

Layout

HTML & CSS Reference

Width & centering

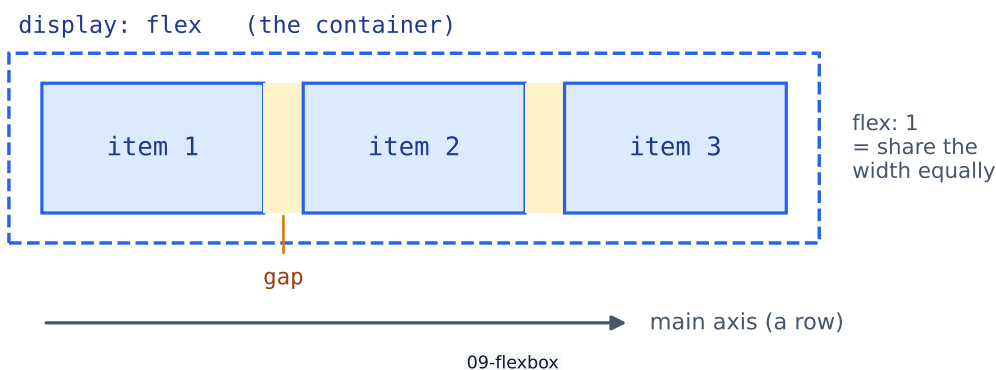
Set a box's width. To centre a block, give it a width and `margin: 0 auto` — `auto` shares the leftover space equally on the left and right.

```
<style>
  .page { width: 300px; margin: 0 auto; background: #e0e7ff;
          padding: 16px; text-align: center; }
</style>
<div class="page">A centered, fixed-width page.</div>
```

Flexbox

Flexbox lays children out in a row (or a column). Set `display: flex` on the parent — the container. `gap` adds space between children; `flex: 1` makes them share the width.

```
<style>
  .row { display: flex; gap: 10px; }
  .row div { background: #93c5fd; padding: 12px; flex: 1; text-align:
  ↪ center; }
</style>
<div class="row">
  <div>One</div>
  <div>Two</div>
  <div>Three</div>
</div>
```



A flex container lays its items along the main axis; gap adds space between them

Mini-project: a profile card

Combine boxes, colours, text and spacing into one small component .

```
<style>
  .card { width: 240px; margin: 0 auto; border: 1px solid #cbd5e1;
          border-radius: 12px; padding: 16px; font-family: sans-serif; }
  .card h2 { margin: 0; color: #1e3a8a; }
  .card .role { color: #64748b; margin: 4px 0 8px; }
</style>
<div class="card">
  <h2>Mei Chen</h2>
  <p class="role">Student · Grade 11</p>
  <p>Loves maths and building web pages.</p>
</div>
```

Common mistakes

- Centre a block with a set width using `margin: 0 auto`.
- Flexbox needs `display: flex` on the CONTAINER, not on the items.
- A width in % is relative to the parent, so check the parent's width too.