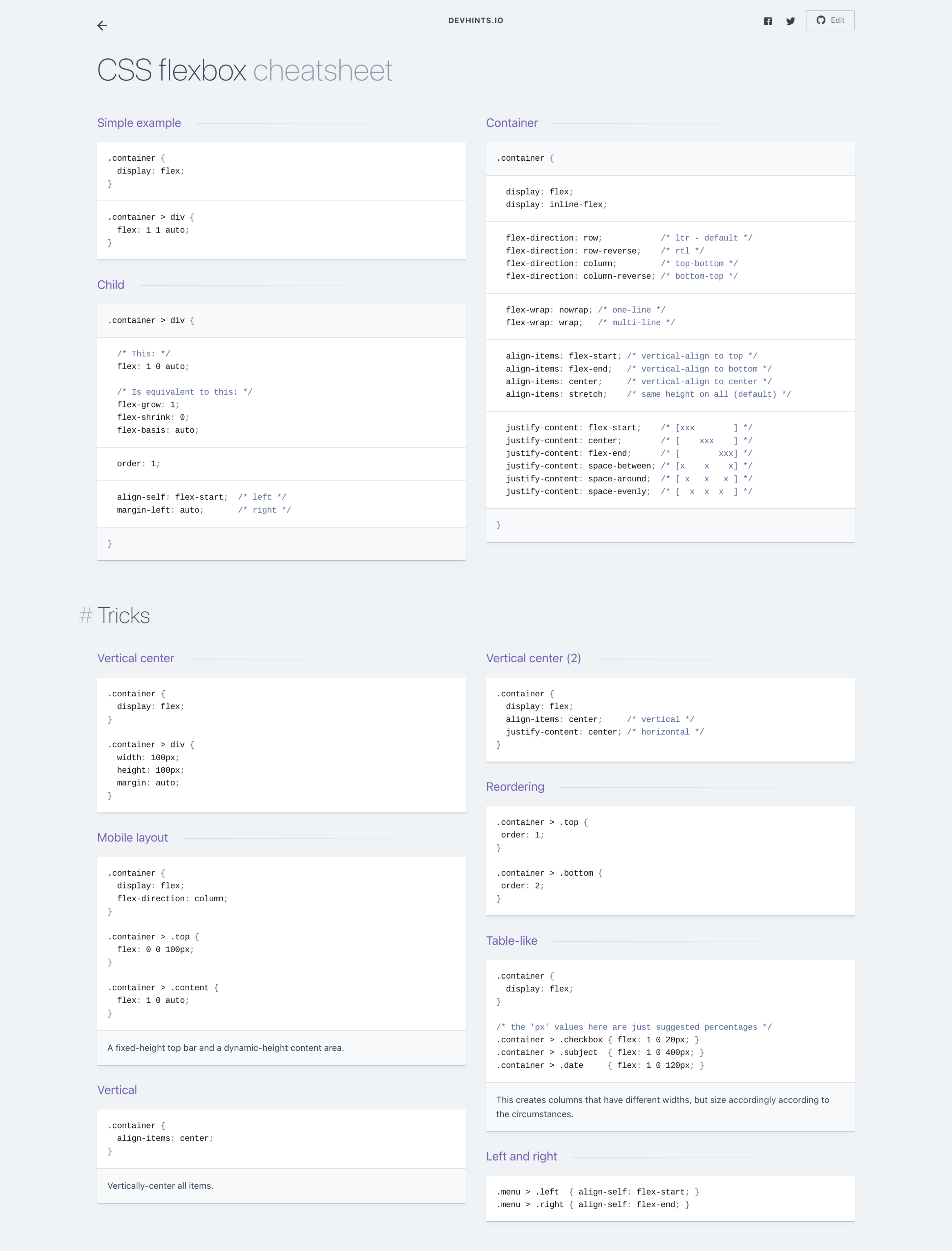

DEVHINTS.IO

Edit

CSS flexbox cheatsheet

Simple example

```
.container {
  display: flex;
}
```

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

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 */

}
```

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 bottom */
  align-items: center;     /* vertical-align to center */
  align-items: stretch;    /* same height on all (default) */

  justify-content: flex-start;    /* [xxx        ] */
  justify-content: center;        /* [    xxx    ] */
  justify-content: flex-end;      /* [        xxx] */
  justify-content: space-between; /* [x    x    x] */
  justify-content: space-around;  /* [ x   x   x ] */
  justify-content: space-evenly;  /* [  x  x  x  ] */

}
```

Tricks

Vertical center

```
.container {
  display: flex;
}

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

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.

Vertical

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

Vertically-center all items.

Vertical center (2)

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

Reordering

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

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

Table-like

```
.container {
  display: flex;
}

/* the 'px' values here are just suggested percentages */
.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.

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 yours!

devhints.io / Search 357+ cheatsheets

Over 357 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