

# Módulo V - Elementos de Página

## Grid

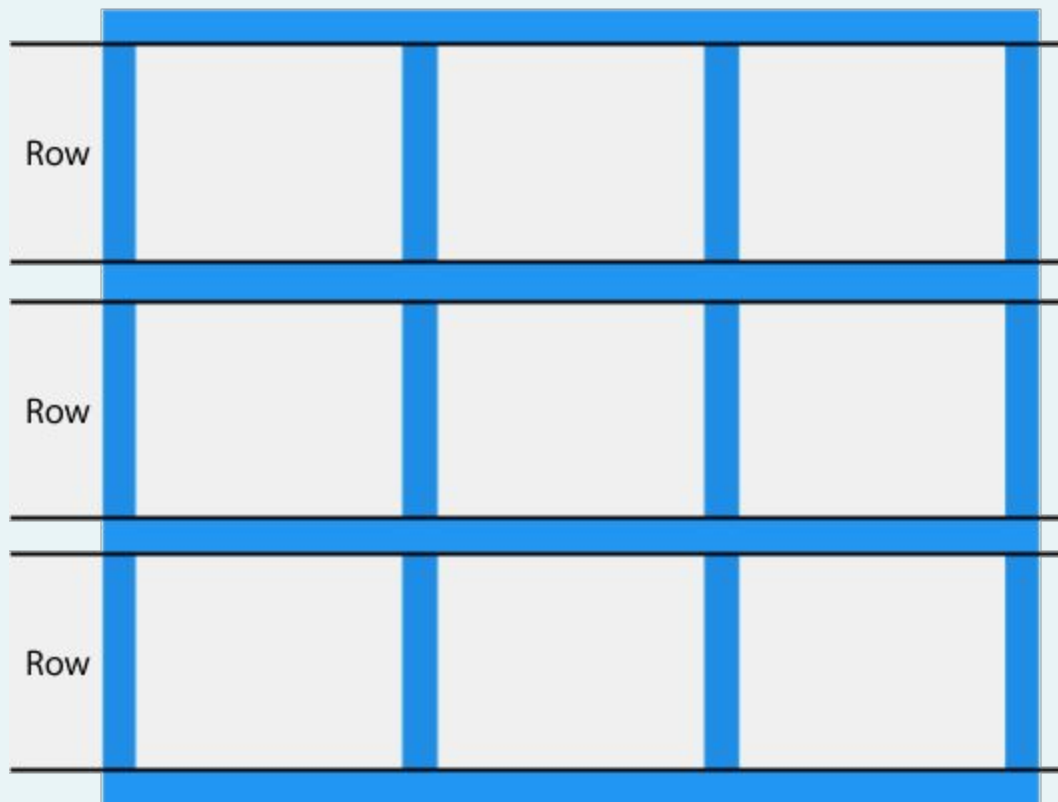
# Grid

- Diferente do Flexbox o sistema de Grid permite a definição de linhas/colunas
- Torna mais simples a criação de layouts sem usar float e posicionamento
- Visão geral
  - Grid container
  - Grid Items

# Colunas

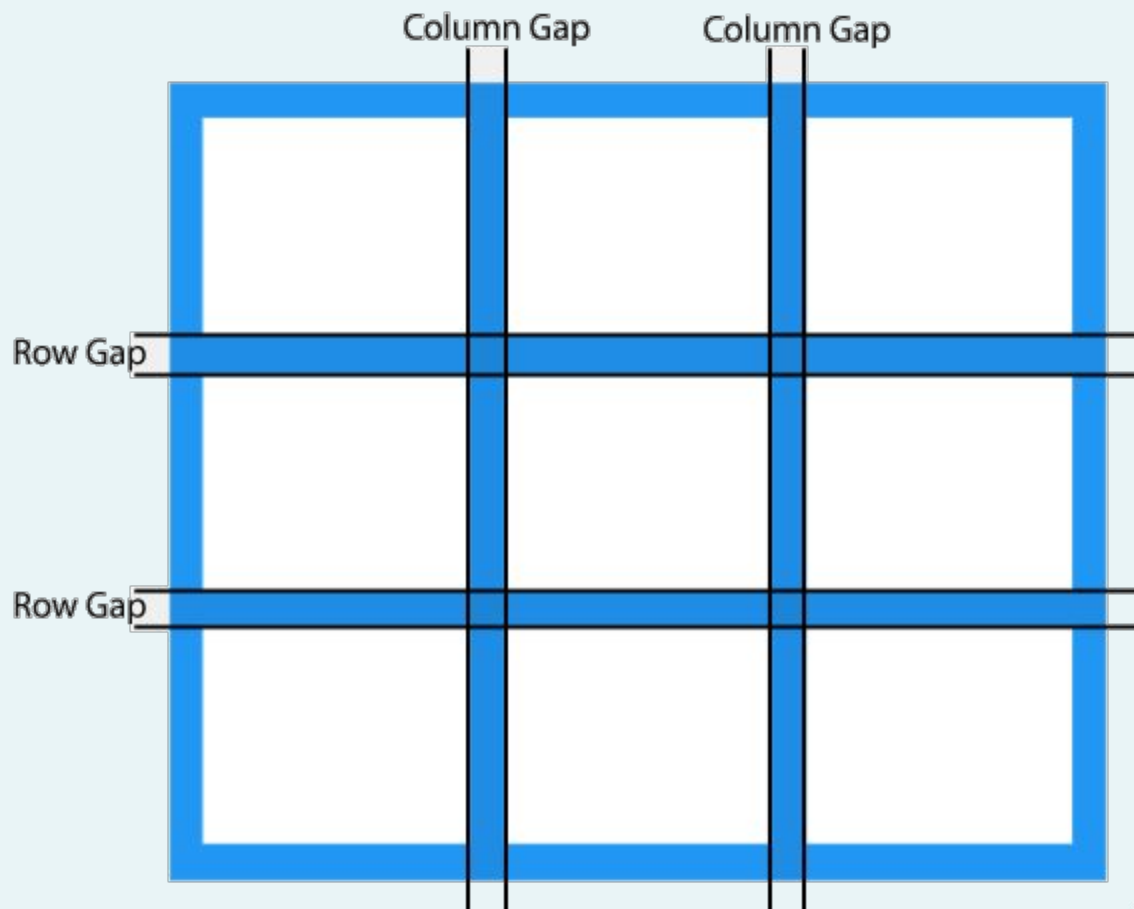
Column	Column	Column

# Linhas



Row			
Row			
Row			

# Espaços



# Grid container

- Definir um Grid container:
  - `.container { display: grid; }`
- Para definir o tamanho da grid utilize:
  - `grid-template-columns: 80px 200px auto 40px;`
  - `grid-template-rows: 60% auto auto;`

# Grid container (alinhamento horizontal)

- Define como os elementos deverão ser alinhados na horizontal
- A propriedade utilizada é **justify-content**:
  - `.container { display: grid; justify-content: center; }`
- Os valores possíveis são:
  - `space-evenly`
  - `space-around`
  - `space-between`
  - `center`
  - `start`
  - `end`

# Grid container (alinhamento vertical)

- Define como os elementos deverão ser alinhados na vertical
- A propriedade utilizada é **align-items**:
  - `.container { display: grid; align-items: center; }`
- Os valores possíveis são:
  - `center`
  - `space-evenly`
  - `space-around`
  - `space-between`
  - `start`
  - `end`



# Expandindo colunas

- Uma célula pode ocupar o espaço de outras células em diferentes colunas
- A propriedade **grid-column** define onde o item vai começar e terminar
- Ocupar 2 colunas (começando na 1 e terminando antes da 3)
  - `.item1 { grid-column: 1 / 3; }`
- Ocupar 2 colunas (começando na 3)
  - `.item2 { grid-column: 3 / span 2; }`

# Expandindo linhas

- Uma célula pode ocupar o espaço de outras células em diferentes linhas
- A propriedade **grid-row** define onde o item vai começar e terminar
- Ocupar 2 linhas (começando na 1 e terminando antes da 3)
  - `.item1 { grid-row: 1 / 3; }`
- Ocupar 2 colunas (começando na 3)
  - `.item2 { grid-row: 3 / span 2; }`

# Exemplos

- <https://codepen.io/rivolli/pen/VwQyKKv>