

## CSS چیست و چرا استفاده می‌شود؟

CSS به تنهایی معنا ندارد و همیشه در کنار HTML عمل می‌کند. HTML ساختار و محتوای صفحه را تعریف می‌کند و CSS مسئول تعیین ظاهر، چیدمان و نحوه نمایش این محتوا است.

در فرآیند رندر شدن صفحه در مرورگر، کد HTML به یک مدل ساختاری به نام DOM تبدیل می‌شود. DOM به صورت یک ساختار درختی، روابط بین عناصر صفحه را مشخص می‌کند. قوانین CSS مستقیماً روی این ساختار اعمال می‌شوند و انتخاب‌گرها (Selectors) تعیین می‌کنند کدام عناصر در DOM تحت تأثیر یک قانون قرار بگیرند.

CSS در کنار HTML و JavaScript سه رکن اصلی Front-End را تشکیل می‌دهد. HTML لایه ساختاری، CSS لایه ظاهری و JavaScript لایه رفتاری و تعاملی را پوشش می‌دهد. CSS یک زبان برنامه‌نویسی محسوب نمی‌شود، بلکه یک زبان توصیفی است که مجموعه‌ای از قوانین نمایشی را برای مرورگر تعریف می‌کند. به همین دلیل، منطق اجرایی و تصمیم‌گیری در حوزه CSS قرار ندارد و تمرکز آن صرفاً بر ظاهر و چیدمان است.

فرض کن HTML اسکلت یک ساختمان است، CSS دقیقاً می‌شود رنگ، نقاشی، دکوراسیون و چیدمان ساختمان

با CSS مشخص می‌کنیم:

- متن‌ها چه رنگی باشند
- اندازه فونت چقدر باشد
- فاصله‌ها چطور تنظیم شوند
- عناصر کنار هم باشند یا زیر هم
- صفحه در موبایل و دسکتاپ چطور دیده شود

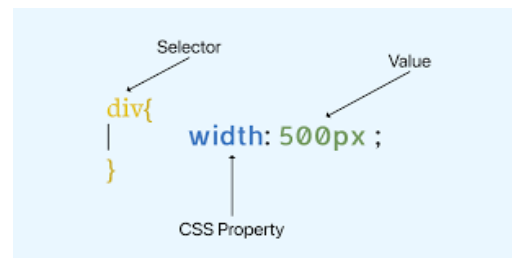
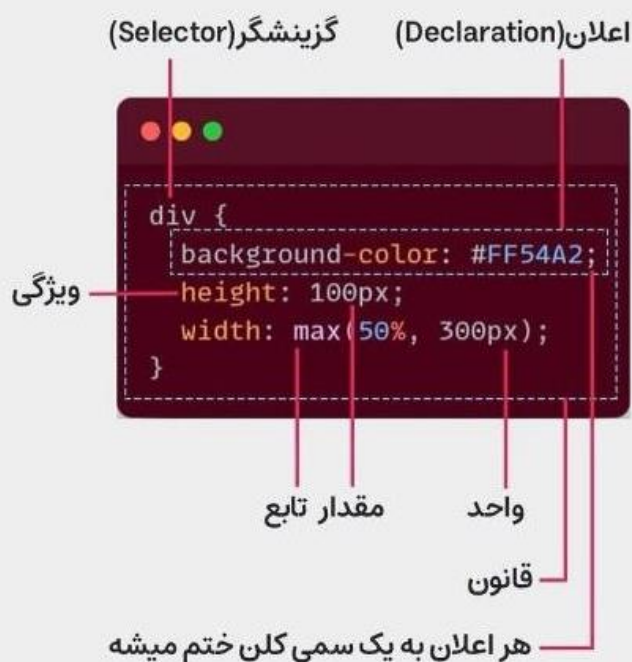
بدون CSS صفحه وب خیلی ساده، زشت و نامرتب است

## قانون CSS از چه بخش‌هایی ساخته شده؟

هر قانون CSS سه قسمت دارد:

```
selector {  
  property: value;  
}
```

- Selector → روی چه چیزی؟
- Property → چه چیزی را تغییر بده؟
- Value → چطور تغییر بده؟



## CSS چطور به HTML وصل می‌شود؟

### ۱) روش Inline (داخل خود تگ)

```
<p style="color:red; font-size:۲۰px;"> سلام دنیا </p>
```

#### ویژگی‌ها:

- سریع و ساده
- مناسب تست یا مثال‌های خیلی کوچک
- برای پروژه واقعی اصلاً پیشنهاد نمی‌شود

### ۲) روش Internal (داخل تگ style)

```
<head>  
<style>  
  p {  
    color: blue;  
    font-size: ۱۸px;  
  }  
</style>  
</head>
```

#### ویژگی‌ها:

- بهتر از Inline
- مناسب پروژه‌های کوچک
- در پروژه‌های بزرگ شلوغ می‌شود

### ۳) روش External (فایل جداگانه CSS) بهترین روش

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

داخل فایل CSS :

```
p {  
  color: blue;  
  font-size: ۱۸px;  
}
```

#### ویژگی‌ها:

- تمیز
- حرفه‌ای
- قابل استفاده برای چند صفحه
- روش استاندارد بازار کار

## Cascade یعنی چی؟

اگر چند قانون روی یک عنصر اعمال شود، مرورگر باید تصمیم بگیرد کدام را اجرا کند.

```
p { color: blue; }  
p { color: red; }
```

📌 نتیجه: متن قرمز می‌شود: چون قانون پایینی جدیدتر است

## Specificity (اولویت قوانین)

بعضی Selector ها قوی‌تر هستند

```
p { color: blue; }  
.text { color: green; }  
#title { color: red; }  
  
<p id="title" class="text"> سلام </p>
```

📌 نتیجه: رنگ قرمز:

چرا؟ چون : اولویت با id > class > tag است

## important! (خیلی قوی ولی خطرناک)

```
p {  
  color: blue !important;  
}
```

این دستور همه قوانین دیگر را شکست می‌دهد

📌 فقط وقتی استفاده کن که:

- واقعاً چاره‌ای نداری
- بدانی داری چه کار می‌کنی