Programação de Aplicativos

Prof. Nicolas Marques dos Santos

Sistema Fiep FIEP SESI SENAI IEL

CSS

O CSS(Cascading Style Sheet) é uma tecnologia utilizada para estilizar/customizar páginas web.

CSS

O CSS depende do HTML para estilizar páginas web, pois utiliza de seus elementos para a estilização.

```
7       <link rel="stylesheet" href="css/main.css" />
8   </head>
9   <body>
10  <header id="header">
11      <img src="images/logo.png" height="246" width="246" alt="Logotipo" />
12      <nav id="header-nav" class="nav">
13          <ul>
14              <li class="fleft"><a href="">Home</a></li>
15              <li class="fleft"><a href="">Quem Somos</a></li>
16              <li class="fleft"><a href="">Portfólio</a></li>
17              <li class="fleft"><a href="">Produtos</a></li>
18              <li class="fleft"><a href="">Contato</a></li>
19          </ul>
20      </nav><!-- #nav -->
21  </header>
22  <article id="article">
```

Sistema Fiep FIEP SESI SENAI IEL

CSS - Tipos

Existem três maneiras de utilizar CSS dentro de um arquivo HTML:

- Inline
- Interno
- Externo

CSS - Inline

```
<html>
   <body>
      <p style="color: blue;">
      This is paragraph</p>
   </body>
</html>
```

CSS - Interno

```
<html>
  <head>
    <style>
      selector {
        property: value;
      }
    </style>
  </head>
</html>
```

Sistema Fiep FIEP SESI SENAI IEL

CSS - Externo

CSS - Seletores

Seletores são usados pra escolher quais elementos HTML você quer estilizar, além de também permitir como fazer esse estilo. Existem diversos tipos de seletores:

Tag	`p`, `h1`, `div`	Todos os elementos dessa tag
Classe	`.destaque`	Elementos com `class="destaque"`
ID	`#cabecalho`	Elemento com `id="cabecalho"`
Universal	`*`	Todos os elementos da página
Descendente	`div p`	`<p>` dentro de `<div>`
Filho direto	`ul > li`	`<li>` que é **filho direto** de `<ul>`
Pseudo-classe	`a:hover`	Estilo quando passa o mouse

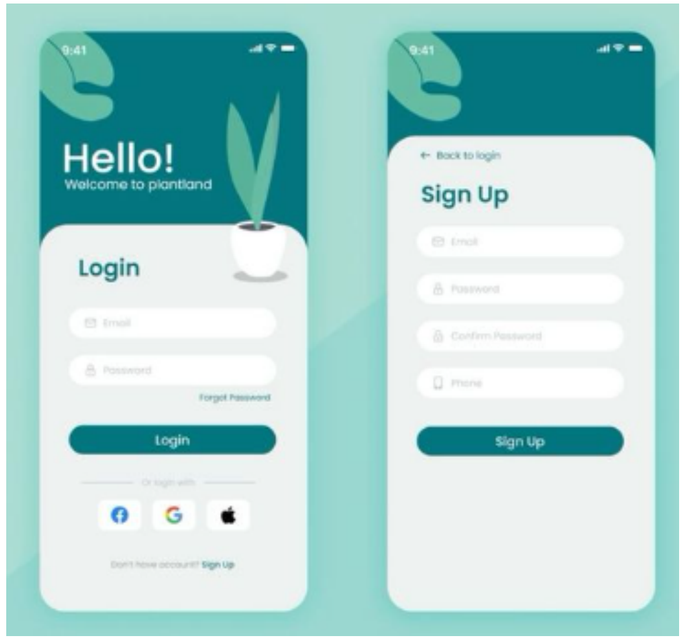
Sistema Fiep FIEP SESI SENAI IEL

CSS - Medidas

O CSS é responsável também por configurar o tamanho dos elementos ou seja, ele trabalho com Largura(Widht) e Altura(Height).

Espaçamento e alinhamento

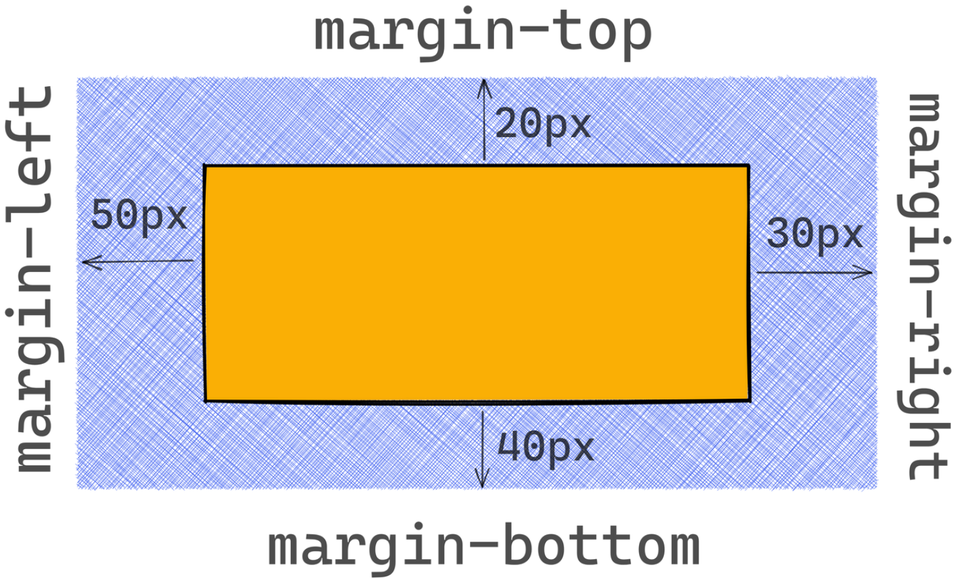
Quando falamos sobre espaçamento e alinhamento em websites, estamos lidando diretamente com aspectos da interface do usuário (UI). Esses elementos são fundamentais para organizar visualmente os componentes da página e proporcionar uma experiência mais agradável, intuitiva e funcional para quem navega.



Margin

Margin é a propriedade do CSS usada pra controlar o espaço fora da borda de um elemento. Ele empurra o elemento para longe dos outros.

```
margin: 20px 30px 40px 50px;
```

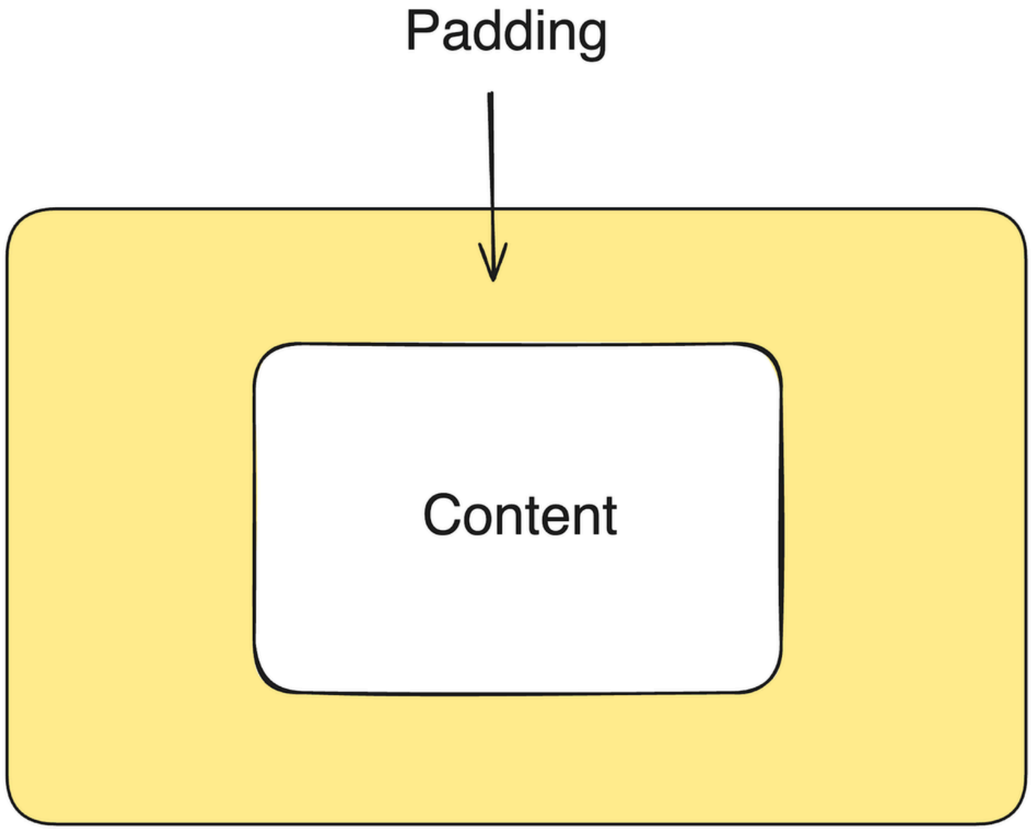


Margin: Uso

```
margin: 20px; /* todos os lados */
margin: 10px 20px; /* vertical (top/bottom) | horizontal (left/right) */
margin: 10px 15px 20px; /* top | horizontal | bottom */
margin: 10px 15px 20px 25px;  /* top | right | bottom | left */
```

Padding

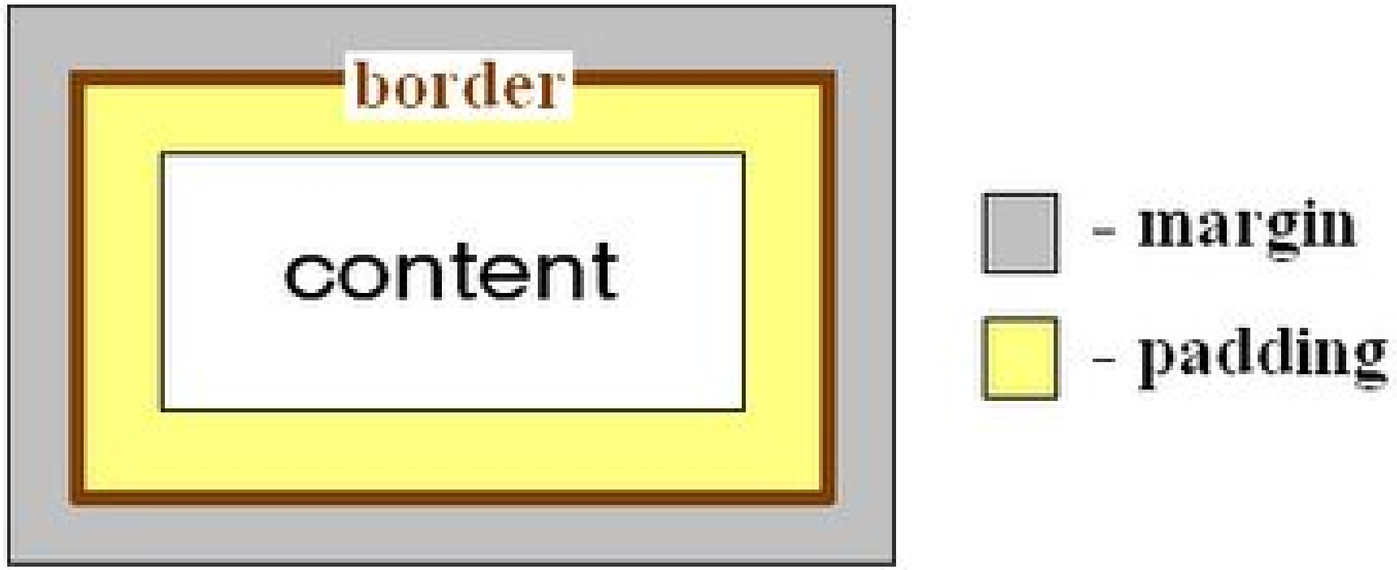
Diferente do margin, o padding é o espaço interno entre o conteúdo de um elemento e sua borda.



Padding: Uso

```
padding: 20px; /* aplica 20px em todos os lados */
padding: 10px 20px; /* vertical | horizontal */
padding: 10px 15px 20px; /* top | horizontal | bottom */
padding: 10px 15px 20px 25px; /* top | right | bottom | left */
```

Padding e Margin



Position

A propriedade position define como um elemento será posicionado no layout, em relação a outros elementos ou ao próprio viewport. Existem alguns tipos:

- static
- relative
- absolute
- sticky
- fixed

Position: Static

Este é o valor default, sem nenhum posicionamento especial. O elemento segue o fluxo normal da página (de cima pra baixo, da esquerda pra direita).

```css
.elemento {
  position: static;
}
```

Position: Relative

Posiciona o elemento relativamente à sua posição original. Ele ainda ocupa o espaço original, mas é “deslocado” visualmente.

```css
.elemento {
  position: relative;
  top: 20px;
  left: 30px;
}
```

Position: Absolute

Posiciona o elemento em relação ao ancestral mais próximo que tenha position: relative | absolute | fixed | sticky.

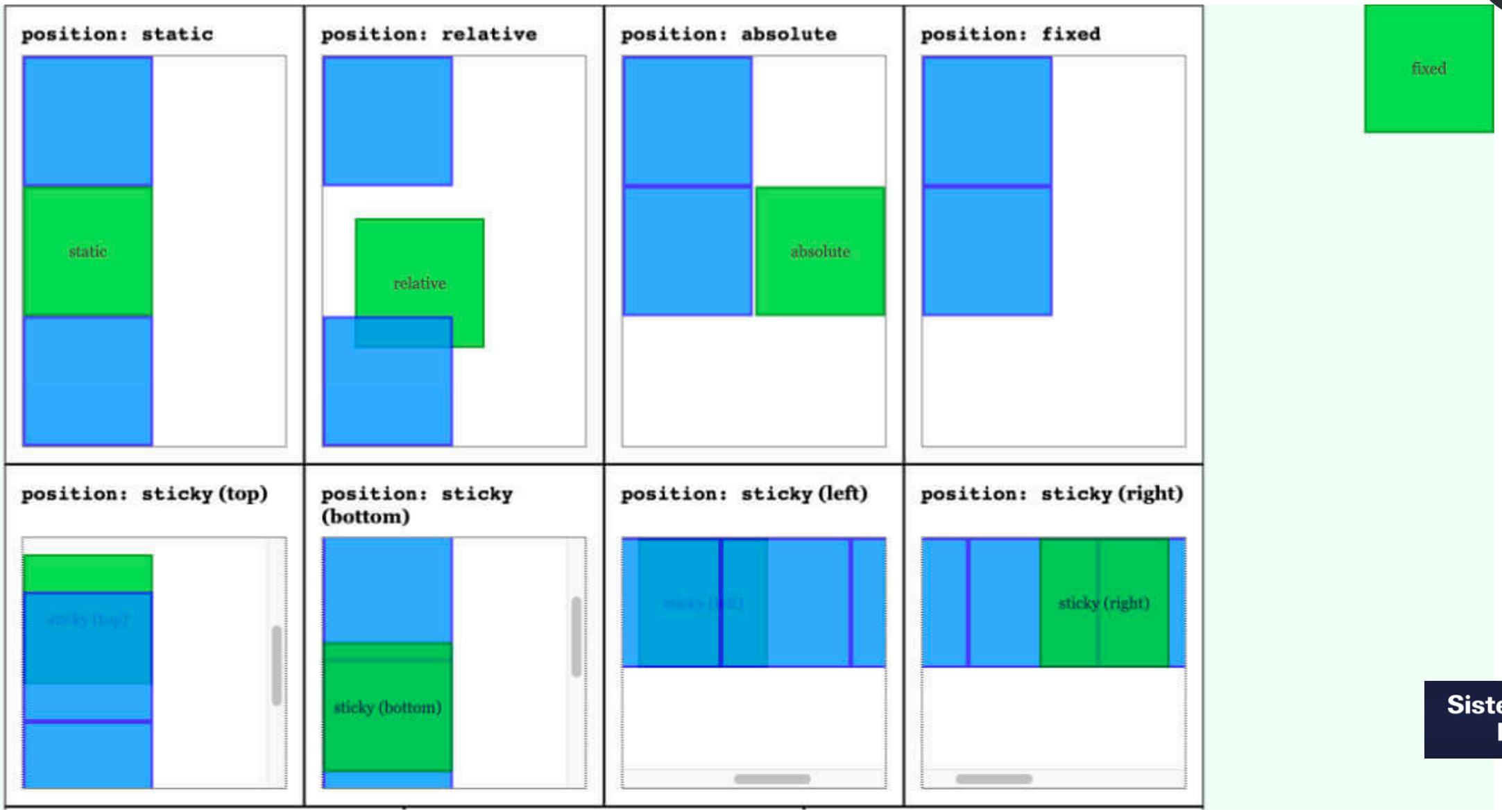
```css
.elemento {
  position: absolute;
  top: 10px;
  left: 10px;
}
```

Position: fixed

Fixa o elemento na tela, independente do scroll. É posicionado em relação ao viewport.

```css
.elemento {
  position: fixed;
  bottom: 0;
  right: 0;
}
```

Position: Exemplos



fixed

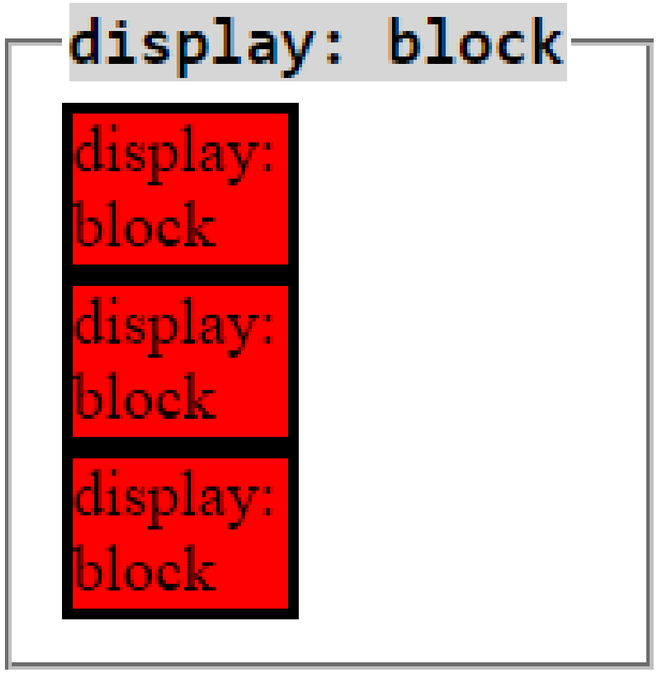
Display

A propriedade display define como um elemento deve ser exibido no layout, ou seja, seu comportamento de caixa (box model) e relacionamento com outros elementos.

- block
- inline
- inline-block
- none
- flex
- grid

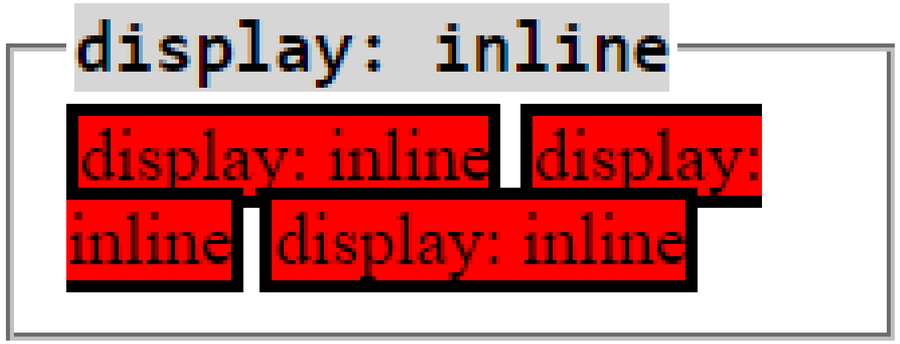
Display: block

- O elemento ocupa toda a largura disponível e inicia em uma nova linha.
- Exemplos que possuem esse comportamento por padrão: , <section>, <p>.
- Permite definir propriedades como width e height.



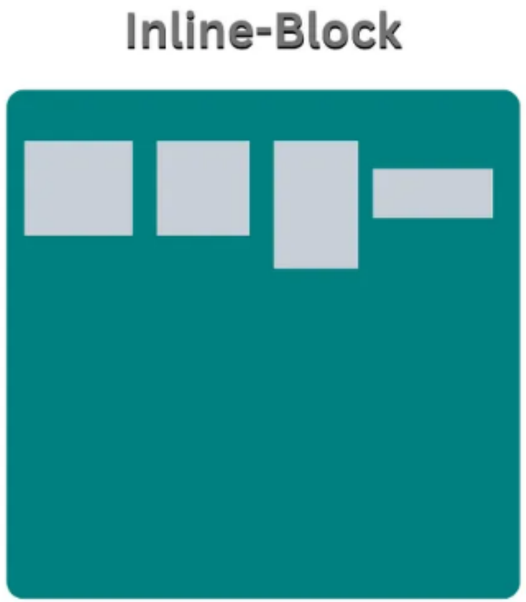
Display: inline

- O elemento não quebra linha e ocupa apenas o espaço necessário para o seu conteúdo.
- Exemplos: , <a>, .
- Não aceita largura e altura de forma convencional.



Display: inline-block

- Combina características de inline (ficar na mesma linha) com block (permitir ajuste de largura e altura).
- Frequentemente utilizado para elementos como botões ou componentes dispostos lado a lado.



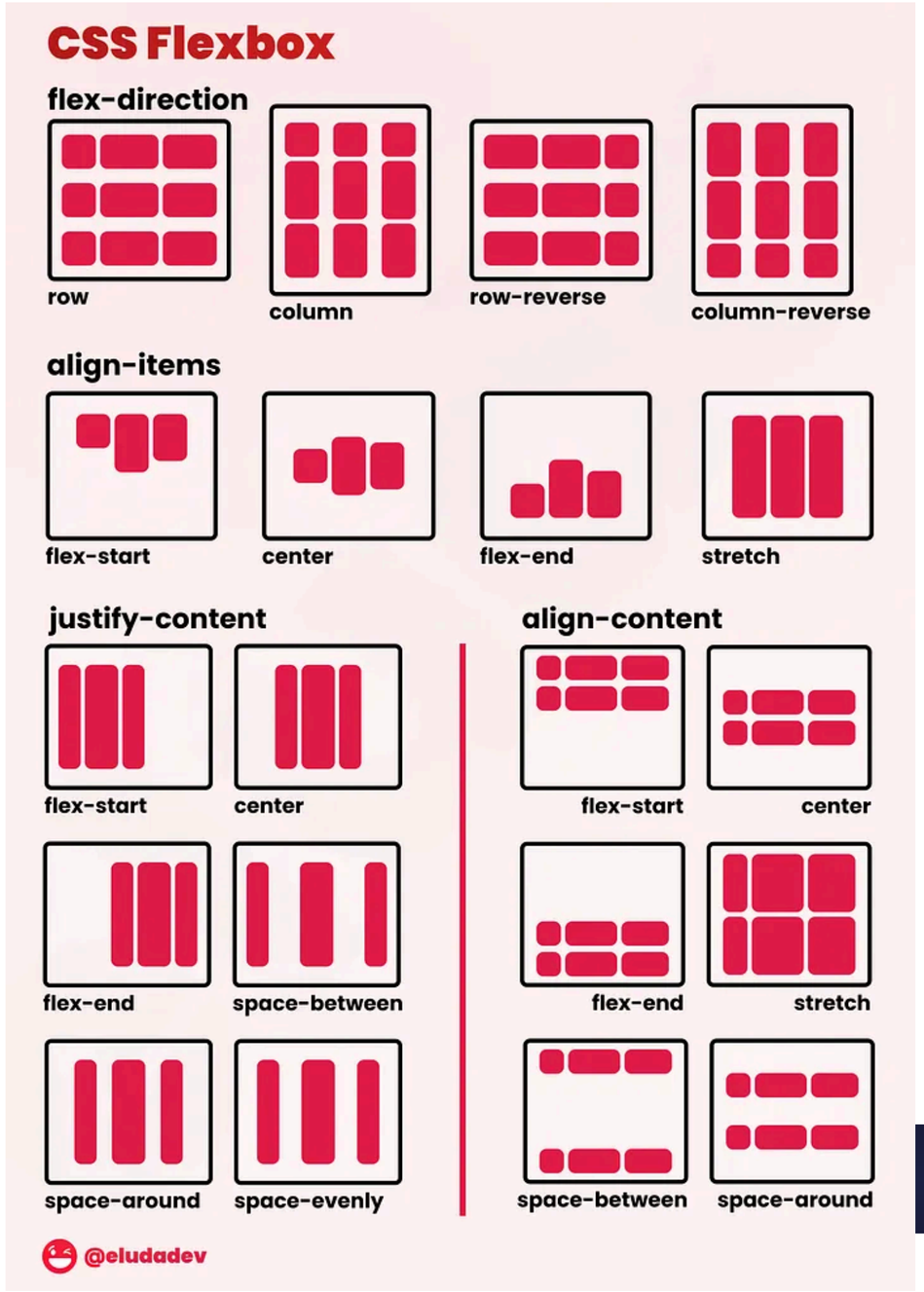
Display: none

- Remove completamente o elemento do fluxo do layout, como se ele não existisse no documento.
- Diferente de visibility: hidden, que mantém o espaço ocupado.

Display: flex

- Ativa o Flexbox, um modelo de layout unidimensional que facilita o alinhamento e a distribuição de espaço entre os elementos filhos.

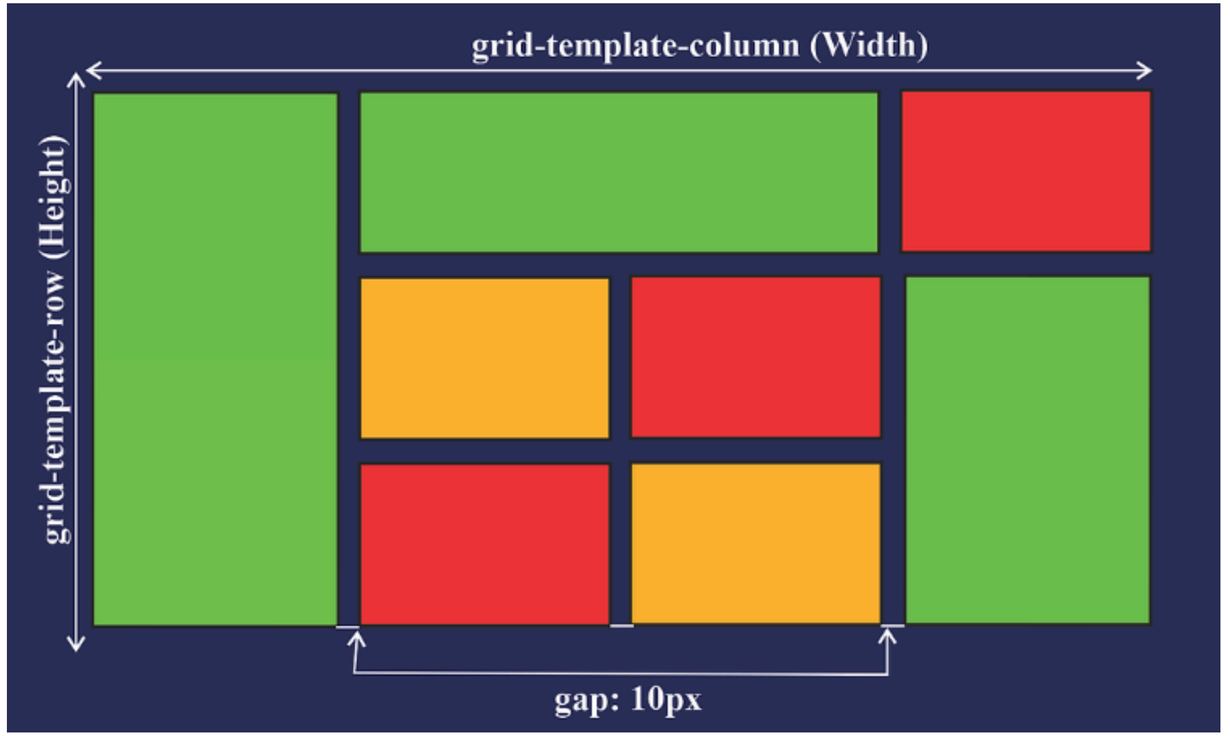
- Permite alinhamento vertical e horizontal de forma simplificada.



Sistema Fiep FIEP SESI SENAI IEL

Display: grid

- Ativa o CSS Grid, um sistema bidimensional que organiza elementos em linhas e colunas.
- Indicado para layouts mais complexos.



Display: grid

```html
<div class="grid">
  <div class="item">1</div>
  <div class="item">2</div>
  <div class="item">3</div>
  <div class="item">4</div>
  <div class="item">5</div>
  <div class="item">6</div>
  <div class="item">7</div>
  <div class="item">8</div>
  <div class="item">9</div>
</div>
```


```css
<style>
  body {
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
    background-color: #f0f0f0;
  }

  .grid {
    display: grid;
    grid-template-columns: repeat(3, 1fr);
    gap: 10px;
    width: 300px;
    height: 300px;
  }

  .item {
    background-color: #4a90e2;
    display: flex;
    justify-content: center;
    align-items: center;
    color: white;
    font-size: 1.5rem;
    font-weight: bold;
    border-radius: 10px;
  }
</style>
```

Sistema Fiep FIEP SESI SENAI IEL

OBRIGADO!

Sistema Fiep
FIEP
SESI
SENAI
IEL