

CSS FLEXBOX

基本

flexboxとは？

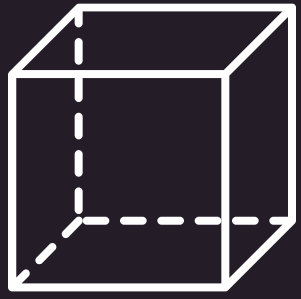
項目を行または列にレイアウトするための1次元のレイアウト方法。

比較的新しい

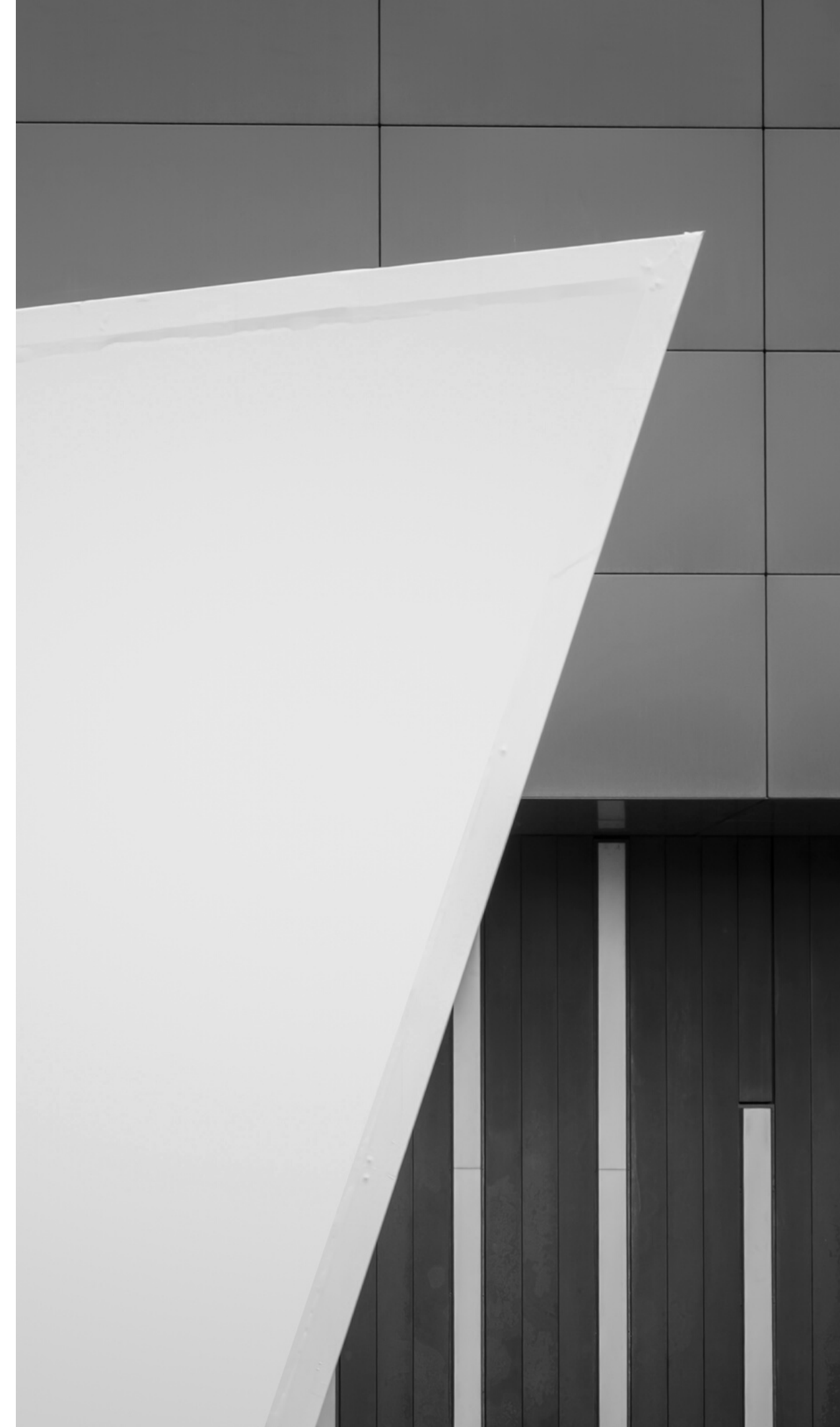
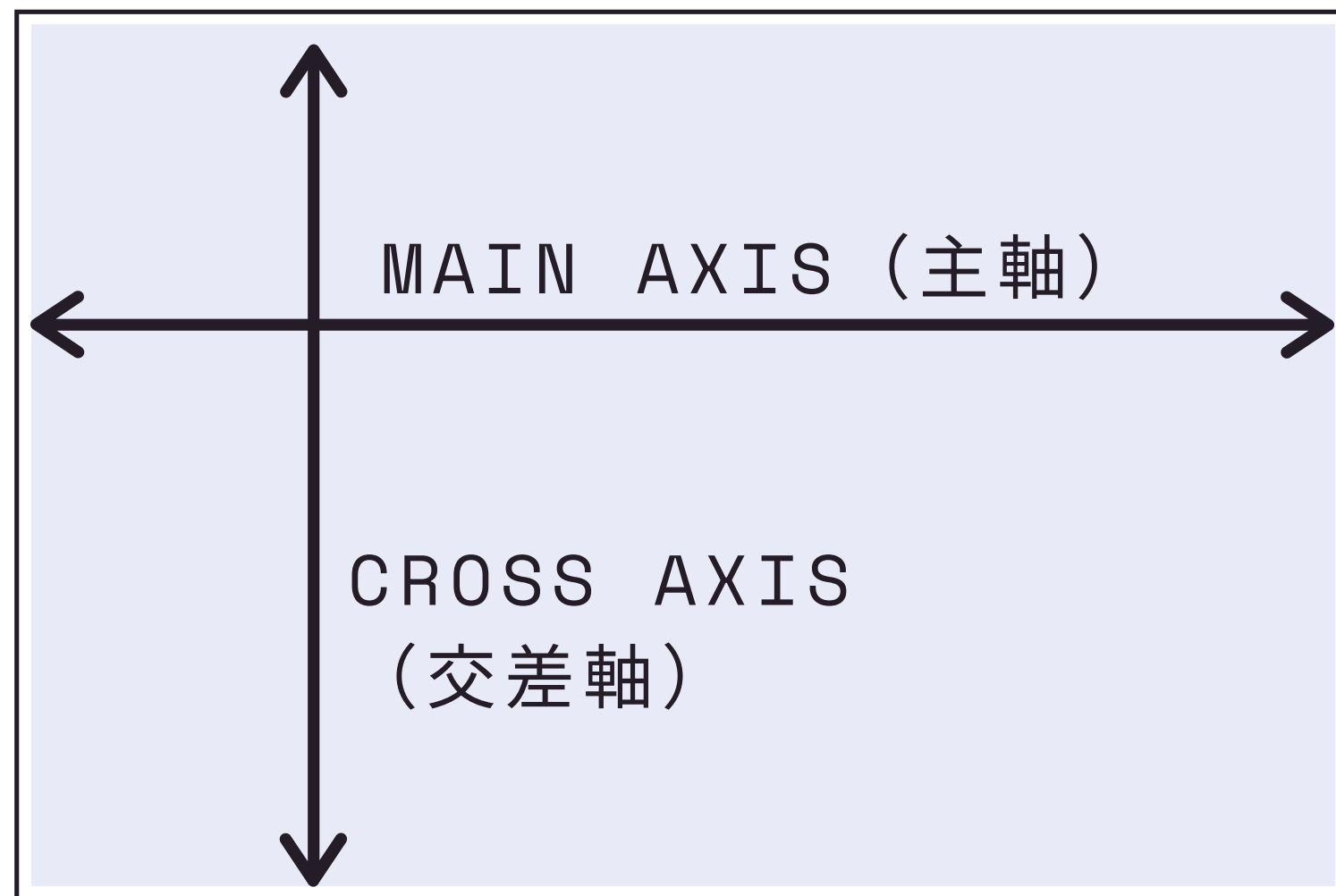
flexboxがCSSに加わったのは比較的最近。多くのレイアウト作業を簡単にしてくれた。

なぜ「flex」？

大きさのわからない要素同士でも、空間をどのように使って要素を配置するのか柔軟に調整できるのがflexboxの強み。



フレックスモデル



FLEX DIRECTION



```
flex-direction: row;
```

FLEX DIRECTION



```
flex-direction: row-reverse;
```

FLEX DIRECTION



```
flex-direction: column;
```



```
flex-direction: column-reverse;
```

FLEX WRAP



```
flex-wrap: wrap;
```

JUSTIFY CONTENT



```
justify-content: flex-start;
```


JUSTIFY CONTENT



```
justify-content: flex-end;
```

JUSTIFY CONTENT



```
justify-content: center;
```

JUSTIFY CONTENT



```
justify-content: space-between;
```

JUSTIFY CONTENT



```
justify-content: space-around;
```

ALIGN ITEMS



```
align-items: flex-start;
```

ALIGN ITEMS



```
align-items: flex-end;
```

ALIGN ITEMS



```
align-items: center;
```

ALIGN ITEMS



```
align-items: stretch;
```


ALIGN CONTENT



```
align-content:space-between;
```

ALIGN CONTENT



```
align-content: flex-start;
```

ALIGN CONTENT



```
align-content: flex-end;
```

ALIGN CONTENT



```
align-content:center;
```

ALIGN SELF




```
align-self: flex-end;
```

フレックスアイテム の大きさに関連する プロパティ



flex-basis

フレックスコンテナ内のフレックスアイテムの基本幅を指定する。



flex-grow

フレックスコンテナ内の残りの空間のうち、どれだけがそのアイテムに割り当てられるかを設定する。



flex-shrink

フレックスアイテムがフレックスコンテナより大きいとき、flex-shrinkに従って縮小する。

RESPONSIVE DESIGN

何なのか & 何故気にすべきなのか

PROBLEM

スマホやタブレットが普及し、どのようにしたらすべてのデバイスで見栄えの良いウェブサイトを作ることができるのか、という課題が生まれた。

初期の解決策

CSSをデバイスの種類毎に作るようにしていた。また、デバイス毎に完全に別のウェブサイトを作ることも珍しくはなかった。

レスポンシブの登場

最近では一つのウェブサイトとCSSで、複数のデバイスに対応できるようにしている。

MEDIA QUERY

メディアクエリーを使って、画面の解像度や端末の種類に応じてスタイルを変えられるようになる。




```

@media (max-width: 800px) {
  .sidebar {
    display: none;
  }

  .main {
    width: 80%;
  }
}

@media (min-width: 30em) and (orientation: landscape) {
  #container {
    flex-direction: column;
    justify-content: center;
  }
}

```

MEDIA QUERY