

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

کارگاه آموزش HTML و CSS

پاییز 1403، مهدی یار پورسالاری



display

با استفاده از این خاصیت می توان نحوه نمایش یک عنصر را مشخص کرد. برای مثال می توان تعیین کرد بصورت block-level یا inline در صفحه قرار بگیرد. پر کاربرد ترین مقادیری که این خاصیت قبول می کند:

- block
- inline
- inline-block
- flex
- grid

معرفی flexbox



FlexBox یک ماژول یا یک قابلیت برای طرح بندی در یک صفحه وب می باشد، که با استفاده از آن میتوان طراحی انعطاف پذیر تری داشته باشیم.

با استفاده از این قابلیت می توان عناصری که در نگهدارنده استفاده می شوند را بصورت عمودی یا افقی کنار هم قرار داد و این "کنار هم قرار دادن" را مدیریت کرد.

```
display=flex;
```



در اولین مرحله برای استفاده از flexbox شما باید یک نگهدارنده با

```
display: flex;
```

ایجاد نمایید، بعد از آن می توان از ویژگی های مرتبط با flex بروی نگهدارنده ای که ایجاد کردیم استفاده کرد.

flexbox attributes



خواصی که برای استفاده از flexbox بکار گرفته می شوند به دو دسته تقسیم می شوند:

خواصی که برای نگهدارنده استفاده می شوند.
خواصی که برای عناصر داخل نگهدارنده استفاده می شوند.

خواص نگهدارنده



با استفاده از ویژگی flex-direction می توان جهت قرار گیری آیتم ها را مشخص کرد. مقادیری که این خاصیت قبول می کند:

- row: جهت دهی فرزندان بصورت افقی.
- row-revers: جهت دهی فرزندان بصورت افقی اما در خلاف جهت جریان عناصر در صفحه
- column: جهت دهی فرزندان بصورت عمودی
- column-reverse: جهت دهی فرزندان بصورت عمودی اما از پایین به بالا

خواص نگهدارنده



با استفاده از ویژگی flex-wrap می توان در صورتی که همه عناصر در یک سطر جای نگرفتند، تعدادی از آنها را به سطر های بعدی منتقل کرد تا عناصر فرزند، بیش از حد فشرده نشوند و شکل اولیه خود را حفظ کنند. این کار بصورت خودکار توسط مرورگر انجام می شود. مقادیری که این خاصیت قبول می کند:

- nowrap: عدم شکستگی عناصر؛ پیش فرض.
- wrap: در صورت لزوم عناصر شکسته و به سطر بعدی منتقل می شوند.
- wrap-reverse: فرزندان در صورت لزوم شکسته و بصورت برعکس نمایش داده می شوند.

خواص نگهدارنده



با استفاده از justify-content می توان در صورتی که عناصر کل فضا را پر نکرده باشند، فضای خالی باقیمانده را در جهتی که flex-direction مشخص می کند مدیریت کرد.

مقادیر justify-content



مقدار	توضیح
flex-start	مقدار پیش فرض، عناصر در ابتدا قرار می گیرند.
flex-end	عناصر در انتهای نگهدارنده قرار می گیرند.
center	عناصر در وسط نگهدارنده قرار می گیرند.
space-between	فضای خالی باقی مانده را بصورت مساوی بین عناصر تقسیم می کند. در این حالت فضای خالی در ابتدا و انتهای نگهدارنده وجود نخواهد داشت.
space-around	بیشتر فضای خالی در اطراف عناصر فرزند، و مابقی آن بصورت مساوی بین آنها تقسیم می شود.
space-evenly	فضای خالی بین عناصر بصورت مساوی در ابتدا و انتهای نگهدارنده و ما بین عناصر فرزند تقسیم می شود.

خواص نگهدارنده



با استفاده از align-items می توان عناصر را در جهت مخالف جهتی که flex-direction مشخص می کند تراز کرد. مثلا اگر

`flex-direction: row;`

باشد، مقدار این خاصیت عناصر را در جهت عمودی و اگر

`flex-direction: column;`

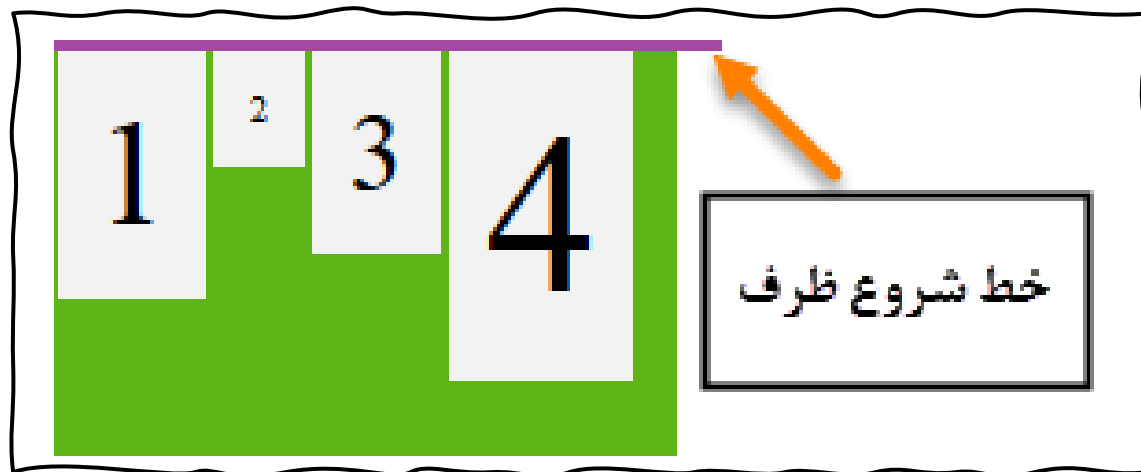
باشد عناصر را در جهت افقی تراز می کند.

مقادیر align-items



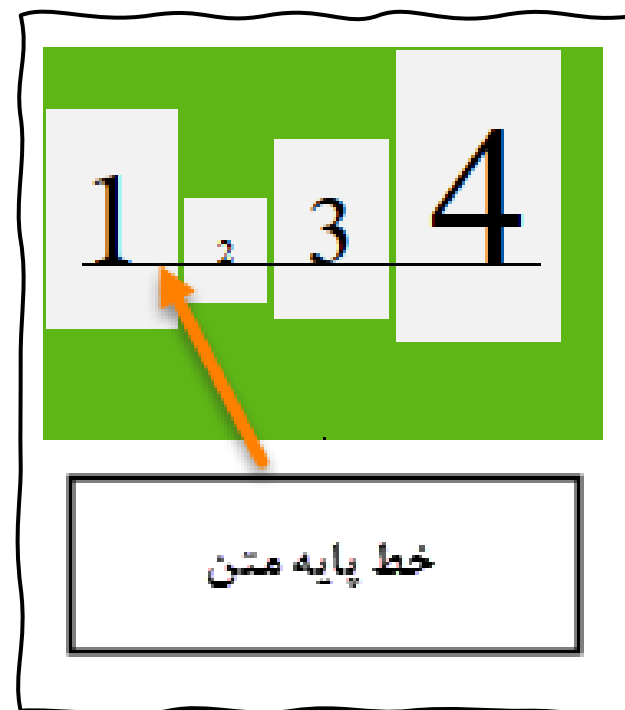
مقدار	توضیح
stretch	مقدار پیش فرض، عناصر کشیده می شوند تا فضای نگهدارنده را پر کنند.
center	عناصر در وسط نگهدارنده قرار می گیرند.
flex-start	عناصر در ابتدای نگهدارنده قرار می گیرند.
flex-end	عناصر در انتهای نگهدارنده قرار می گیرند.
base-line	آیتم ها به نسبت خط پایه متن تراز می شوند.

base-line



align-items: flex-start

align-items: base-line



خواص نگهدارنده



در نظر بگیرید:

`flex-wrap: wrap | wrap-reverse;`

در این حالت می توان خاصیت `justify-content` برای عناصری که در سطر های بعدی می افتند را با استفاده از `align-content` تنظیم کرد. این خاصیت تمام مقادیری که برای `justify-content` ذکر شد را قبول می کند.

خواص نگهدارنده



با استفاده از gap می توان بین عناصر فرزند فاصله ای را مشخص کرد. **دقت کنید** که این فاصله مستقل از خاصیت justify-content اعمال می شود و justify-content این فضای اضافی بین عناصر را در نظر نمی گیرد و کاری با آن ندارد.

نحو این خاصیت بدین شکل است:

`gap: row-gap column-gap; => gap: 20px 60px;`

`gap: row-gap and column-gap; => gap: 30px;`

خواص فرزندان



flex-grow مشخص می کند که یک عنصر چقدر به نسبت بقیه عناصر داخل نگهدارنده یکسان رشد خواهد کرد. به زبان دیگر این خاصیت مشخص می کند که هر آیتم درون یک کانتینر Flexbox چقدر از **فضای اضافی** در محور اصلی را اشغال کند.

بصورت پیش فرض مقدار این خاصیت صفر است که یعنی عنصر به اندازه min-width یا width خود فضا اشغال می کند و از فضای اضافه سهمی ندارد.

اما در صورتی که عناصر flex-grow بیشتر از یک داشته باشند، تمام flex-grow ها با هم جمع شده و فضای خالی بین آنها با توجه به سهمی که دارند پخش می شود.

خواص فرزندان



flex-shrink مشخص می کند که یک عنصر چقدر به نسبت بقیه عناصر داخل نگهدارنده در صورت کمبود فضا فشرده خواهد شد.

بصورت پیش فرض مقدار این خاصیت صفر است که یعنی عنصر به اندازه بقیه عناصر فشرده شود.

در صورتی که فضا کم باشد، اندازه هر آیتم از این طریق محاسبه می شود.

$$\text{اندازه ی کاهش یافته} = \text{اندازه ی اصلی} - \left(\text{فضای اضافی کل} \times \frac{\text{flex-shrink}}{\text{flex-shrink ها جمع}} \right)$$

خواص فرزندان



مثال 1: وقتی `flex-shrink` برابر باشد

فرض کنید سه آیتم دارید:

- آیتم 1: عرض اصلی = `flex-shrink: 1`, 200px
- آیتم 2: عرض اصلی = `flex-shrink: 1`, 300px
- آیتم 3: عرض اصلی = `flex-shrink: 1`, 500px

و فضای کل کانتینر فقط 800px است.

- جمع عرض‌ها = $1000px = 500 + 300 + 200$
- فضای اضافی = $200px = 800 - 1000$

چون همه آیتم‌ها مقدار `flex-shrink` یکسان دارند، فضای اضافی به طور مساوی نسبت به اندازه اصلی هر آیتم کم می‌شود:

- آیتم 1: $160px = 200 \times (1000 \div 200) - 200$
- آیتم 2: $240px = 200 \times (1000 \div 300) - 300$
- آیتم 3: $400px = 200 \times (1000 \div 500) - 500$

خواص فرزندان



با استفاده از align-self می توان یک یا چندتا از عناصر فرزند را در جهت مخالف جهتی که flex-direction مشخص می کند تراز کرد. مثلا اگر

`flex-direction: row;`

باشد، مقدار این خاصیت عناصر را در جهت عمودی و اگر

`flex-direction: column;`

باشد عناصر را در جهت افقی تراز می کند.

این خاصیت تمامی مقادیری که برای align-items ذکر شد را قبول می کند.

خواص فرزندان



flex-basis یکی از ویژگی‌های مهم در Flexbox است که تعیین می‌کند آیتم فرزندان، قبل از اعمال مقادیر flex-grow و flex-shrink چه اندازه‌ای داشته باشد. این ویژگی در واقع اندازه اولیه یک آیتم را در راستای محور اصلی مشخص می‌کند.

مقدار پیش فرض آن auto است که در این حالت اندازه را از width یا height مشخص شده می‌گیرد. اما در حالت کلی تمامی مقادیری که بعنوان width و height می‌توان استفاده کرد را می‌توان به این خاصیت نسبت داد.

خواص فرزندان



با استفاده از order می توان ترتیب عناصری که در یک نگهدارنده یکسان قرار دارند را مشخص کرد. این خاصیت بعنوان مقدار یک عدد قبول می کند.