CSS Layout Cheat Sheet: Flexbox & Grid 🧩🎨

🌟 Flexbox (1D Layout)

Flexbox is used for one-dimensional layouts (rows or columns).

Parent (Container) Properties

Property	Description	Values
`display: flex;`	Activates flexbox layout for the container.	N/A
`flex-direction`	Defines the direction of flex items.	`row` (default), `row-reverse`, `column`, `column-reverse`
`justify-content` 🔄	Aligns items **horizontally** in the container.	`flex-start`, `flex-end`, `center`, `space-between`, `space-around`, `space-evenly`
`align-items` ⬆️⬇️	Aligns items **vertically** in the container.	`flex-start`, `flex-end`, `center`, `baseline`, `stretch`
`align-content` ➡️⬅️	Aligns multiple lines of items (if wrapping is enabled) **vertically**.	Same values as `align-items`
`flex-wrap`	Controls whether items wrap to a new row/column if needed.	`nowrap` (default), `wrap`, `wrap-reverse`

Child (Item) Properties

Property	Description	Values
`order`	Changes the order of items.	Integer (e.g., `1`, `-1`)
`flex-grow`	Controls how much an item grows relative to others.	Number (e.g., `0`, `1`, `2`)
`flex-shrink`	Controls how much an item shrinks relative to others.	Number (e.g., `0`, `1`, `2`)
`flex-basis`	Specifies the initial size of an item (before growing/shrinking).	`auto`, length units (e.g., `50%`, `100px`)
`align-self` ⬆️	Overrides `align-items` for a specific item.	Same values as `align-items`

🌟 Grid (2D Layout)

Grid is used for two-dimensional layouts (rows and columns).

Parent (Container) Properties

Property	Description	Values
`display: grid;`	Activates grid layout for the container.	N/A
`grid-template-columns`	Defines the columns in the grid.	Length units (e.g., `100px`, `1fr`, `auto`)
`grid-template-rows`	Defines the rows in the grid.	Same as above
`grid-template-areas`	Defines named areas of the grid layout.	Text values matching child `grid-area` names.
`gap`	Sets spacing between rows and columns.	`10px`, `1rem`, etc.
`justify-items` 🔄	Aligns items **horizontally** within their grid cell.	`start`, `end`, `center`, `stretch`
`align-items` ⬆️⬇️	Aligns items **vertically** within their grid cell.	Same values as `justify-items`
`justify-content` 🔄	Aligns the entire grid **horizontally** inside the container.	Same values as Flexbox
`align-content` ⬆️⬇️	Aligns the entire grid **vertically** inside the container.	Same values as Flexbox

Child (Item) Properties

Property	Description	Values
`grid-column`	Specifies which column(s) an item spans.	`start / end` (e.g., `1 / 3`)
`grid-row`	Specifies which row(s) an item spans.	Same as `grid-column`
`grid-area`	Assigns an item to a named area in the grid.	Matches the names in `grid-template-areas`.
`justify-self` 🔄	Aligns an item **horizontally** within its cell (overrides container).	`start`, `end`, `center`, `stretch`
`align-self` ⬆️⬇️	Aligns an item **vertically** within its cell (overrides container).	Same values as `justify-self`

🧠 Justify vs. Align (Flex/Grid)

Aspect	Justify	Align
Direction	Horizontal 🔄	Vertical ⬆️⬇️
Target	Items inside the container	Items or the entire container
When Used	Space distribution across the main axis	Space distribution along the cross axis

---

This cheat sheet should help you quickly reference and understand the key properties of Flexbox and Grid! 🎉