="/action_page.php">  
  First name:<br>  
  <input type="text" value="Mickey"><br>  
  Last name:<br>  
  <input type="text" name="lastname" value="Mouse"><br><br>  
  <input type="submit" value="Submit">  
</form>
```

First name:

Last name:

## Submitted Form Data

Your input was received as:

lastname=Mouse

The server has processed your input and returned this answer.



code24.txt

# گروه بندی داده‌های فرم با <fieldset>

- <fieldset> برای گروه بندی داده های مرتبط در یک فرم بکار می رود. که با <legend> می توان برای گروه ایجاد شده عنوانی هم انتخاب کرد.

## Example

```
<form action="/action_page.php">
  <fieldset>
    <legend>Personal information:</legend>
    First name:<br>
    <input type="text" name="firstname" value="Mickey"><br>
    Last name:<br>
    <input type="text" name="lastname" value="Mouse"><br><br>
    <input type="submit" value="Submit">
  </fieldset>
</form>
```

Personal information: \_\_\_\_\_

First name:

Last name:



code25.txt

# خصایت های form

Attribute	Description
accept-charset	Specifies the charset used in the submitted form (default: the page charset).
action	Specifies an address (url) where to submit the form (default: the submitting page).
autocomplete	Specifies if the browser should autocomplete the form (default: on).
enctype	Specifies the encoding of the submitted data (default: is url-encoded).
method	Specifies the HTTP method used when submitting the form (default: GET).
name	Specifies a name used to identify the form (for DOM usage: document.forms.name).
novalidate	Specifies that the browser should not validate the form.
target	Specifies the target of the address in the action attribute (default: _self).

# Input Type Password

`<input type="password">` defines a **password field**:

## Example

```
<form>
  User name:<br>
  <input type="text" name="username"><br>
  User password:<br>
  <input type="password" name="psw">
</form>
```

User name:

User password:

The characters in a password field are masked (shown as asterisks or circles).

# Input Type Reset

`<input type="reset">` defines a **reset button** that will reset all form values to their default values:

## Example

```
<form action="/action_page.php">  
  First name:<br>  
  <input type="text" name="firstname" value="Mickey"><br>  
  Last name:<br>  
  <input type="text" name="lastname" value="Mouse"><br><br>  
  <input type="submit" value="Submit">  
  <input type="reset">  
</form>
```

First name:

Mickey

Last name:

Mouse

Submit

Reset



# Input Type Checkbox

`<input type="checkbox">` defines a **checkbox**.

Checkboxes let a user select ZERO or MORE options of a limited number of choices.

## Example

```
<form>
  <input type="checkbox" name="vehicle1" value="Bike"> I have a bike<br>
  <input type="checkbox" name="vehicle2" value="Car"> I have a car
</form>
```

☐ I have a bike

☐ I have a car

# HTML5 Input Types

• Input های جدید معرفی شده در html5:

Select your favorite color:

Birthday:

- color
- date
- datetime-local
- email
- month
- number
- range
- search
- tel
- time
- url
- week

[https://www.w3schools.com/html/html\\_form\\_input\\_types.asp](https://www.w3schools.com/html/html_form_input_types.asp)

# عناصر جدید HTML5

- عناصر جدیدی که در نسخه HTML5 معرفی شده اند عبارتند از:

New **semantic elements** like `<header>`, `<footer>`, `<article>`, and `<section>`.

New **attributes of form elements** like number, date, time, calendar, and range.

New **graphic elements**: `<svg>` and `<canvas>`.

New **multimedia elements**: `<audio>` and `<video>`.

# API های معرفی شده در HTML5

- New HTML5 API's (Application Programming Interfaces)
  - HTML Geolocation
  - HTML Drag and Drop
  - HTML Local Storage
  - HTML Application Cache
  - HTML Web Workers
  - HTML SSE

# عناصر حذف شده HTML5

- عناصر زیر در نسخه html4 وجود داشتند که در نسخه html5 حذف شده اند.

Removed Element	Use Instead
<acronym>	<abbr>
<applet>	<object>
<basefont>	CSS
<big>	CSS
<center>	CSS
<dir>	<ul>
<font>	CSS
<frame>	
<frameset>	
<noframes>	
<strike>	CSS, <s>, or <del>
<tt>	CSS


# عناصر معنایی Semantic Elements

- یک عنصر معنایی به طور واضح خود را برای مرورگر ( browser ) و توسعه دهنده ( developer )، توصیف می کند.

Examples of **non-semantic** elements: `<div>` and `<span>` - Tells nothing about its content.

Examples of **semantic** elements: `<form>`, `<table>`, and `<article>` - Clearly defines its content.

# پشتیبانی مرورگرها از عناصر معنایی

				
Yes	Yes	Yes	Yes	Yes

HTML5 semantic elements are supported in all modern browsers.

In addition, you can "teach" older browsers how to handle "unknown elements".

# عناصر معنایی جدید در HTML5

- بسیاری از وب سایت ها شامل کدهایی به صورت زیر برای بخش های مختلف صفحه هستند.
- `<div id="nav">`
- `<div class="header"> <div id="footer">`
- HTML5 برای تعریف قسمت های مختلف صفحه عنصرهای جدید را پیشنهاد داده است.



# عناصر معنایی جدید در HTML5

<article>

<aside>

<details>

<figcaption>

<figure>

<footer>

<header>

<main>

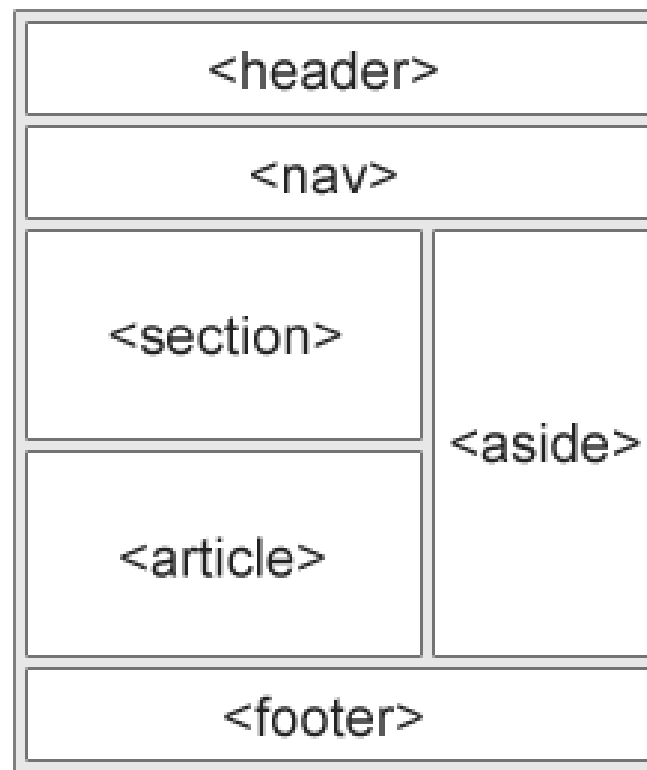
<mark>

<nav>

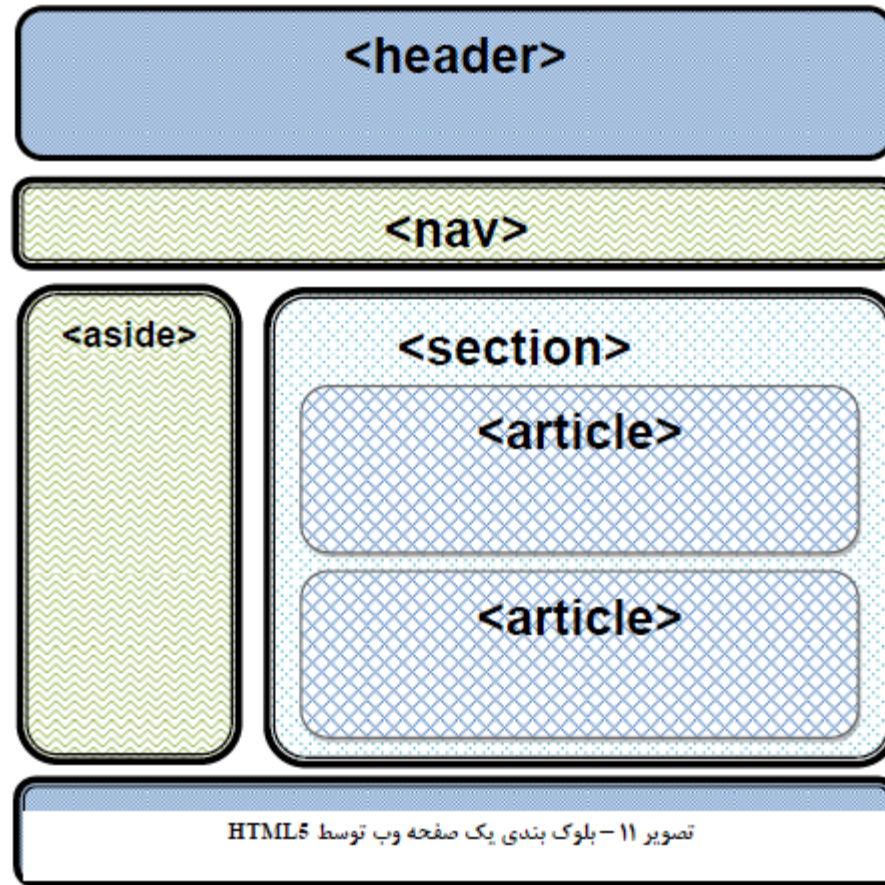
<section>

<summary>

<time>



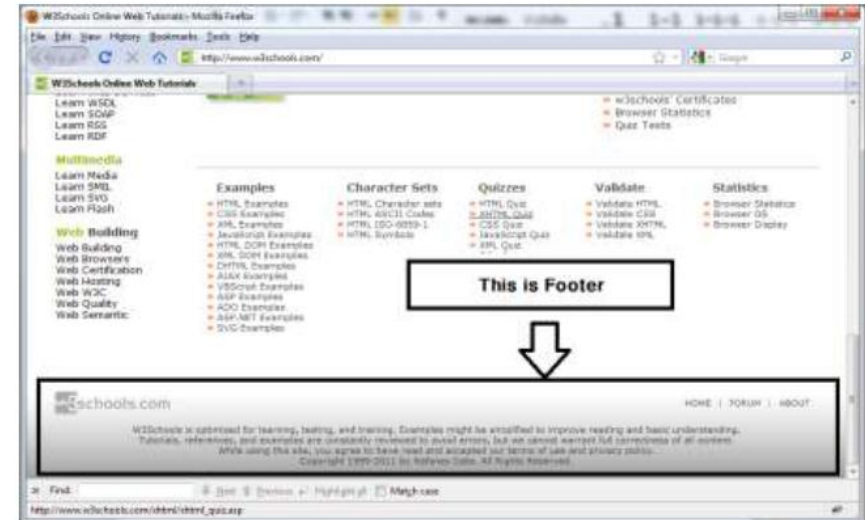
# بلوک بندی ہی صفحہ وب بر اساس html5



# Footer و Header



تصویر ۱۲ - Header یک سایت



تصویر ۱۳ - footer یک سایت

برای ایجاد header و footer از تگ مربوط به هر یک استفاده می کنیم. برای نمونه تکه کد زیر، چگونگی استفاده از تگ header را html5 نمایش می دهد.

```
<header>
  <img src = "/images/logo.png">
</header>
```

تکه کد زیر نیز نحوه استفاده از تگ footer در html5 را نمایش می دهد.

```
<footer>
  <p>Copyright © 2011 , company name.</p>
</footer>
```

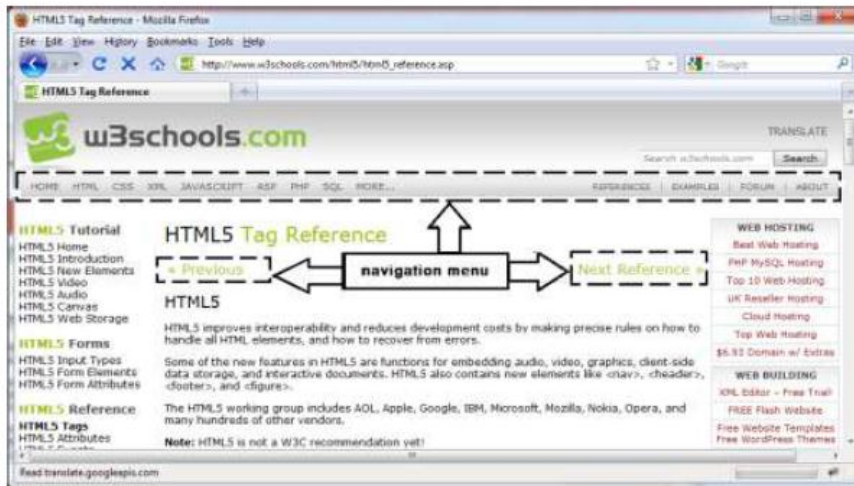
# کاربرد تگ section و article

تگ <section> برای تعیین "بخش ها" در صفحه بکار می رود. و در واقع مطالب صفحه توسط این تگ به بخش های مجزا از یکدیگر بلوک بندی می شوند. تگ <article> در داخل تگ <section> قرار میگیرد و محتوی موجود در تگ <section> را بلوک بندی کرده و هر کدام را به عنوان یک عنصر مجزا در نظر می گیرد. تصویر زیر این موضوع را مشخص می کند.

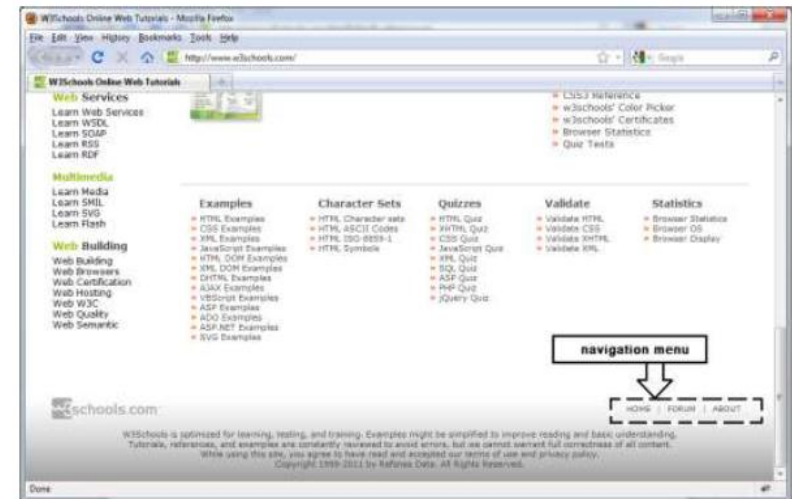


تصویر ۱۴ - section و article در یک صفحه وب

# نگ nav



تصویر ۱۵ - منوهای ناوبری بالای صفحه



تصویر ۱۶ - منوهای ناوبری پایین صفحه

<nav>

```
<a href="home.html">HOME</a> |  
<a href="forum.html">FORUM</a> |  
<a href="about.html">ABOUT</a>
```

</nav>



# تگ aside

```
<aside>
  <nav>
    <p><a href="#">learn html5</a></p>
    <p><a href="#">learn css3</a></p>
    <p><a href="#">learn javascript</a></p>
    <p><a href="#">learn php</a></p>
  </nav>
</aside>
```



تصویر ۱۷ - تگ <aside>

محتوی سایت که در کنار صفحه و جدا از محتوی اصلی سایت تعریف شده و با محتوی قسمت های دیگر صفحه مرتبط می باشند، در تگ <aside> قرار می گیرند. معمولا لینک های صفحه در این قسمت قرار می گیرند. در تصویر زیر نمونه ای از تگ <aside> را مشاهده می نمایید که با استفاده از خط چین مشخص شده است.

## تگ <section>

- این عنصر یک بخش را در سند تعریف می کند. یک صفحه وب می تواند به صورت عادی به بخش های مقدمه، محتوا و اطلاعات تماس تقسیم شود.

- تعریف بخش طبق مستندات W3C's HTML5:

- "A section is a thematic grouping of content, typically with a heading."

به استناد کنسرسیوم جهانی وب <section> شرح گروه بندی محتویات است که معمولاً با یک هدینگ (عنوان) به کار می رود.

## Example

```
<section>  
  <h1>WWF</h1>  
  <p>The World Wide Fund for Nature (WWF) is.....</p>  
</section>
```



code26.txt



## تگ <article>

- این عنصر محتوای مستقل را تعیین می کند. از تگ article برای تعیین یک مقاله که با محتویات سایت متفاوت باشد، استفاده می شود. نمونه هایی که این تگ برای آنها می تواند بکار رود عبارتند از:

Forum post ✓

Blog post ✓

Newspaper article ✓

## Example

```
<article>
  <h1>What Does WWF Do?</h1>
  <p>WWF's mission is to stop the degradation of our planet's natural environment,
    and build a future in which humans live in harmony with nature.</p>
</article>
```

## What Does WWF Do?

WWF's mission is to stop the degradation of our planet's natural environment, and build a future in which humans live in harmony with nature.



code27.txt

<https://www.w3.org/TR/2014/REC-html5-20141028/sections.html#the-article-element>

<article> 

The article element represents a complete, or self-contained, composition in a document, page, application, or site and that is, in principle, independently distributable or reusable, e.g. in syndication. This could be a forum post, a magazine or newspaper article, a blog entry, a user-submitted comment, an interactive widget or gadget, or any other independent item of content.

#### Code Example:

This example shows a blog post using the article element, with some schema.org annotations:

```
<article itemscope itemtype="http://schema.org/BlogPosting">
  <header>
    <h1 itemprop="headline">The Very First Rule of Life</h1>
    <p><time itemprop="datePublished" datetime="2009-10-09">3 days ago</time></p>
    <link itemprop="url" href="?comments=0">
  </header>
  <p>If there's a microphone anywhere near you, assume it's hot and
  sending whatever you're saying to the world. Seriously.</p>
  <p>...</p>
  <footer>
    <a itemprop="discussionUrl" href="?comments=1">Show comments...</a>
  </footer>
</article>
```

# تودرتو به کار بردن < article > در < section > یا بالعکس؟

- تگ < article > می توان یک محتوای مستقل باشد .
- تگ < section > برای تقسیم بندی سند می باشد .
- در اینترنت می توان صفحاتی با تگ های < section > که شامل تگ هایی از < article> باشند و همین طور تگ های < article > که حاوی تگ هایی از < section> پیدا کنید .
- به عنوان مثال در یک روزنامه : مقاله های ورزشی در قسمت ورزش و یک بخش فنی در هر مقاله به کار برد .

**Code Example:**

In the following example, we see an article (part of a larger Web page) about apples, containing two short sections.

```
<article>
  <header>
    <h2>Apples</h2>
    <p>Tasty, delicious fruit!</p>
  </header>
  <p>The apple is the pomaceous fruit of the apple tree.</p>
  <section>
    <h3>Red Delicious</h3>
    <p>These bright red apples are the most common found in many
      supermarkets.</p>
  </section>
  <section>
    <h3>Granny Smith</h3>
    <p>These juicy, green apples make a great filling for
      apple pies.</p>
  </section>
</article>
```

# عنصر <header>

- این تگ یک هدر را برای سند (document) یا بخش (section) تعیین می کند. همچنین در یک سند می توان از چندین تگ <header> استفاده کرد.

## Example



code2&txt

```
<article>
  <header>
    <h1>What Does WWF Do?</h1>
    <p>WWF's mission:</p>
  </header>
  <p>WWF's mission is to stop the degradation of our planet's natural environment,
  and build a future in which humans live in harmony with nature.</p>
</article>
```

## عنصر <footer>

• این تگ برای ایجاد فوتر در صفحه وب بکار می رود که معمولاً شامل اطلاعاتی از قبیل زیر است:

- author of the document
- copyright information
- links to terms of use
- contact information
- etc...

# مثال

```
<!DOCTYPE html>
<html>
<body>

<footer>
  <p>Posted by: Hege Refsnes</p>
  <p>Contact information: <a href="mailto:someone@example.com">
    someone@example.com</a>.</p>
</footer>

</body>
</html>
```

Posted by: Hege Refsnes

Contact information: [someone@example.com](mailto:someone@example.com).



code29.txt



## عنصر <nav>

- در این تگ لینک های داخلی یا خارجی در آن قرار می گیرند.

Notice that NOT all links of a document should be inside a <nav> element. The <nav> element is intended only for major block of navigation links.

# مثال

```
<!DOCTYPE html>
<html>
<body>

<nav>
  <a href="/html/">HTML</a> |
  <a href="/css/">CSS</a> |
  <a href="/js/">JavaScript</a> |
  <a href="/jquery/">jQuery</a>
</nav>

</body>
</html>
```

[HTML](#) | [CSS](#) | [JavaScript](#) | [jQuery](#)



code30.txt

## عنصر <aside>

- این تگ برای تعریف عناصری که در sidebar قرار می گیرند بکار می رود.

The <aside> element defines some content aside from the content it is placed in (like a sidebar).

The aside content should be related to the surrounding content.

---

```
<!DOCTYPE html>
<html>
<body>

<p>My family and I visited The Epcot center this summer.</p>

<aside>
  <h4>Epcot Center</h4>
  <p>The Epcot Center is a theme park in Disney World, Florida.</p>
</aside>

</body>
</html>
```

My family and I visited The Epcot center this summer.

#### Epcot Center

The Epcot Center is a theme park in Disney World, Florida.

# تگ های <figure> و <figcaption>

## Example

```
<figure>  
    
  <figcaption>Fig1. - The Pulpit Rock, Norway.</figcaption>  
</figure>
```



code31.txt



Fig.1 - The Pulpit Rock, Norway.

# Why Semantic Elements?

- According to the W3C, a **Semantic Web**: "Allows data to be shared and reused across applications, enterprises, and communities."

# خلاصه تگ های معنایی

Tag	Description
<u>&lt;article&gt;</u>	Defines an article
<u>&lt;aside&gt;</u>	Defines content aside from the page content
<u>&lt;details&gt;</u>	Defines additional details that the user can view or hide
<u>&lt;figcaption&gt;</u>	Defines a caption for a <figure> element
<u>&lt;figure&gt;</u>	Specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.
<u>&lt;footer&gt;</u>	Defines a footer for a document or section
<u>&lt;header&gt;</u>	Specifies a header for a document or section
<u>&lt;main&gt;</u>	Specifies the main content of a document
<u>&lt;mark&gt;</u>	Defines marked/highlighted text
<u>&lt;nav&gt;</u>	Defines navigation links
<u>&lt;section&gt;</u>	Defines a section in a document
<u>&lt;summary&gt;</u>	Defines a visible heading for a <details> element
<u>&lt;time&gt;</u>	Defines a date/time

# چند رسانه ای در وب HTML Multimedia

• در صفحات وب با فایل های چند رسانه ای زیر معمولاً سروکار داریم.



Sound ✓

Music ✓

Videos ✓

Movies ✓

Animations ✓



# HTML5 Video

- در html5 تگ `<video>` برای قرار گیری فایل ویدئویی در صفحه وب معرفی شده است.

## Browser Support

The numbers in the table specify the first browser version that fully supports the `<video>` element.

Element					
<code>&lt;video&gt;</code>	4.0	9.0	3.5	4.0	10.5

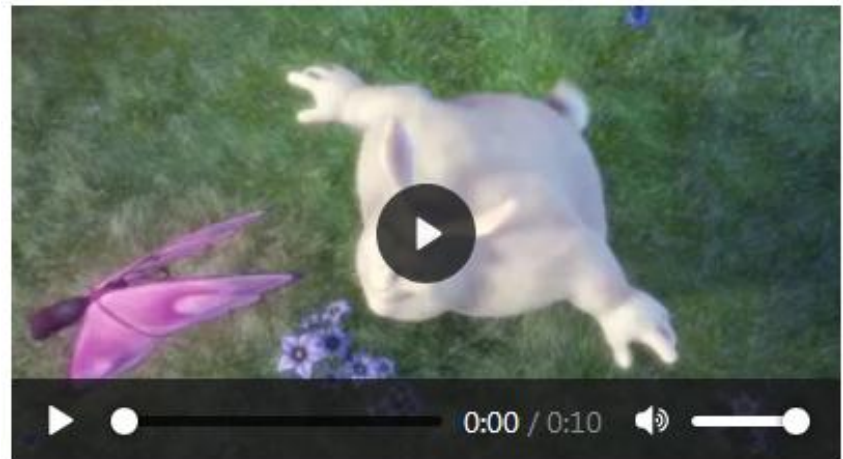
# The HTML <video> Element

## Example

```
<video width="320" height="240" controls>  
  <source src="movie.mp4" type="video/mp4">  
  <source src="movie.ogv" type="video/ogg">  
  Your browser does not support the video tag.  
</video>
```



code32.txt



# پخش خودکار ویدئو Autoplay

- برای پخش خودکار ویدئو با لود شدن صفحه وب می توان از خاصیت Autoplay استفاده کرد.

## Example

```
<video width="320" height="240" autoplay>  
  <source src="movie.mp4" type="video/mp4">  
  <source src="movie.ogg" type="video/ogg">  
Your browser does not support the video tag.  
</video>
```



The autoplay attribute does not work in mobile devices like iPad and iPhone.

# HTML Video - Browser Support

- در html5 از سه فرمت ویدئویی MP4, WebM, Ogg پشتیبانی می شود.

Browser	MP4	WebM	Ogg
Internet Explorer	YES	NO	NO
Chrome	YES	YES	YES
Firefox	YES	YES	YES
Safari	YES	NO	NO
Opera	YES (from Opera 25)	YES	YES

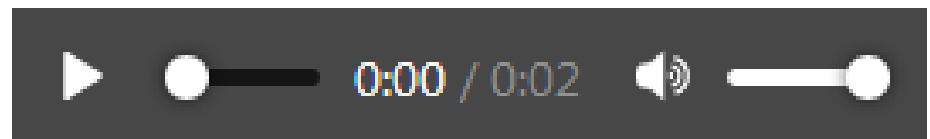
# Audio آڈیو

## Example

```
<audio controls>  
  <source src="horse.ogg" type="audio/ogg">  
  <source src="horse.mp3" type="audio/mpeg">  
  Your browser does not support the audio element.  
</audio>
```



code33.txt



# تگ Audio

- فرمت های صوتی پشتیبانی شده در html5 عبارتند از:

MP3 ✓

Wav ✓

Ogg ✓

Browser	MP3	Wav	Ogg
Internet Explorer	YES	NO	NO
Chrome	YES	YES	YES
Firefox	YES	YES	YES
Safari	YES	YES	NO
Opera	YES	YES	YES

# HTML YouTube Videos

- یکی از راه های خوب نمایش ویدئو در وب سایت استفاده از [youtube](#) می باشد. هر یک از ویدئوهای یوتیوب یک شناسه ای دارد که می توان از آن برای نمایش ویدئوی مورد نظر در وب سایت خود استفاده کرد.

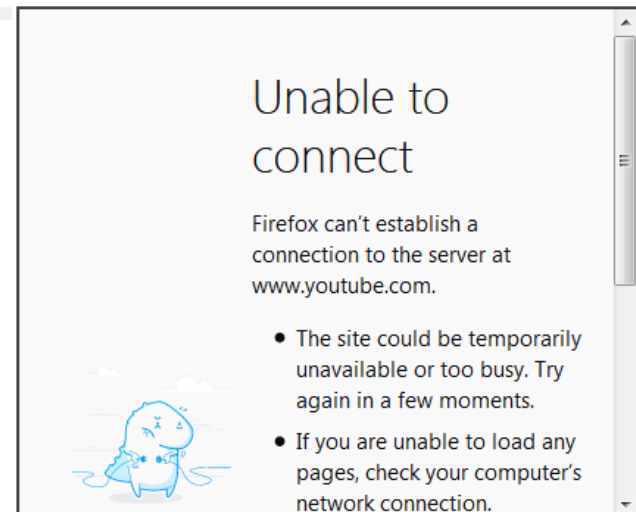
## Playing a YouTube Video in HTML

To play your video on a web page, do the following:

- Upload the video to YouTube
- Take a note of the video id
- Define an `<iframe>` element in your web page
- Let the `src` attribute point to the video URL
- Use the `width` and `height` attributes to specify the dimension of the player
- Add any other parameters to the URL (see below)

## Example - Using iFrame (recommended)

```
<iframe width="420" height="315"  
src="https://www.youtube.com/embed/tgbNymZ7vqY">  
</iframe>
```





# YouTube Autoplay

- می توان برای برای پخش خودکار ویدئوها با لود شدن صفحه وب به پارامتر autoplay مقدار ۱ را نسبت داد. البته این کار توصیه نمی شود.


Value 0 (default): The video will not play automatically when the player loads.

Value 1: The video will play automatically when the player loads.

**Note:** Take careful consideration when deciding to autoplay your videos. Automatically starting a video can annoy your visitor and end up causing more harm than good.

## YouTube - Autoplay

```
<iframe width="420" height="315"  
src="https://www.youtube.com/embed/tgbNymZ7vqY?autoplay=1">  
</iframe>
```



# YouTube Controls

Value 0: Player controls does not display.

Value 1 (default): Player controls display.

## YouTube - Controls

```
<iframe width="420" height="315"  
src="https://www.youtube.com/embed/tgbNymZ7vqY?controls=0">  
</iframe>
```



Linked Data