CSS Layout

Block Layout

- `display: block`
- Default for <div> and most other elements
- Takes up all of the available horizontal space
- Block elements stack on top of each other
- Best for blocks of flowing text

Flex Layout

- `display: flex`
- The parent element continues to behave like a block element, but it changes the way its child elements are laid out — into rows OR columns.
- Children flow in one dimension.
    - By default, with just `display: flex` applied to the parent, the children will be distributed horizontally.
    - The flow direction can be switched from horizontal back to vertical with `flex-direction: column`.
- Children size themselves with the `flex` property.
- Children can wrap if there’s too many to fit in one row/column.
    - `flex-wrap: wrap`

Grid Layout

- `display: grid`
- The parent element continues to behave like a block element, but it changes the way its child elements are laid out — into rows AND columns like a table.
- Children flow in two dimensions, determined by parent.
- Parent defines the size of children.

Centering

```
<div class="centered">Hello world!</div>
<style>
  .centered {
    display: flex; /* or display: grid */
    height: 100%;
    place-items: center;
    place-content: center;
  }
</style>
```

Centering Block Elements Horizontally

```
<div class="horizontally-centered">Hello world!</div>
<style>
  .horizontally-centered {
    margin-left: auto;
    margin-right: auto;
    width: 50%;
  }
</style>
```