

فصل اول : HTML

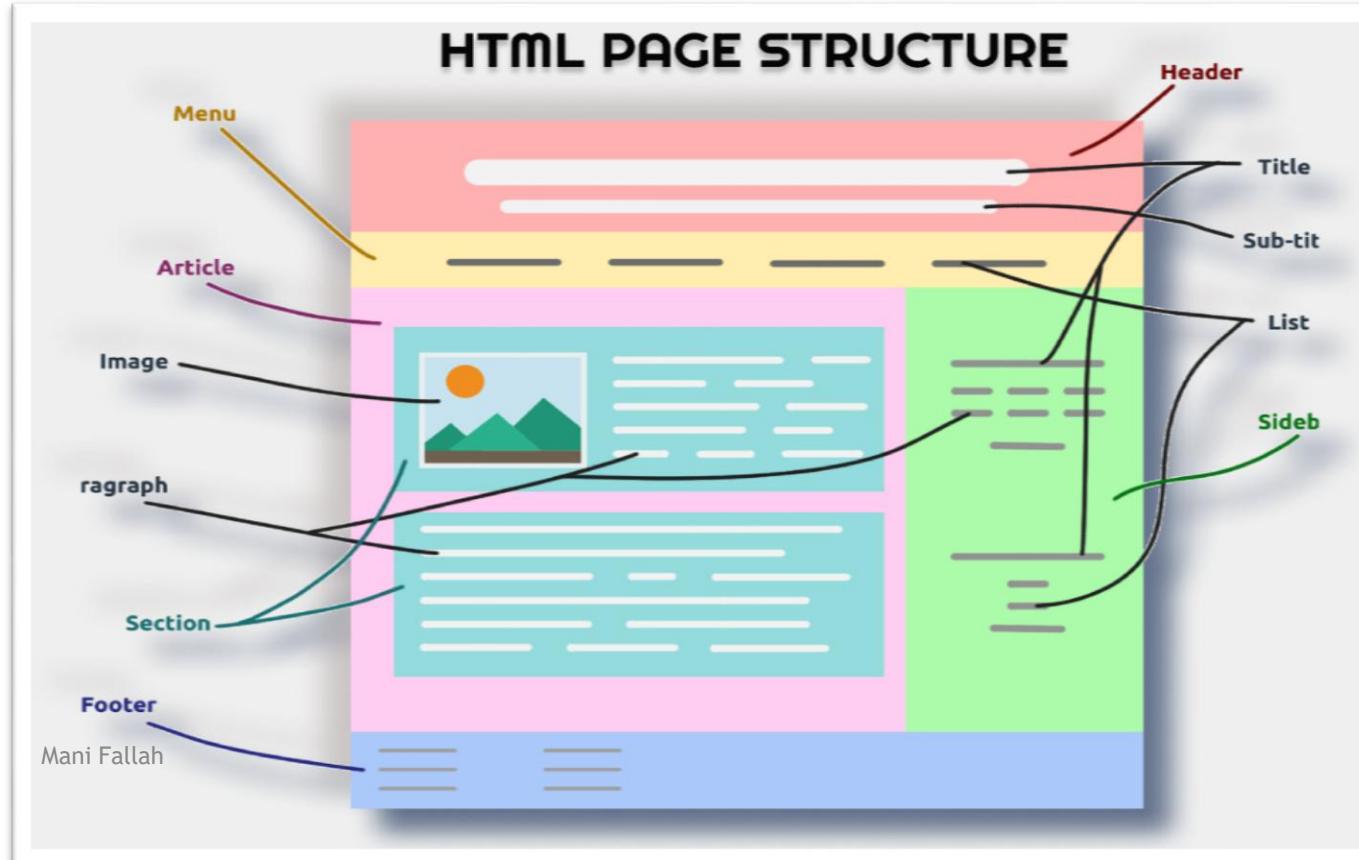
بخش اول: مقدمه‌ای بر html، المان‌های html، ساختار html، المان‌های متن در html

مقدمه‌ای بر html

- ▶ عبارت **HTML** مخفف **HyperText Markup Language** یک زبان نشانه‌گذاری است، زبانی که می‌توان به کمک آن اسکلت یک صفحه‌ی وب را پیاده‌سازی کرد. زبان **HTML** در تاریخ ۱۹۹۱ توسط برنز لی روی اینترنت قرار گرفت و با ۱۸ عنصر اولیه توصیف شده بود که به آنها **HTML Tag** می‌گفتند. تاکنون ۵ نسخه‌ی اصلی از این زبان منتشر شده است.
- ▶ سوال مهم : تفاوت زبان برنامه نویسی و زبان تگ چیست؟؟؟

کاربرد html چیست ؟

HTML یک زبان نشانه‌گذاری است که برای ساختاردهی و نمایش محتوای صفحات وب استفاده می‌شود. از HTML برای ایجاد عناصر اصلی مانند عنوان‌ها، پاراگراف‌ها، لیست‌ها، لینک‌ها، تصاویر و فرم‌ها استفاده می‌شود. این زبان پایه‌اً اصلی طراحی وب است و به همراه CSS برای ظاهر و JavaScript برای تعامل، ساختار اولیه‌هُر سایت را تشکیل می‌دهد. بدون HTML، مرورگرها نمی‌توانند محتوای وب را به درستی نمایش دهند.



اولین کد html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
        content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
</head>
<body>
</body>
</html>
```

<!Doctype html> چیست؟

- ▶ این یک خط کد `html` به مرورگر نشان می‌دهد که وب از کدام نسخه `html` استفاده می‌کند.
- ▶ امروزه تقریباً از نسخه‌های دیگر `html` بجز نسخه ۵ استفاده نمی‌شود پس همین نوع را بد باشید کافی است.

تگ‌های `html`

- ▶ از خط دوم کد دیگر سر و کار ما با تگ‌های اصلی شروع می‌شود.
- ▶ تگ‌های استفاده شده در سند صفحه قبل هستند.`title,meta,head,body,Html`
- ▶ از بخش دوم فصل با تگ‌های `html` بصورت کامل آشنا خواهیم شد.

المان های html

صفحات html از المان های آن تشکیل شده اند. این المان ها ساختار صفحه را مشخص می کنند.



این تصویر یک المان تیتر را نشان میدهد که محتوا آن hello world است.
در ادامه راجب تیترها بطور کامل صحبت می کنیم.

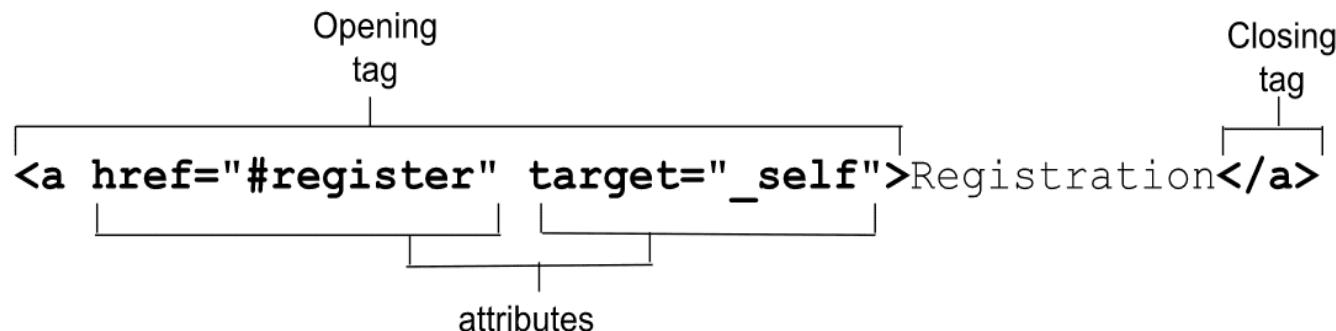
المان‌های void و یا self closing

► یک سری از المان‌ها نیازی به بستن ندارند و گویا خودشان ذاتاً بسته هستند. مثل یک خط افقی (`<hr>`) و یا شکستن خط و رفتن به خط بعدی (`
`) لیست زیر نسخه کامل این نوع المان‌ها است:

- `<area>`
- `<base>`
- `
`
- `<hr>`
- `<input>`
- `<link>`
- `<col>`
- ``
- `<source>`
- `<wbr>`
- `<track>`

Attribute (صفات)

- ▶ برای دادن یکسری از ویژگی‌ها به یک تگ لازم است که از attribute‌های متفاوت استفاده کنیم.
- ▶ هر صفت از دو بخش نام و مقدار تشکیل شده است.
- ▶ در تصویر زیر یک صفت کلاس با مقدار important به تگ p داده شده است.
- ▶ در فصل دوم CSS با اینکه کلاس چیست آشنا می‌شویم.
- ▶ صفت را فقط در تگ آغازین قرار می‌دهیم.
- ▶ کلاس important به تنها یک کاری انجام نمی‌دهد و نیاز به CSS داریم.



ساختار کد HTML

```
<div> index.html <x>
1  <!doctype html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport"
6          |      content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
7      <meta http-equiv="X-UA-Compatible" content="ie=edge">
8      <title>Document</title>
9  </head>
10 <body>
11
12 </body>
13 </html>
```

در ادامه این بخش با تک تک تگ های این تصویر آشنا میشویم.

ساختار html خانوادگی است !!!

- ▶ در یک سند html ساختار به این شکل در می‌آید که یکسری از المان‌ها فرزند المان دیگری هستند برای مثال در کد قبل تگ html والد (parent) تگ head فرزند (child) فرزند html تگ است.
- ▶ همین رابطه نیز بین تگ‌های html و body برقرار است اما رابطه تگ‌های body و head از نوع برادری یا خواهری (siblings) است.
- ▶ حال هر المانی که داخل تگ head و یا body قرار بگیرد تگ html جد (ancestor) آن خواهد بود، به عبارتی تگ html جد تگ meta است.
- ▶ در ادامه با تک تک المان‌ها آشنا می‌شویم.

المان های ساختاری: html

- ▶ این المان تمام المان های دیگر را در خود می گنجاند و محدوده‌ی کد HTML را مشخص می‌کند. به این المان، نیز می‌گویند. بدون استفاده از تگ html این امکان وجود دارد که مرورگر به درستی کد را تفسیر نکند و محتوای نمایش داده شده درست نباشد.
- ▶ عموما در این تگ از صفت dir استفاده می‌شود که مشخص می‌کند صفحه ما راست چین است یا چپ چین.
- ▶ dir="rtl" نشان دهنده راست چین بودن و dir="ltr" نشان دهنده چپ چین بودن است.
- ▶ صفت lang نیز برای نشان دادن زبان صفحه است.

المان‌های ساختاری: head

► المان **head** که فرزند المان **root** یا همان **html** است، دربردارنده‌ی تمام تگ‌هایی است که مستقیماً مربوط به نمایش محتوا روی صفحه نیستند. داخل این المان، تگ‌هایی قرار می‌گیرند که حاوی اطلاعاتی درباره‌ی صفحه‌ی وب هستند، در اینجا به بررسی مهم‌ترین این تگ‌ها می‌پردازیم:

- تگ **meta** : از این تگ برای ایجاد **metadata** های صفحه‌ی وب استفاده می‌شود. **meta** ها اطلاعاتی برای صفحه‌ی وب هستند که مستقیماً محتوای نمایش داده شده روی صفحه نیستند اما داده‌هایی در اختیار مرورگر و حتی سایر سایتها، نرم افزارها و ... درباره‌ی این وبسایت قرار می‌دهند. یکی از کاربردهای این تگ، تعریف **encoding** کarakترهای متنی برای مرورگر است که با دادن **attribute** ای به نام **charset** به صورت زیر انجام می‌شود:

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

- تگ **title** : این تگ عنوان صفحه را مشخص می‌کند که در قسمت تب مرورگر نمایش داده می‌شود. همچنین اگر صفحه‌ای را در مرورگر **bookmark** کنید، از این عنوان برای ذخیره‌ی آن استفاده می‌شود.

المان‌های ساختاری: body

- ▶ این المان شامل محتوایی است که در صفحه‌ی وب مشاهده می‌کنیم مانند متون، لینک‌ها، محتواهای چندرسانه‌ای و ... همان‌طور که گفته شد، اگر می‌خواهید محتوای چیزی که در صفحه‌ی وب می‌بینید را تغییر دهید، چیزی به آن اضافه یا از آن کم کنید، باید سراغ تگ‌های داخل المان **body** بروید. با المان‌هایی که داخل **body** قرار می‌گیرند به طور مفصل در بخش‌های بعدی آشنا می‌شویم.
- ▶ تگ‌هایی مثل **p**، **br**، **hr**، **a**، **h1** و ... در این بخش قرار می‌گیرند.
- ▶ در ادامه با تگ‌های متن که در این بخش نوشته می‌شوند آشنا خواهیم شد.

المان‌های عنوان (Heading elements)

- ▶ در html شش المان برای تیترها داریم.
- ▶ در هر سند html حداکثر یک تگ h1 باید قرار دهیم.
- ▶ عدد تگ برای نشون دادن اهمیت آن تیتر (عنوان) است نه سایز!!! منظور این است که تگ h1 نشان میدهد آن تیتر مهمترین تیتر کل سند است ، نه اینکه صرفا بزرگتر از بقیه تیترها است.
- ▶ پس تگ h1 نشان دهنده مهمترین تیتر و تگ h6 نشان دهنده کم اهمیت ترین تیتر است.

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

منظور از اهمیت چیست؟؟؟

- ▶ ۱. اهمیت از نظر موتورهای جستجو : موتورهای جستجو مثل google از عنوان‌های مهم وبسایت‌ها استفاده می‌کنند تا بتوانند موضوع صفحات را بفهمند و آنها را به کاربران نشان بدهند.(اصطلاحاً صفحات را index کنند)
- ▶ ۲. اهمیت از نظر جذب کاربر : کاربران نیز در نگاه اول جذب عنوان‌ها و محتواهای تصویری می‌شوند و درصورتی که از این تگ‌ها به درستی استفاده کنید، می‌توانید کاربران بیشتری را جذب کنید.
- ▶ تکلیف شماره ۱ : یک پاراگراف در مورد مفهوم index کردن تحقیق کنید.

المان پاراگراف

- ▶ این المان که با تگ `<p>your text</p>` نمایش داده میشود برای اضافه کردن یک خط متن جدید به سند است.
- ▶ این المان فاصله‌های اضافی و شکستن خط را حذف میکند.
- ▶ هربار که این تگ را استفاده کنید به خط بعد می‌رود.

```
<p>. این یک متن امتحانی است</p>  
ادامه متن تستی برای دوره با اضافه کردن فاصله و رفتن به خط بعدی  
فاصله و  
رفتن به خط بعدی  
</p>
```

این یک متن امتحانی است.

ادامه متن تستی برای دوره با اضافه کردن فاصله و رفتن به خط بعدی

ادامه المان پاراگراف

- ▶ برای آنکه داخل یک پاراگراف بتوانیم به خط بعدی برویم و چندین بار از تگ p استفاده نکنیم باید داخل تگ p از تگ
 void
 استفاده کنید.
- ▶ اگ بخواهیم به خط بعد نرویم و قسمتی از متن را انتخاب کنیم. باید آن قسمت از متن را داخل تگ span قرار بدھیم.
- ▶ کاربرد اصلی span برای موقعی است که میخواهیم به بخشی از متن استایل متفاوت از بقیه متن بدھیم.

```
<p> . است <span></span> امتحانی</p>  
<p>  
    شکستن خط<br> متن امتحانی برای  
</p>
```

این یک متن امتحانی است.

متن امتحانی برای
شکستن خط

تگ‌هایی که به متن استایل میدهند

- ▶ تگ **i** برای italic یا مورب کردن متن استفاده می‌شود.
- ▶ تگ **b** برای بولد کردن متن استفاده می‌شود.
- ▶ تگ **em** همانند **i** است با این تفاوت که کلمه مهمی در سایت به حساب خواهد آمد و برای موتورهای جستجو مهم خواهد بود.
- ▶ تگ **strong** همانند **b** است با این تفاوت که کلمه مهمی در سایت به حساب خواهد آمد.

تکلیف شماره ۲

- ▶ درمورد تگ های زیر تحقیق کنید:
- ▶ المان **blockquote** و ویژگی **cite** در آن به چه معناست؟
- ▶ درمورد تگ **time** و ویژگی **datetime** تحقیق کنید.
- ▶ المان **address** چه کاربردی دارد؟

المان نمایش زمان و تاریخ

▶ فرض کنید که می‌خواهیم متن ۲۰۲۵ February ۱۰ را نمایش دهیم. تمام حالات زیر معادل یا مشابه این تاریخ هستند:

```
<p>10 February 2025</p>
<p>10th February 2025</p>
<p>Feb 10 2025</p>
<p>2025 دهم فوریه</p>
```

▶ پس لازم است یک فرمتی را مشخص کرده و از آن استفاده کنیم:

```
<time datetime="2025-02-10T22:30">10.30pm, 10 February 2025</time>
<p>
  Event will start <time datetime="2025-03-12T15:30">Tomorrow at 3:30pm</time>
</p>
```

المان‌های نمایش نقل قول

- ▶ **blockquote**: برای نمایش یک نقل قول که از خط جدید آغاز شود از این المان استفاده می‌کنیم. از ویژگی **cite** برای مشخص کردن منبع دقیق نقل قول استفاده کرد. همچنین از المان **cite** می‌توان برای نمایش نام منبع بهره برد.
- ▶ دقت کنید که ویژگی **cite** نمایش داده نمی‌شود و آدرس منبع را به عنوان مقدار می‌پذیرد، در حالی که المان **blockquote** نام منبع را در خود جای می‌دهد و در صفحه نیز نمایش داده می‌شود، ضمناً این المان تنها مختص **blockquote** نیست و می‌توان خارج از این چارچوب هم از آن استفاده کرد.

```
<p>Here's a quotation:</p>
<blockquote cite="https://www.huxley.net/bnw/four.html">
  <p>
    Words can be like X-rays, if you use them properly—they'll go through
    anything. You read and you're pierced.
  </p>
</blockquote>
<p>
  —Aldous Huxley,
  <br />
  <cite>Brave New World</cite>
</p>
```

المان (short inline quotation) q

- از این المان برای نمایش نقل قول در همان خط به کار می‌رود. متن داخل آن بین علامت نقل قول باز و بسته (" ") قرار می‌گیرد و همچنین برای مشخص کردن منبع دقیق نقل قول می‌توان از ویژگی cite بهره برد.

```
<p>
  According to Mozilla's website,
  <q cite="https://www.mozilla.org/en-US/about/history/details/">
    Firefox 1.0 was released in 2004 and became a big success.
  </q>
</p>
```

According to Mozilla's website, “ Firefox 1.0 was released in 2004 and became a big success. ”

المان نمایش آدرس و اطلاعات تماس

- ▶ برای نمایش نشانی مکان و اطلاعات تماس مربوط به آن صفحه یا وبسایت، آنها را داخل المان address قرار می‌دهیم. مانند مثال زیر:

```
<address>
  <p>
    Azadi<br />
    Tehran<br />
    Tehran<br />
    Iran
  </p>

  <p>Tel: 01234 567 890</p>

  <p>Email: me@quera.org</p>
</address>
```

- ▶ نکته: فقط برای آدرس‌های مربوط به وبسایت خود استفاده کنید.

*Azadi
Tehran
Tehran
Iran*

Tel: 01234 567 890

Email: me@quera.org

ساخت لینک

- ▶ از تگ **a** برای ساخت لینک استفاده می‌کنیم. آنچه میان تگ باز و بسته قرار دارد چیزی است که کاربر می‌بیند و می‌تواند یک تکست یا یک المان دیگر باشد.
- ▶ از ویژگی **href** برای مشخص کردن مقصد لینک در اینجا (www.test.com) استفاده می‌کنیم. با کلیک روی محتوای لینک، کاربر به این مقصد منتقل می‌شود. در ادامه با انواع مقادیر این **attribute** نیز آشنا خواهید شد.

```
<a href="https://www.w3schools.com/"> تست 1 </a>  
<a href="https://www.w3schools.com/"><h2> تست 2 </h2></a>
```

تست 1

تست 2

- ▶ با استفاده از این تگ می‌توان به بخش‌های مختلف پروژه خود نیز رفت.

```
<a href="contact-us.html">صفحه‌ی تماس با ما</a>  
<a href="products/product1.html">صفحه‌ی کالا</a>  
<a href="products/details/product-detail.html">صفحه‌ی جزئیات کالا</a>  
<a href=".. /anotherSite/main.html">صفحه‌ی سایت دیگر</a>
```

تمرین

- ▶ درمورد آدرس مطلق جستجو کنید.
- ▶ درمورد استفاده از تگ لینک (a) برای دانلود فایل ها جستجو کنید و سعی کنید کدی برای دانلود یک فایل pdf از پروژه خود بنویسید.

لیست‌ها

- ▶ لیست‌ها یکی از ابزارهای پرکاربرد در طراحی صفحات وب هستند که برای نمایش مجموعه‌ای از عناصر مرتبط به کار می‌روند. در بسیاری از مواقع لازم است اطلاعات را به صورت منظم و قابل فهم برای کاربر ارائه کنیم؛ مانند فهرست گزینه‌های منوی سایت، مجموعه‌ای از وظایف قابل انجام، یا لیستی از لینک‌هایی که در بخش بالایی صفحه قرار می‌گیرند. استفاده از لیست‌ها باعث نظم دهی بهتر محتوا و بهبود تجربه کاربری در وبسایت می‌شود.
- ▶ لیست‌ها بصورت دو نوع ترتیبی و بدون ترتیب وجود دارند.

لیست بدون ترتیب (Unordered list)

- ▶ منظور از لیست‌های بدون ترتیب (نامرتب)، لیست‌هایی هستند که آیتم‌ها در آن‌ها، دارای ترتیب نمی‌باشند و در واقع ترتیب آن‌ها دارای اهمیت نیست.

```
<ul>
  <li>Benz</li>
  <li>BMW</li>
  <li>Pride</li>
  <li>Renault</li>
</ul>
```

- Benz
- BMW
- Pride
- Renault

- ▶ همان‌طور که در کد مشاهده می‌شود، برای ساخت یک لیست بدون ترتیب باید از تگ ul استفاده کنیم. سپس هر مورد از لیست را درون تگ li بنویسیم. به این ترتیب، مجموعه‌ای از تگ‌های li ادر داخل تگ ul قرار می‌گیرند و یک لیست بدون ترتیب را تشکیل می‌دهند.

لیست با ترتیب (ordered list)

- ▶ منظور از این نوع لیست ها این است که ترتیب مهم است و رتبه بندی داریم:

```
<ol type="1">
  <li>Benz</li>
  <li>BMW</li>
  <li>Pride</li>
  <li>Renault</li>
</ol>
```

- 1. Benz
- 2. BMW
- 3. Pride
- 4. Renault

- ▶ بعنوان تمرین سایر حالات ویژگی `type` را بررسی کنید همچنین ویژگی های `start`, `reversed` را هم بررسی کنید.

لیست‌های توضیحی (Description Lists)

- ▶ نوع دیگری از لیست‌ها وجود دارد که در آن هر آیتم یک عنوان و دارای توضیحی می‌باشد. در مثال زیر نمونه‌ای از این نوع لیست را می‌بینیم:

```
<dl>
  <dt>HTML</dt>
  <dd>HyperText Markup Language</dd>
  <dt>CSS</dt>
  <dd>Cascading Style Sheets</dd>
</dl>
```

- ▶ همان طور که در کد قابل مشاهده است، در این نوع لیست‌ها، کل لیست داخل تگ `dl` قرار دارد و عنوان هر آیتم در تگ `dt` و توضیح آن در تگ `dd` (Description Term) (Description Details) قرار می‌گیرد.

لیست‌های تودرتو (Nested Lists)

- ▶ تا کنون در تمام لیست‌های بالا، داخل المان `li` یا `dd` از متن استفاده کرده‌ایم. اما داخل این گونه المان‌ها که نمایش‌دهنده‌ی آیتم‌های لیست هستند می‌توان از سایر المان‌ها نیز استفاده کرد مانند لینک، عکس و
- ▶ به همین ترتیب می‌توان داخل آیتم‌های یک لیست، یک لیست دیگر قرار داد، به مثال زیر دقت کنید:

```
<ol type="I">
  <li>
    Chapter 1
    <ol type="a">
      <li>Lesson 1</li>
      <li>Lesson 2</li>
    </ol>
  </li>
  <li>
    Chapter 2
    <ol type="a">
      <li>Lesson 1</li>
      <li>Lesson 2</li>
      <li>Lesson 3</li>
    </ol>
  </li>
  <li>
    Chapter 3
    <ol type="a">
      <li>Lesson 1</li>
    </ol>
  </li>
</ol>
```

عکس‌ها در HTML

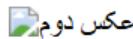
- ▶ برای عکس از تگ **img** استفاده می‌کنیم.
- ▶ ویژگی **src** برای مشخص کردن مکان عکس است که می‌تواند یک لینک از دیگر سایت‌ها و یا آدرسی لوکال باشد.
- ▶ ویژگی **alt** یک متن است که اگر به هر دلیلی عکس پیدا نشود آن را نشان میدهد.

```
  
  

```



Mani Fall



- ▶ همانطور که می‌بینید در تگ دوم به دلیل نامعتبر بودن آدرس، تصویر نشان داده نشد و صرفاً متن داخل **alt** را دیدیدم.

ویژگی‌های width , height

- ▶ دو ویژگی مهم دیگر که میتوان به تگ `img` داد طول و ارتفاع هستند که باعث میشوند عکس‌ها با ابعاد دلخواه ما در پروژه قرار بگیرند.

```

```

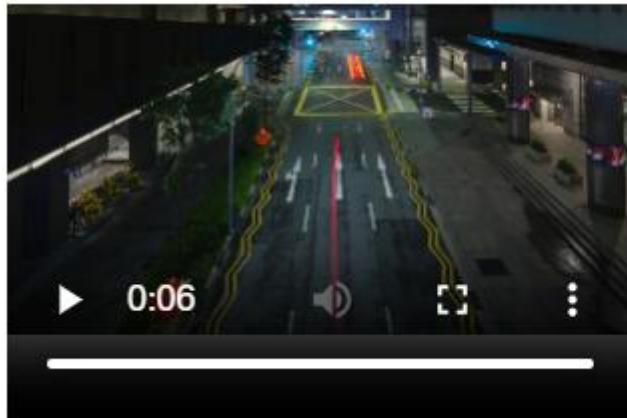
- ▶ نکته: برای مرتب نویسی تگ **img** توصیه شده که هر ویژگی در یک خط نوشته شود. مثل تصویر بالا.
- ▶ نکته ۲: استفاده از ویژگی **loading="lazy"** باعث می‌شود که مرورگر صفحه ما را زودتر بارگذارد.

نمایش ویدئو

- ▶ تگ باز و بسته **video** برای نمایش ویدئو استفاده میشود. هرآنچه ک درون تگ قرار بگیرد درصورتی نمایش داده میشود که مرورگر نتواند ویدئو را اجرا کند.
- ▶ برای مشخص کردن منبع ویدئو دو روش وجود دارد:
 ۱. استفاده از ویژگی **src**
 ۲. استفاده از تگ **source**
- ▶ ویژگی **controls** به ما نواری برای کم و زیاد کردن صدا و یا نگهداشتن و اجرا ویدئو میدهد.
- ▶ ویژگی **autoplay** باعث میشود ویدئو بصورت خودکار اجرا شود
- ▶ ویژگی **loop** باعث میشود ویدئو پس از اتمام دوباره اجرا شود.

مثال ویدئو

```
<video controls autoplay width="250" height="200">
  <source src="videos/street.mp4" type="video/mp4">
    !!! مرورگر قادر به اجرای ویدئو نیست
</video>
```

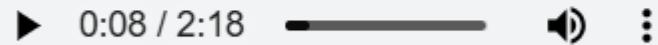


پخش صدا

▶ برای پخش صدا از تگ **audio** استفاده میکنیم. ویژگی های این تگ دقیقا مشابه **video** است.

```
<audio controls autoplay loop src="music/Things%20From%20The%20Past.mp3">  
    مرورگر قادر به اجرای فایل نیست  
</audio>
```

مثال برای پخش صدا



بخش دوم

- ▶ المان `div` و بخش‌بندی صفحه‌ی وب
- ▶ تفاوت المان‌های `block` و `inline`
- ▶ کار با جدول‌ها
- ▶ کار با وب‌فرم‌ها
- ▶ المان `embed` و `iframe` کردن صفحات وب
- ▶ استفاده از تگ‌های معنادار
- ▶ آشنایی با المان‌های مورد استفاده داخل المان `head`

بخش‌بندی صفحات با `div`

- ▶ یکی از پرکاربردترین المان‌ها در طراحی صفحات وب، المان `div` است. نام این المان از واژه‌ی `division` به معنای «بخش» یا «قسمت» گرفته شده است. این تگ به صورت پیش‌فرض ظاهر خاصی در صفحه ندارد، اما برای تقسیم‌بندی و سازمان‌دهی محتوا درون تگ `body` مورد استفاده قرار می‌گیرد.
- ▶ به مثال اسلاید بعد دقت کنید...

مثال حالت بدون div

در این کد تقریبا مشخص است که صفحه چند بخش مختلف دارد. ولی مشخص کردن دقیق تعداد بخش‌ها کمی دشوار است.

برای اینکه بخش‌بندی به درستی انجام شود لازم است از تگ **div** استفاده کنیم.

توجه داشته باشید که کد ما بسیار کوچک است ولی باز هم بدون دقت به محتوا نمی‌توان بخش‌های مختلف صفحه را حدس زد.

در اسلاید بعدی نسخه بخش‌بندی شده این کد را می‌بینیم...

```
<h1>/دوره‌ی آموزش توسعه‌ی وب</h1>
<h2>/توضیحاتی در مورد دوره</h2>
<p>
    یک متن خیلی خیلی طولانی در مورد
    <em>/دوره‌ی آموزش توسعه‌ی وب</em>
</p>
<p>
    بخش دوم توضیحات درباره‌ی
    <strong>/دوره‌ی آموزش توسعه‌ی وب</strong>
</p>

<a href="#">/صفحه اصلی</a>
<h2>/ویژگی‌های دوره</h2>
<p>
    یک متن
    <span>/خیلی</span>
    طولانی در مورد ویژگی‌های دوره
</p>
```

نسخه بخش‌بندی شده با div

```
<h1></h1>دوره‌ی آموزش توسعه‌ی وب

<div>
  <h2>توضیحاتی در مورد دوره</h2>
  <p>
    یک متن خیلی خیلی طولانی در مورد
    <em>دوره وеб</em>
  </p>
  <p>
    بخش دوم توضیحات درباره
    <strong>دوره وеб</strong>
  </p>
  <a href="#">صفحه اصلی</a>
</div>

<div>
  <h2>ویژگی‌های دوره</h2>
  <p>
    یک متن
    <span>خیلی</span>
    طولانی در مورد ویژگی‌های دوره
  </p>
</div>
```

المان‌های inline

- ▶ المان‌های HTML بر اساس نحوه نمایش پیش‌فرض، به دو دسته‌ی کلی inline و block تقسیم می‌شوند. در این چند اسلاید با این دسته‌بندی آشنا خواهیم شد.
- ▶ المان‌های inline : این المان‌ها بصورت پیش‌فرض در همان خط قبلی قرار می‌گیرند برای مثال تگ span

```
<p>this is a test for <span>inline</span> elements</p>
```

this is a test for inline elements

- ▶ همانطور که در خروجی قابل مشاهده است تگ span به خط بعدی نرفته و صرفا به اندازه عرض محتوا فضا گرفته است.
- ▶ بیشترین استفاده از تگ span زمانی است که بخواهیم به بخشی از متن استایل خاصی بدهیم.
- ▶ خود تگ span استایل خاصی ندارد.

المان های block

- ▶ در طرف مقابل المان های **block** قرار دارند. این گونه المان ها برای نمایش به خط جدید رفته و همچنین به صورت افقی تمام فضایی که از **container** خود در اختیار دارند را اشغال می کنند. ارتفاع المان های **block** به اندازه ای ارتفاع محتوای آنها خواهد بود. برای مثال المان **p** از نوع **block** است.

```
<p>this is a test</p>
<p>for block elements</p>
```

this is a test

for block elements

جدول‌ها

- ▶ یکی از مهم‌ترین مباحث در HTML، نمایش اطلاعات در قالب جدول می‌باشد. با استفاده از جدول می‌توان داده‌ها را به صورت منظم در سطر و ستون‌هایی نمایش داد. احتمالاً در وبسایت‌های مختلف جداول زیادی پیدا کنید. برای مثال برنامه هفتگی را می‌توان با جدول نوشت.
- ▶ برای ایجاد جدول باید از تگ اصلی **table** استفاده کرد و درون آن با استفاده از تگ‌های **tr, th, td** ساختار و محتوای جدول را مشخص کرد.

```
<table>
  <tr>
    <th>Item</th>
    <th>Count</th>
  </tr>
  <tr>
    <td>Mango</td>
    <td>125</td>
  </tr>
  <tr>
    <td>Orange</td>
    <td>75</td>
  </tr>
</table>
```

Item	Count
Mango	125
Orange	75

المان‌های جدول

- ▶ المان **table** : ابتدا با استفاده از تگ‌های باز و بسته‌ی **table** جدول را تعریف می‌کنیم.
- ▶ المان **tr (Table Row)** : برای ایجاد سطر، ردیف یا **row** در جدول از المان **tr** استفاده می‌کنیم.
- ▶ المان **td (Table Data Cell)** : برای مشخص کردن داده‌های هر سطر از المان **td** استفاده می‌کنیم. محتوای این المان می‌تواند هر چیز دلخواهی باشد. همان‌طور که در نمونه کد اسلاید قبل مشاهده کردید، این المان فرزند **tr** محسوب می‌شود.
- ▶ المان **th (Table Header)** : داده‌های سطر اول جدول معمولاً مشخص‌کننده‌ی نوع داده‌های هر ستون هستند در این مثال **Count** و **Item** به همین دلیل این داده‌ها را به جای **td** داخل **th** که برای مشخص کردن **header** هستند قرار می‌دهیم. به طور پیش‌فرض متنی که داخل تگ باز و بسته‌ی **th** قرار بگیرد، به صورت **bold** نمایش داده خواهد شد. استفاده از چنین المانی اجباری نیست اما به کار گیری آن توصیه می‌شود. همان‌طور که در نمونه کد اسلاید قبل مشاهده کردید، این المان فرزند **tr** محسوب می‌شود.

ویژگی border

► این ویژگی باعث می‌شود یک جدا کننده در بین خانه‌های جدول ایجاد شود. یک عدد می‌گیرد که ضخامت جداکننده را نشان می‌دهد

```
<table border="1">
  <tr>
    <th>Item</th>
    <th>Count</th>
  </tr>
  <tr>
    <td>Mango</td>
    <td>125</td>
  </tr>
  <tr>
    <td>Orange</td>
    <td>75</td>
  </tr>
</table>
```

Item	Count
Mango	125
Orange	75

حل مشکل جدول قبل

- ▶ همانطور که در اسلاید قبل دیدید ویژگی **border** برای ما خطوط حاشیه و جداکننده عناصر جدول را اضافه کرد ولی ظاهر مطلوبی ندارد برای حل مشکل این جدول یک استایل به آن می‌دهیم.

```
<table border="1" style="border-collapse: collapse">
  <tr>
    <th>Item</th>
    <th>Count</th>
  </tr>
  <tr>
    <td>Mango</td>
    <td>125</td>
  </tr>
  <tr>
    <td>Orange</td>
    <td>75</td>
  </tr>
</table>
```

Item	Count
Mango	125
Orange	75

ویژگی colspan

► به کمک ویژگی colspan می‌توان تعداد ستون‌هایی که یک خانه از جدول اشغال می‌کند را تغییر داد، به مثال زیر دقت کنید:

```
<table border="1" style="border-collapse: collapse">
  <tr>
    <th>Item</th>
    <th>Count</th>
    <th>Value</th>
  </tr>
  <tr>
    <td>Mango</td>
    <td>125</td>
    <td>100</td>
  </tr>
  <tr>
    <td>Orange</td>
    <td>75</td>
    <td>50</td>
  </tr>
  <tr>
    <td colspan="2">Total</td>
    <td>150</td>
  </tr>
</table>
```

Item	Count	Value
Mango	125	100
Orange	75	50
Total		150

ویژگی rowspan

► به کمک ویژگی rowspan می‌توان تعداد سطرهایی که یک خانه از جدول اشغال می‌کند را تغییر داد، به مثال زیر دقت کنید:

```
<table border="1" style="border-collapse: collapse">
  <tr>
    <th>Name</th>
    <th>Course</th>
    <th>Grade</th>
  </tr>
  <tr>
    <td rowspan="3">Mani</td>
    <td>English</td>
    <td>B</td>
  </tr>
  <tr>
    <td>Math</td>
    <td>A</td>
  </tr>
  <tr>
    <td>science</td>
    <td>C</td>
  </tr>
</table>
```

Name	Course	Grade
Mani	English	B
	Math	A
	science	C

المان captino

▶ به کمک المان **caption** می‌توان توضیح مختصری (عنوان) به بالای یک جدول اضافه کرد.

```
<table border="1" style="border-collapse: collapse">
  <caption>this is Mani's Scores</caption>
  <tr>
    <th>Name</th>
    <th>Course</th>
    <th>Grade</th>
  </tr>
  <tr>
    <td rowspan="2">Mani</td>
    <td>English</td>
    <td>B</td>
  </tr>
  <tr>
    <td>Math</td>
    <td>A</td>
  </tr>
</table>
```

this is Mani's Scores

Name	Course	Grade
Mani	English	B
	Math	A

ساختاردهی به جدول

- ▶ با افزایش تعداد سطراها و دادههای جدول نیاز است تا از نظر محتوایی، ساختاری برای جدول ایجاد کنیم. معمولاً جدولها از سه بخش هدر **header** ، محتوا **body** و فوتر **footer** تشکیل می‌شوند. برای مثال به جدول زیر توجه کنید.
- ▶ برای بخش هدر از تگ **thead** استفاده میکنیم و هدر را در آن قرار می‌دهیم.
- ▶ برای بخش بدن از تگ **tbody** استفاده میکنیم که برای محتوا اصلی جدول است.
- ▶ برای بخش پایینی و سطر اخر از **tfooter** استفاده می‌کنیم.

مثال جدول

```
<table border="1" style="border-collapse: collapse">
  <thead>
    <tr>
      <th>id</th>
      <th>product</th>
      <th>count</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>1</td>
      <td>sponge</td>
      <td>2</td>
    </tr>
    <tr>
      <td>2</td>
      <td>Apple</td>
      <td>2</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td colspan="2" style="text-align: right;">Total
```

id	product	count
1	sponge	2
2	Apple	2
Total		6

تمرین

جدول زیر را پیاده‌سازی کنید. ►

برنامه زمانی

روز					ساعت
چهارشنبه	پنجشنبه	دوشنبه	سه شنبه	شنبه	
زبان	ریاضی	علوم	تاریخ	ریاضی	9:۰۰ - ۸:۰۰
ریاضی	زبان	ریاضی	علوم	تاریخ	1۰:۰۰ - ۹:۰۰
تاریخ	زبان	ریاضی	علوم	تاریخ	1۱:۰۰ - ۱۰:۰۰
علوم	ریاضی	زبان	تاریخ	ریاضی	1۲:۰۰ - ۱۱:۰۰

تمرین

▶ جدول زیر را پیاده‌سازی کنید. (محتوای دلخواه بدید ولی ساختار دقیق همین)

Name	Subject	Marks
Mark Smith	English	67
	Maths	82
	Science	91
David Johnson	English	80
	Maths	67
	Science	43
power of god tehrani	English	20
	Maths	77
	Science	46
kevin LA	English	100
	Maths	100
	Science	100

المان‌های فرم

▶ در اکثر وبسایت‌ها می‌توانید فرم‌های مختلفی را مشاهده کنید. از فرم‌ها برای دریافت اطلاعات از کاربر استفاده می‌شود. برای مثال دریافت اطلاعات کاربر برای ثبت‌نام یا دریافت نظر کاربر درباره‌ی یک پست و بلاگ و ... همگی به کمک فرم انجام می‌شود. نمونه‌ای از فرم‌ها را می‌توانید در تصاویر زیر مشاهده کنید:

CREATE YOUR ACCOUNT
This is step 1

Email

Password

Confirm Password

Next

المان form

▶ اولین قدم برای ایجاد یک وبفرم استفاده از المان **form** است. این المان به خودی خود نمود ظاهری خاصی ندارد اما دربردارندهٔ سایر المان‌های مربوط به فرم مانند ورودی‌های متنی، دکمه‌ها و ... است. همچنین این المان خواصی برای نحوهٔ ارسال اطلاعات به سرور دارد اما همان طور که در بالا به آن اشاره کردیم، این موارد کمی از سطح این دوره بالاتر هستند. درون این تگ می‌توان از هر تگ دیگری مثل **p** ، **span** و ... استفاده کرد.

▶ یک نمونه فرم بسیار ساده:

```
<form>
  <span>name</span>
  <br>
  <input type="text">
  <br>
  <span>family</span>
  <br>
  <input type="text">
  <br>
  <button type="submit">send</button>
</form>
```

The image shows a screenshot of a web browser displaying a simple form. The form consists of two text input fields and one button. The first input field is labeled "name" and the second is labeled "family". Both fields are currently empty. Below the inputs is a blue "send" button.

المان input

اصلی ترین المان برای دریافت اطلاعات از کاربر المان **input** است. به کمک این المان می‌توان انواع مختلفی از ورودی‌ها از جمله ورودی متنی، عدد، گزینه‌ای، فایل، تاریخ و ... را از کاربر دریافت کرد. مهمترین ویژگی این المان است که مشخص کننده نوع اطلاعاتی است که قرار است این المان قرار است دریافت کند.

► مهمترین مقادیر ویژگی **: type**

.1 **Text** : مشخص می‌کند ورودی **input** از نوع متنی است که حالت پیش‌فرض هم است.

.2 **Email** : مشخص می‌کند کاربر باید ایمیل خود را در این بخش وارد کند. درصورتیکه کاربر ایمیل خود را درست وارد نکند مرورگر به او اخطار خواهد داد.

.3 **number** : برای موقعی است که میخواهیم از کاربر مقادیر عددی بگیریم. مثل کد ملی یا کد امنیتی ورود به سایت.

.4 **Password** : برای دریافت پسورد است که ورودی را نشان نمی‌دهد.

► ویژگی **required** : این ویژگی باعث می‌شود که کاربر مجبور شود این **input** را پر کند و نمی‌تواند خالی باشد، به قولی فیلد اجباری خواهد بود.

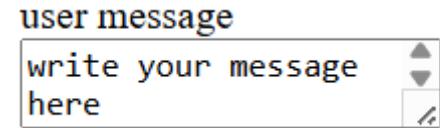
► ویژگی **name** : به کمک این ویژگی می‌توان به هر داده‌ای که از کاربر دریافت می‌کنیم، یک نام یا عنوان اختصاص دهیم. این نام همراه با مقدار داده پس از ثبت فرم به سرور ارسال خواهد شد.

```
<input name="user_name" type="text" id="name" required>
```

المان textarea

- ▶ همانند تگ **input** است با این تفاوت که برای دریافت متن های طولانی (مثل توضیحات) استفاده میشود.
- ▶ برعکس **input** تگ باز و بسته است.
- ▶ دو ویژگی **col** , **row** مهمترین ویژگی های این المان هستند که برای مشخص کردن ابعاد متن ورودی بکار میروند. و کاربر را محدود می کند.
- ▶ اگر متنی بین تگ باز و بسته آن نوشته شود عنوان متن پیشفرض به کاربر نشان داده می شود.

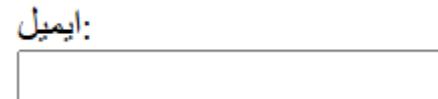
```
<textarea cols="20" id="msg"
           name="user_message" required rows="2">
    write your message here
</textarea>
```



المان label

- ▶ هنگام ایجاد فرمها، بهتر است تا برای المان‌هایی که مربوط به دریافت داده از کاربر هستند، یک عنوان یا برجسب مشخص کنیم تا کاربر بداند که چه داده‌ای را در آن المان وارد کند. مثلاً برای **input** مربوط به **email** به شکل زیر **label** تعریف کردہایم:

```
<label for="mail">ایمیل</label>
<br />
<input id="mail" type="email" name="user_email" />
```



- ▶ مهم است که در ویژگی **for** مقدار **id** تگ **input** را قرار دهیم، اینگونه برای مرورگر مشخص می‌شود که این لیبل مربوط به کدام **input** است.
- ▶ انجام این کار برای کاربرانی که از **screen reader** استفاده می‌کنند حیاتی است تا به درستی محتوای فرم برای آن‌ها خوانده شود.

المان button

- ▶ به کمک المان **button** می‌توان برای فرم، دکمه‌هایی به منظور انجام کارهای مختلف قرار داد. عملکرد این دکمه به مقدار ویژگی **type** آن بستگی دارد.
- ▶ ویژگی **type** برای **button** سه مقدار دریافت می‌کند:
- ▶ مقدار **submit** : اگر دکمه‌ای داخل یک **form** این مقدار **type** را داشته باشد، با کلیک روی آن فرم **submit** می‌شود، در واقع مرورگر تلاش می‌کند مقدار آن را برای سرور ارسال کند. از آن جایی که در مثال بالا آدرس سرور، نوع فرستادن اطلاعات و ... مشخص نشده است، با کلیک روی دکمه‌ی «ثبت» صرفاً صفحه‌ی مرورگر یک بار **refresh** می‌شود.
- ▶ مقدار **reset** : با کلیک روی دکمه‌ای که مقدار **type** آن **reset** باشد، مقادیر موجود در المان‌های ورودی در فرم، به حالت اولیه برمی‌گردند. در مثال بالا با کلیک روی دکمه‌ی «بازنشانی» مقادیر موجود در ورودی‌ها پاک می‌شوند.
- ▶ مقدار **button** : این مقدار عملکرد پیش‌فرضی ندارد و عملکرد دکمه از طریق **JavaScript** تعیین می‌شود.

```
<button type="submit">submit</button>

<button type="reset">reset</button>

<button type="button">do anything</button>
```

چند نوع دیگر input

نوع color ►

این نوع از `input` این امکان را به کاربر می‌دهد تا رنگ مورد نظر خود را از طریق **Color Picker**، انتخاب کند.
دقت کنید که ظاهر المان `input` با دادن این `type` از حالت نوشتمن متن کاملاً تغییر می‌کند.

نوع date ►

به کمک این نوع `input` می‌توان از کاربر یک تاریخ مشخص دریافت کرد. ظاهر این نوع از المان `input` مناسب وارد کردن یک تاریخ است و همچنین کاربر می‌تواند با استفاده از یک **Date Picker** به راحتی تاریخ مورد نظر خود را انتخاب کند.

نوع file ►

برای فراهم کردن امکان انتخاب فایل برای آپلود، از نوع `file` المان `input` استفاده می‌کنیم. ظاهر این نوع `input` مانند دکمه‌ای است که با کلیک بر روی آن، فضای **File Picker** برای کاربر باز می‌شود. همچنین به کمک `accept` می‌توان فرمات فایل‌های قابل قبول را مشخص کرد تا کاربر فایل غیرمجاز انتخاب نکند.

Select box

► در وب فرم‌ها رایج است که برای انتخاب از میان لیستی از گزینه‌ها، از **dropdown** استفاده کنیم. یکی از راه‌های اصلی ایجاد **dropdown** استفاده از المان **select** است. به مثال زیر دقت کنید

```
<label for="fruits"> fruits</label>
<select name="fruits" id="fruits">
  <option value="1" >Apple</option>
  <option value="2" selected>Banana</option>
  <option value="3">Cherry</option>
</select>
```

► همان طور که در کد قابل مشاهده است، برای استفاده از المان **:select**

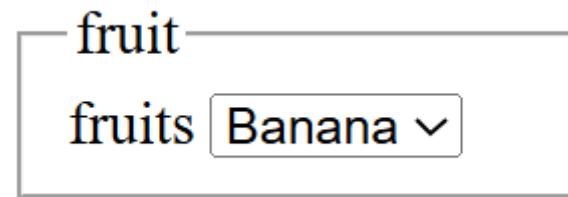
۱. گزینه‌ها را به کمک المان **option** تعریف می‌کنیم، مقداری که به کاربر نمایش می‌دهیم را میان تگ باز و بسته‌ی **option** قرار می‌دهیم. همچنین مقداری که می‌خواهیم پس از ثبت فرم به سرور ارسال کنیم را به ویژگی **value** می‌دهیم.

۲. المان‌های **option** تعریف شده را داخل المان **select** قرار می‌دهیم.

المان fieldset

- ▶ به کمک المان **fieldset** می‌توان چند المان ورودی که در یک گروه خاص قرار می‌گیرند را دسته بندی کرد. اطراف المان‌هایی که داخل یک **fieldset** قرار می‌گیرند، یک حاشیه‌ی جداگانده کشیده می‌شود.
- ▶ برای دادن نامی مشخص به هر بخش در اول آن از المان **legend** استفاده می‌کنیم.

```
<fieldset>
  <legend>fruit</legend>
  <label for="fruits"> fruits</label>
  <select name="fruits" id="fruits">
    <option value="1" >Apple</option>
    <option value="2" selected>Banana</option>
    <option value="3">Cherry</option>
  </select>
</fieldset>
```



datalist

► قبلتر با `select` برای ایجاد `dropdown` آشنا شدید. به کمک آن المان می‌توان از میان چند گزینه، یکی را انتخاب کرد. حال اگر بخواهیم تنها به کاربر چند گزینه پیشنهاد بدهیم اما کاربر بتواند ورودی دلخواه خود را نیز وارد کند، می‌توان از المان `datalist` در کنار یک المان `input` استفاده کنیم.

```
<label for="city_choice">city</label>
<br>
<input list="city_options" name="city_choice" id="city_choice">
<datalist id="city_options">
  <option value="tehran">tehran</option>
  <option value="tabriz">tabriz</option>
  <option value="rasht">rasht</option>
  <option value="shiraz">shiraz</option>
  <option value="isfahan">isfahan</option>
</datalist>
```



المان iframe

▶ با استفاده از المان **iframe** می‌توان یک صفحه‌ی وب (HTML) دیگر را در وbsایت خود قرار داد یا اصطلاحاً **embed** کرد. برای استفاده از این المان به صورت زیر عمل می‌کنیم:

```
<iframe  
    id="inlineFrameExample"  
    title="Inline Frame Example"  
    width="300"  
    height="200"  
    src="https://www.openstreetmap.org/export/  
/>
```



پایان فصل اول