

# شروع کار با CSS

## بخش اول: اصول اولیه CSS

---

سید کاوه احمدی

### اجزای یک صفحه وب

■ یک صفحه وب می‌تواند به سه لایه تقسیم شود:

- لایه **Content** (محتوا): مشخص کردن المان‌های موجود در صفحه وب (و محل قرارگیری آنها) که بوسیله **HTML** ایجاد می‌شود.
- لایه **Presentation** (نمایش): نحوه نمایش المان‌ها و خصوصیات ظاهری صفحه که بوسیله **CSS** ایجاد می‌شود (خوشگلازیسیون).
- لایه **Behavior** (رفتار): برای واکنش نشان دادن به رویدادهای کاربر که بوسیله **JavaScript** ایجاد می‌شود.

## CSS چیست؟

- **Cascading Style Sheets** یا همان **CSS** تکنیکی است که برای کنترل لایه نمایش در یک صفحه وب استفاده می‌شود.
  - برگه‌های سبک آشنایی
- سبک (**Style**) تعریف می‌کند که عناصر **HTML** چگونه نمایش داده شوند.

## CSS چیست؟

- هدف اصلی استفاده از **CSS** برای مشخص کردن ظاهر صفحه، جدا کردن قالب و فرمت صفحه از محتویات آن است.
- می‌توان تمامی فرمت‌های مد نظر برای المان‌های درون صفحات متعدد وب را در یک فایل **CSS** نگهداری کرد.
  - با **CSS** کنترل بیشتری بر روی ظاهر تگ‌های **HTML** خواهید داشت.
  - دلیل برگه خوانده شدن **CSS** همین است: می‌توانیم یک برگه از سبک‌ها داشته باشیم.

## Style ها یک مشکل رایج را حل می کنند:

- در HTML 1.0، تگ های HTML فقط برای تعریف اجزای یک صفحه یا document طراحی شدند.
- فرض بر این بود که وظیفه آرایش صفحات بر عهده مرورگر (Browser) است
  - مرورگر هر تگ را با یک فرمت از پیش مشخص شده نمایش می داد.
  - بدون استفاده از هرگونه دستور قالب بندی.
  - قالب بندی اصولاً معنایی نداشت!

## Style ها یک مشکل رایج را حل می کنند:

- در HTML 3.0 و با اضافه شده تگ هایی مثل `<font>` و یا صفاتی مثل `color` به HTML، طراحی وب سایت به کار سختی مبدل گشت چرا که محتوای صفحات HTML به طور کامل با آرایش صفحه مخلوط شده بود.
  - فرض کنید چندین پاراگراف داشته باشید که باید رنگ و فونت مشابه داشته باشند:
    - کدهای تعریف رنگ و فونت باید برای هر پاراگراف تکرار شود.
    - فرض کنید بخواهید رنگ پاراگراف ها را تغییر دهید: چندین تغییر (و احتمالاً در چندین صفحه) مورد نیاز است.

## Style ها یک مشکل رایج را حل می کنند:

- برای حل این مشکل کنسرسیوم وب (W3C)، استفاده از استایل ها را به همراه HTML 4.0 ایجاد کرد.
- Style Sheet ها می توانند در کارهای زیادی صرفه جویی کنند:
  - استایل چگونگی نمایش اجزای HTML را تعریف می کند دقیقا مثل تگ font و یا صفت color در HTML 3.2.
  - معمولا استایل ها در فایل های خارجی با پسوند CSS ذخیره می شوند. فایل های استایل خارجی به شما این امکان را می دهند که آرایش تمام صفحات وب خود را تنها با ویرایش یک فایل CSS انجام دهید.

## Style ها یک مشکل رایج را حل می کنند:

- CSS یک پیشرفت بزرگ در HTML محسوب می شود چراکه به طراحان و توسعه دهندگان این امکان را می دهد که آرایش و طرح بندی چندین صفحه را به صورت همزمان انجام دهند. به عنوان یک طراح شما می توانید برای هر عنصر HTML یک سبک (Style) تعریف کنید و آن را به هر تعداد صفحه که بخواهید به کار ببرید و برای یک تغییر کلی کافی است که سبک (Style) را تغییر دهید و تمامی عناصر و صفحات به صورت خودکار به روزرسانی خواهند شد.

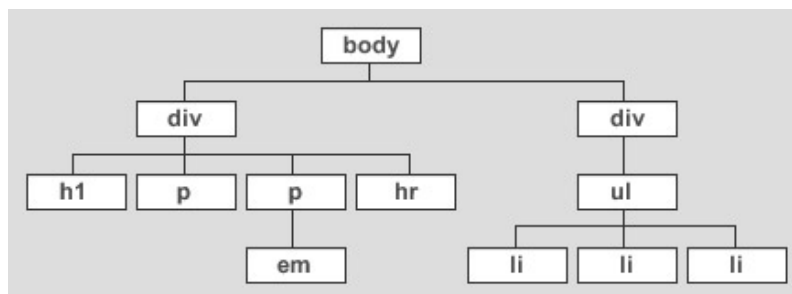
# موقعیت عناصر در یک سند HTML

- نحوه چینش عناصر تمامی سندهای HTML به صورت درختی قابل تحلیل است.
  - می توان روابط بین این عناصر را با روابط انسانی مقایسه نمود
- روابط زیر که بین المان های یک صفحه وجود دارد، می تواند کمک بزرگی به درک بهتر برخی مفاهیم در CSS کند.
  - Ancestor (جد)
  - Descendants (اولاد)
  - Parent (پدر)
  - Childs (فرزندان)
  - Siblings (برادران)

# موقعیت عناصر در یک سند HTML

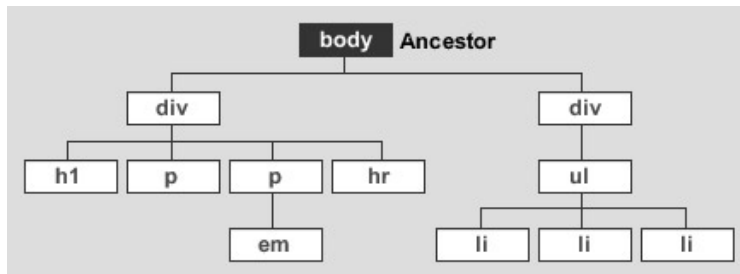
- نمودار درختی عناصر

```
<body>
  <div id="content">
    <h1>Heading here</h1>
    <p>Lorem ipsum dolor sit amet.</p>
    <p>Lorem ipsum dolor <em>sit</em> amet.</p>
    <hr />
  </div>
  <div id="nav">
    <ul>
      <li>item 1</li>
      <li>item 2</li>
      <li>item 3</li>
    </ul>
  </div>
</body>
```



## موقعیت عناصر در یک سند HTML

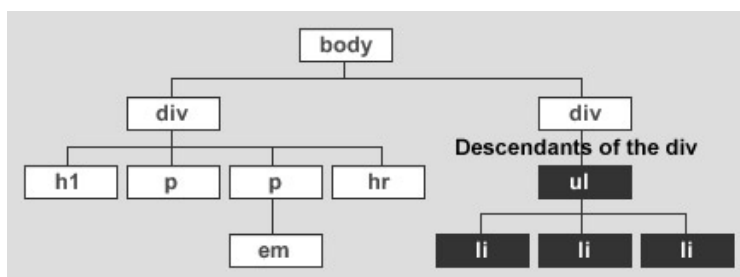
- **جد (Ancestor):** اشاره دارد به عنصری که در بالای یک سطح از درخت قرار می‌گیرد.



عنصر **body** جد تمامی عناصر موجود یک سند HTML است

## موقعیت عناصر در یک سند HTML

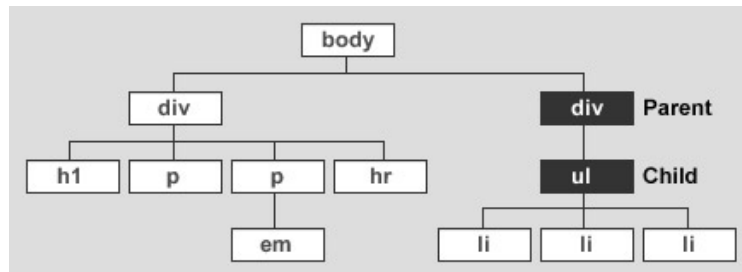
- **اولاد - نسل (Descendants):** تمامی عناصری که در درخت در زیر مجموعه یک عنصر قرار می‌گیرند، نسل آن عنصر هستند.



اولاد div

## موقعیت عناصر در یک سند HTML

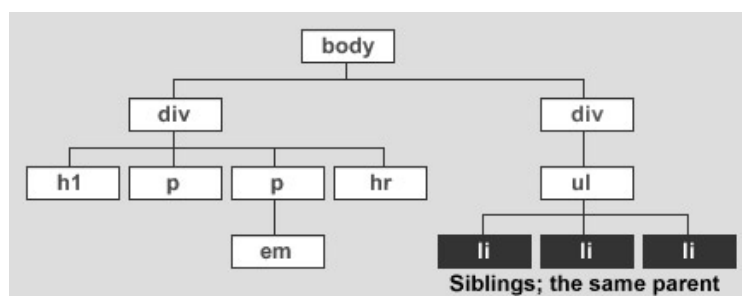
- **والد (Parent):** اشاره به عنصر بالایی و متصل در درخت.
- **فرزندان (Childs):** اشاره به عنصر پایینی و متصل در درخت.



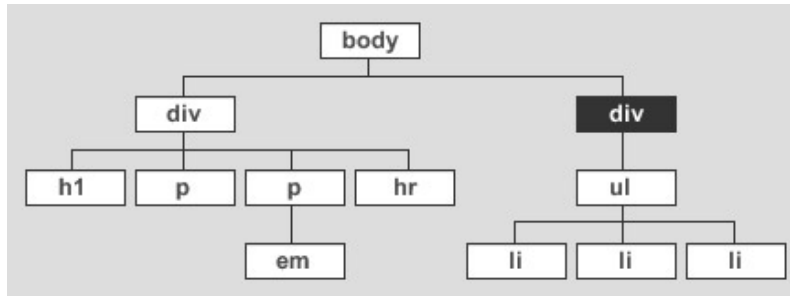
گره والد و فرزند آن

## موقعیت عناصر در یک سند HTML

- **خواهر و برادر (Siblings):** عناصری که والد مشترکی دارند (بدیهی است که باید در یک سطح باشند).



## موقعیت عناصر در یک سند HTML



▪ فرزند `body`

▪ والد `ul`

▪ جد `ul` و `li`ها ( `ul` و `li`ها اولاد او هستند)

▪ برادر دیگر عنصر `div`

## موقعیت عناصر در یک سند HTML

▪ یک المان خصوصیاتی را از جد و والد خود به شکل آبشاری به ارث می برد.

– دلیل آبشاری خوانده شدن سبکها در CSS همین مورد است!



## چگونه دستورات CSS را به صفحات خود اضافه کنیم؟

### ■ استایل شیت‌های خارجی (External Style Sheet):

– در این روش تمامی استایل‌ها و تعاریف نمایشی درون فایلی جداگانه قرار گرفته و بسیار مناسب حالت‌هایی است که قرار است که استایلی به بیش از یک صفحه HTML اعمال شود.

– اتصال به فایل‌های استایل شیت خارجی

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

– دلیل برگه خوانده شدن CSS همین است: می‌توانیم یک برگه از سبک‌ها داشته باشیم.

## چگونه دستورات CSS را به صفحات خود اضافه کنیم؟

### ■ استایل شیت‌های داخلی (Internal Style Sheet):

– کدهای حاوی تعریف استایل‌ها درون `<style>` قرار می‌گیرد و معمولاً نتیجه درون بخش `head` صفحه HTML گذاشته می‌شود.

```
<style type="text/css">
  h2 {color: red}
  p {margin-left: 20px}
</style>
```

– کاربرد این روش در حالت‌هایی است که صفحه HTML نیازمند تعریف استایلی منحصر به فرد و مجزا بوده و استایل طراحی شده فقط به درد آن صفحه خواهد خورد.

## چگونه دستورات CSS را به صفحات خود اضافه کنیم؟

### ▪ استایل شیت‌های درجا (Inline Styles) :

– استفاده از خصوصیت `style` برای تگ‌ها و مشخص کردن استایل‌های مورد نظر برای آن تگ.

```
<p style="color: red; margin-left: 20px">  
  This is a paragraph  
</p>
```

– عدم استفاده از این روش به شدت توصیه می‌شود.

- حجم صفحات را افزایش می‌دهد.
- کدها را به هم ریخته و ناخوانا می‌کند و ویرایش آنها را مشکل
- در کل بسیاری از دلایلی که به خاطر آنها CSS معرفی شد را زیر سوال می‌برد!

## دستورات CSS

### ▪ یک دستور CSS:

– یک قانون

– مجموعه‌ای از قواعد

## دستورات CSS

- یک قانون با کاراکتر «@» شروع و به «i» ختم می‌شود (این دستور را از سایر دستورات جدا می‌کند).

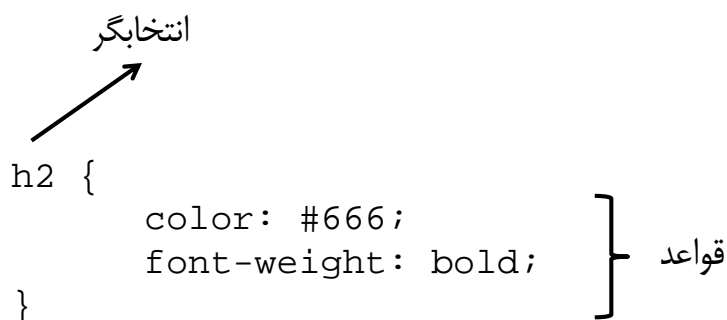
```
@import url(base.css);
```

## دستورات CSS

- یک دستور می‌تواند مجموعه‌ای از قواعد باشد که مابین کاراکترهای { و } قرار می‌گیرند.

- این دستورات به تمامی عناصری که توسط انتخابگر (Selector) دستور مشخص می‌شود اعمال می‌شوند.

انتخابگر



```
h2 {  
    color: #666;  
    font-weight: bold;  
}
```

قواعد

- یک قاعده از دو بخش تشکیل می شود:
  - یک خاصیت و یک مقدار.
- قواعد با استفاده از «`:`» از هم جدا می شوند.

```
property: value;
```