

# آموزش واحدهای تعیین اندازه فونت در CSS



## آموزش واحدهای تعیین اندازه فونت در CSS

فونت ها نقش مهمی را در ایجاد یک وب سایت کاربر پسند و افزایش قابلیت خوانایی محتوای یک وب سایت بازی می کنند. در این آموزش قصد داریم واحدهای مختلف اندازه گیری برای تعیین اندازه فونت را بررسی کنیم . یکی از نکات مهم در تایپوگرافی، اندازه متن است که با استفاده از خاصیت `font-size` بصورت زیر تعریف می شود:

```
font-size: size;
```

اینجا size یک واحد اندازه گیری طول در CSS است. مقادیر اندازه برای هر یک از این اندازه گیری ها می تواند اعداد کامل ( ۰ ، ۱ ، ۲ ، ... ) یا اعشار ( ۰٫۵ ، ۱٫۶ ، ۳٫۹ ، ... ) باشد. طول (و عرض) در CSS به صورت واحدهای مطلق یا واحدهای نسبی بیان می شود.

### واحدهای مطلق

واحدهای مطلق واحدهایی هستند که بدون توجه به دستگاه خروجی اندازه ثابت دارند و معمولاً فقط در رسانه چاپی استفاده می شوند. آنها در یکی از پنج واحد استاندارد اندازه گیری مشخص می شوند: mm (میلی متر)، cm (سانتی متر)، in (اینچ)، pt (نقطه).

به عنوان مثال، برای تنظیم اندازه فونت صفحه خود روی 12pt، می توانید از روش زیر استفاده کنید:

```
body  
{  
  font-size: 12pt;  
}
```

توجه داشته باشید که نباید فاصله ای بین مقدار اندازه و واحد مخفف شده وارد کنید.

## واحدهای نسبی

واحدهای مطلق کاربرد محدود دارند زیرا در بیشتر موارد، طراح صفحه از مشخصات دقیق دستگاه ارائه دهنده صفحه اطلاع ندارد. به جای واحدهای مطلق، طراحان از واحدهای نسبی استفاده می کنند که نسبت به اندازه سایر اشیاء موجود در صفحه وب یا نسبت به خصوصیات نمایشگر بیان می شوند.

واحد اصلی اکثر دستگاه ها پیکسل (px) است که یک نقطه واحد در دستگاه خروجی را نشان می دهد. پیکسل یک واحد نسبی است زیرا اندازه پیکسل واقعی به وضوح و چگالی دستگاه خروجی بستگی دارد. یک مانیتور رومیزی می تواند دارای تراکم پیکسلی در حدود 96ppi (پیکسل در هر اینچ) باشد، لپ تاپ ها در حدود ۱۰۰ تا ۱۳۵ ppi هستند، در حالی که تلفن های همراه دارای نمایشگرهای متراکم با ۲۰۰ تا ۳۰۰ ppi یا بیشتر هستند. به طور معمول، بیشتر مرورگرها اندازه فونت پایه 16px را برای متن اصلی اعمال می کنند. می توانید با تنظیمات خاص خود این اندازه های پیش فرض را نادیده بگیرید. به عنوان مثال، قوانین سبک زیر اندازه فونت متن موجود در صفحه اصلی را به 10px و اندازه فونت تمام عناوین h1 را به ۱۴ پیکسل تنظیم می کند:

```
body
{
  font-size: 10px;
}
h1
{
  font-size: 14px;
}
```

ظاهر دقیق متن تا حد زیادی به تراکم پیکسل دستگاه بستگی دارد. در حالی که یک فونت 10px ممکن است در یک مانیتور دسک تاپ خوب باشد، اما همان اندازه فونت می تواند در یک دستگاه تلفن همراه قابل خواندن نباشد.

## مقیاس گذاری فونت ها با %، ems و rems

از آنجا که طراح صفحه ویژگی های دقیق دستگاه کاربر را نمی داند، روش معمول این است که متن را با تمام اندازه های فونت بیان شده نسبت به اندازه فونت پیش فرض مقیاس پذیر کنید. سه اندازه گیری نسبی وجود دارد که برای فراهم آوردن مقیاس پذیری استفاده می شود: درصد، ems و rems.

درصد (٪) اندازه فونت را به عنوان درصد فونت مورد استفاده توسط تگ پدر تعیین می کند. به عنوان مثال، دستور زیر اندازه فونت عنوان h1 را به ۲۰۰٪ یا دو برابر اندازه فونت تگ پدر تنظیم می کند:

```
h1
{
  font-size: 200%;
}
```

واحد em مانند نسبت درصد عمل می کند و اندازه فونت را نسبت به اندازه فونت تگ پدر بیان می کند.

```
h1
{
  font-size: 2em;
}
```

به عنوان مثال، اگر عنصر h1 مستقیماً درون body قرار بگیرد که اندازه فونت body آن 16px باشد، عنصر h1 اندازه فونت را دو برابر اندازه یا 32px خواهد داشت. از طرف دیگر، اگر h1 درون پاراگرافی قرار بگیرد که اندازه فونت آن برابر با 9px باشد، اندازه فونت آن 18px خواهد بود. به طور کلی، می توانید اندازه فونت ها را بر اساس درصد و واحدهای em به نسبت اندازه متن مجاور تعیین کنید.

```
body
{
  font-size: 16px;
}
body > article {font-size: 0.75em;}
body > article > h1 {font-size: 1em;}
```

در اینجا می توانید نتیجه بگیرید که اندازه فونت عنصر h1 از اندازه فونت مورد استفاده در المان article (چون  $1em > 0.75em$ ) بزرگتر است. با این حال، این طور نیست و هر دو اندازه فونت یکسان هستند. زیرا به یاد داشته باشید، em واحد اندازه متن را نسبت به اندازه فونت مورد استفاده در تگ پدر بیان می کند در اینجا اندازه فونت عنصر article برابر ۱۲ پیکسل است یعنی  $(16px * 0.75em)$  و اندازه فونت h1 نیز برابر ۱۲ پیکسل است یعنی  $(12px * 1em)$

به دلیل این سردرگمی، برخی از طراحان مدافع استفاده از واحد rem یا root em هستند که در آن اندازه فونت ها همیشه نسبت به اندازه فونت مورد استفاده در عنصر html بیان می شوند. با استفاده از rems، روش زیر اندازه فونت متن article را به ۷۵٪ از ۱۶ پیکسل یا ۱۲ پیکسل تنظیم می کند، در حالی که اندازه عنوان h1 روی ۱۶ پیکسل یعنی  $(1 * 16)$  تنظیم شده است:

```
html {font-size: 16px;}
article {font-size: 0.75rem;}
article > h1 {font-size: 1rem;}
```

با افزایش پشتیبانی مرورگر، واحد rem محبوب تر شده و استفاده از آن ممکن است در سالهای آینده جایگزین استفاده از واحد em شود.

## استفاده از واحدهای Viewport

واحد نسبی دیگر واحد Viewport است که در آن طول ها به عنوان درصدی از عرض یا ارتفاع پنجره مرورگر بیان می شود. با تغییر اندازه پنجره مرورگر، اندازه متن بر اساس واحد Viewport برای مطابقت تغییر می کند. CSS3 چهار واحد Viewport را معرفی کرد: vw, vh, vmin و vmax که هر کدام عبارتند از:

- 1vw : ۱٪ عرض پنجره مرورگر
- 1vh : ۱٪ ارتفاع پنجره مرورگر
- 1vmin : 1vw یا 1vh (هر کدام که کوچکتر است)
- 1vmax : 1vw یا 1vh (هر کدام که بزرگتر است)

به عنوان مثال، اگر پنجره مرورگر ۱۳۶۶ پیکسل عرض داشته باشد، طول 1vw برابر با ۱۳,۶۶ پیکسل خواهد بود. اگر عرض پنجره به ۷۸۰ پیکسل کاهش یابد، 1vw به طور خودکار به ۷,۸ پیکسل کاهش می یابد. مزیت مقیاس پذیری خودکار این است که اندازه فونت هایی که با یک واحد Viewport تنظیم می شوند، مطابق با اندازه پنجره مرورگر تغییر می کنند. معایب آن این است که اگر پنجره مرورگر خیلی کوچک شود، متن صفحه به سرعت قابل خواندن نخواهد بود.

## کلمات کلیدی سایز

در آخر، شما همچنین می توانید اندازه فونت را با استفاده از کلمات کلیدی زیر بیان کنید: x-small, xx small, small, medium, large, x-large, xx large, larger یا smaller. اندازه فونت مربوط به هر یک از این کلمات کلیدی توسط مرورگر تعیین می شود. توجه داشته باشید که کلمات کلیدی larger و smaller اندازه های نسبی هستند و باعث می شود اندازه فونت، اندازه ای بزرگتر یا کوچکتر از اندازه فونت تگ پدر داشته باشد. به عنوان مثال، قوانین زیراندازه فونت تگ aside را با یک فونت کوچک نمایش می دهد، در حالی که یک عنصر h1 که درون aside قرار گرفته است با فونت یک اندازه بزرگتر (متوسط) نمایش داده می شود:

```
aside
{
font-size: small;
}
```

```
aside > h1 {font-size: larger;}
```