CSS flexbox cheatsheet

Proudly sponsored by

Airbrake.io Full-stack error tracking & analytics for Ruby developers. Try it Free! 🚀

ethical ad by CodeFund

Simple example

```
.container {
  display: flex;
}

.container > div {
  flex: 1 1 auto;
}
```

Container

```
.container {

  display: flex;
  display: inline-flex;

  flex-direction: row;            /* ltr - default */
  flex-direction: row-reverse;    /* rtl */
  flex-direction: column;         /* top-bottom */
  flex-direction: column-reverse; /* bottom-top */

  flex-wrap: nowrap; /* one-line */
  flex-wrap: wrap;   /* multi-line */

  align-items: flex-start; /* vertical-align to top
  align-items: flex-end;   /* vertical-align to bottc
  align-items: center;     /* vertical-align to cent
  align-items: stretch;    /* same height on all (def

  justify-content: flex-start; /* horizontal alignmer
  justify-content: flex-end;
  justify-content: center;

}
```

Child

```
.container > div {

  /* This: */
  flex: 1 0 auto;

  /* Is equivalent to this: */
  flex-grow: 1;
  flex-shrink: 0;
  flex-basis: auto;

  order: 1;

  align-self: flex-start;  /* left */
  margin-left: auto;       /* right */

}
```

Tricks

Vertical center

```
.container {
  display: flex;
}

.container > div {
  width: 100px;
  height: 100px;
  margin: auto;
}
```

Vertical center (2)

```
.container {
  display: flex;
  align-items: center;     /* vertical */
  justify-content: center; /* horizontal */
}
```

Reordering

```
.container > .top {
  order: 1;
}

.container > .bottom {
  order: 2;
}
```

Mobile layout

```
.container {
  display: flex;
  flex-direction: column;
}

.container > .top {
  flex: 0 0 100px;
}

.container > .content {
  flex: 1 0 auto;
}
```

A fixed-height top bar and a dynamic-height content area.

Table-like

```
.container {
  display: flex;
}

/* the 'px' values here are just suggested percentage
.container > .checkbox { flex: 1 0 20px; }
.container > .subject  { flex: 1 0 400px; }
.container > .date     { flex: 1 0 120px; }
```

This creates columns that have different widths, but size accordingly according to the circumstances.

Vertical

```
.container {
  align-items: center;
}
```

Vertically-center all items.

Left and right

```
.menu > .left  { align-self: flex-start; }
.menu > .right { align-self: flex-end;   }
```

References

- MDN: Using CSS flexbox
- Ultimate flexbox cheatsheet

▸ **1 Comment** for this cheatsheet. Write your

Search 380+ cheatsheets

Over 380 curated cheatsheets, by developers for developers.

Devhints home

Other CSS cheatsheets

Sass cheatsheet

CSS system fonts cheatsheet

cssnext cheatsheet

Stylus cheatsheet

Bootstrap cheatsheet

Bulma cheatsheet

Top cheatsheets

Elixir cheatsheet

ES2015+ cheatsheet

React.js cheatsheet

Vimdiff cheatsheet

Vim cheatsheet

Vim scripting cheatsheet