* seletor de ID (#id) aplica estilo a um único elemento que possui um ID específico na página HTML
* seletor de classe (.classe) aplica estilo a todos os elementos que partilham a mesma classe
* o seletor de elemento (por exemplo, p ou div) seleciona todos os elementos de um determinado tipo (tag) na página, sem necessidade de atributos adicionais.

**Quando usar qual?**

* **ID:** Use para elementos que devem ser únicos e específicos na página, como um cabeçalho principal ou um formulário particular.
* **Classe:** Use para estilos que precisam ser aplicados a vários elementos de forma consistente, como botões, itens de lista ou links que se repetem.
* **Elemento:** Use para aplicar estilos gerais a todos os elementos de um tipo, definindo um estilo base para elementos como parágrafos ou divisões.

**1. Seletor de ID (#id)**

* **Propósito:** Identificar um único elemento na página.
* **Característica:** Cada ID deve ser único dentro de um documento HTML.
* **Notação no CSS:** Usa a hashtag (#) seguida pelo nome do ID.
* **Exemplo:**

#menu-principal {

background-color: #f0f0f0; /\* Estiliza o único elemento com id="menu-principal" \*/

}

**2. Seletor de Classe (.classe)**

* **Propósito:** Agrupar múltiplos elementos com estilos semelhantes, permitindo a aplicação do mesmo estilo a vários elementos.
* **Característica:** Um elemento pode ter várias classes, e uma classe pode ser aplicada a muitos elementos.
* **Notação no CSS:** Usa um ponto (.) seguido pelo nome da classe.
* **Exemplo:**

.botao {

padding: 10px 15px;

background-color: blue;

color: white;

}

**3. Seletor de Elemento (tag)**

* **Propósito:** Selecionar todos os elementos de um determinado tipo, como todos os parágrafos (p) ou todos os cabeçalhos (h1).
* **Característica:** Aplica estilos a todos os elementos de uma tag específica sem precisar de um ID ou classe.
* **Notação no CSS:** Usa o nome da tag HTML.
* **Exemplo:**

p {

font-family: Arial, sans-serif; /\* Estiliza todos os parágrafos da página \*/

}

**Comparação entre classe, id e seletor de elemento em CSS?**

**1. Classe (.nome)**

* Usada para **reutilizar estilos** em vários elementos.
* Começa com **.** no CSS.
* Um elemento pode ter **várias classes**.

**Exemplo:**

.vermelho {

color: red;

}

<p class="vermelho">Texto em vermelho</p>

<p class="vermelho">Outro texto também em vermelho</p>

**2. ID (#nome)**

* Usado para identificar **um único elemento** na página.
* Começa com **#** no CSS.
* Deve ser **único** (cada id só pode ser usado uma vez no documento).

**Exemplo:**

#cabecalho {

background-color: lightgray;

}

<div id="cabecalho">Cabeçalho único da página</div>

**3. Seletor de elemento (tag)**

* Aplica estilo diretamente a **todas as tags** daquele tipo.
* **Não precisa de classe nem id**.

**Exemplo:**

p {

font-size: 18px;

}

<p>Esse parágrafo recebe o estilo</p>

<p>E este também</p>

**Resumindo:**

| **Tipo** | **Sintaxe** | **Reuso** | **Restrição** |
| --- | --- | --- | --- |
| **Classe** | .classe | Pode ser usada em vários elementos | Um elemento pode ter várias classes |
| **ID** | #id | Deve ser único | Um elemento só pode ter **um id** |
| **Elemento** | tag | Afeta todos os elementos dessa tag | Não distingue casos individuais |

**Dica prática:**

* Use **classes** para **estilização em grupo**.
* Use **id** para identificar **um único elemento específico** (geralmente usado em JavaScript ou âncoras).
* Use **seletor de elemento** para estilos **globais**.

**Exemplo classe, id e seletor de elemento sendo usados ao mesmo tempo:**

**CSS (style.css)**

/\* Seletor de elemento - afeta todos os <p> \*/

p {

font-size: 18px;

color: black;

}

/\* Classe - pode ser usada em vários elementos \*/

.destaque {

color: blue;

font-weight: bold;

}

/\* ID - único, aplicado só a um elemento \*/

#especial {

color: red;

font-size: 22px;

text-decoration: underline;

}

**HTML**

<!DOCTYPE html>

<html lang="pt-BR">

<head>

<meta charset="UTF-8">

<title>Exemplo CSS</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<!-- Exemplo de seletor de elemento -->

<p>Este parágrafo recebe o estilo aplicado a TODOS os &lt;p&gt;.</p>

<!-- Exemplo de classe -->

<p class="destaque">Este parágrafo usa a classe "destaque".</p>

<p class="destaque">Outro parágrafo com a mesma classe "destaque".</p>

<!-- Exemplo de id -->

<p id="especial">Este é único e usa o id "especial".</p>

</body>

</html>

**O que vai acontecer:**

* Todos os <p> terão fonte de 18px (regra do seletor de elemento).
* Os parágrafos com class="destaque" ficarão **azuis e em negrito**.
* O parágrafo com id="especial" ficará **vermelho, maior e sublinhado**, independente de outros estilos.

**Prioridade entre classe, id e elemento (quem “vence” quando estilos se sobrepõem)?**

**Especificidade (Prioridade** entre classe, id e elemento) em CSS, e define **quem vence** quando mais de uma regra tenta estilizar o mesmo elemento.

**Ordem de prioridade (do mais fraco para o mais forte):**

1. **Seletor de elemento** (ex: p, div, h1)
2. **Classe** (ex: .destaque, .azul)
3. **ID** (ex: #especial)
4. **Estilo inline** (ex: <p style="color:green;">)
5. **!important** (força a regra a ter a maior prioridade — mas deve ser usado com cuidado)

**Exemplo prático**

**CSS**

/\* seletor de elemento \*/

p {

color: black;

}

/\* classe \*/

.destaque {

color: blue;

}

/\* id \*/

#exemplo {

color: red;

}

**HTML**

<p id="exemplo" class="destaque">Texto de teste</p>

Resultado: **o texto ficará vermelho** porque o **id** tem mais prioridade do que a classe ou o seletor de elemento.

**Se colocarmos inline**

<p id="exemplo" class="destaque" style="color: green;">Texto de teste</p>

Resultado: **verde**, porque o estilo **inline** ganha do id.

**Se usarmos !important**

.destaque {

color: blue !important;

}

Resultado: **azul**, porque o !important sobrepõe até o id e o inline (a menos que o inline também tenha !important).

**Resumo da hierarquia:**

Elemento < Classe < ID < Inline < !important

**Exemplo que mostra cada nível de prioridade em ação.**

**CSS (style.css)**

/\* 1. Seletor de elemento \*/

p {

color: black;

font-size: 18px; }

/\* 2. Classe \*/

.classe {

color: blue; }

/\* 3. ID \*/

#unico {

color: red; }

/\* 5. Classe com !important \*/

.importante {

color: green !important; }

**HTML (index.html)**

<!DOCTYPE html>

<html lang="pt-BR">

<head>

<meta charset="UTF-8">

<title>Exemplo de Especificidade CSS</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<h2>Exemplo de Especificidade em CSS</h2>

<!-- 1. Só seletor de elemento -->

<p>Sou afetado apenas pelo seletor <code>p</code>.</p>

<!-- 2. Classe -->

<p class="classe">Tenho a classe <code>.classe</code>.</p>

<!-- 3. ID -->

<p id="unico">Tenho o ID <code>#unico</code>.</p>

<!-- 4. Inline -->

<p id="inline" class="classe" style="color: orange;">

Tenho estilo <code>inline</code>, que vence os anteriores.

</p>

<!-- 5. !important -->

<p id="forcado" class="classe importante" style="color: purple;">

Tenho <code>!important</code> na classe, que vence até o inline.

</p>

</body>

</html>

**O que acontece na tela:**

1. O primeiro <p> → **preto** (vem do seletor p).
2. O segundo <p class="classe"> → **azul** (classe vence o seletor de elemento).
3. O terceiro <p id="unico"> → **vermelho** (id vence a classe e o elemento).
4. O quarto <p style="color: orange;"> → **laranja** (inline vence id, classe e elemento).
5. O quinto <p class="importante"> → **verde** (porque !important vence até o inline).

**Diferença entre id e name**

**id="usuario"**

* **Identificador único** de um elemento no documento.
* Cada id deve aparecer **apenas uma vez** em toda a página.
* Usado para:
  + Aplicar estilos via **CSS** (#usuario { ... }).
  + Selecionar o elemento via **JavaScript** (document.getElementById("usuario")).
  + Criar **âncoras** ou referências internas.

**name="usuario"**

* **Nome do campo** usado em **formulários**.
* Esse valor é enviado ao servidor quando o formulário é submetido.
* Pode ser repetido em mais de um campo (ex.: checkbox de múltipla escolha).
* É usado pelo backend para identificar **o dado enviado**.

**Exemplo:**

<form action="/login" method="post">

<label for="usuario">Usuário:</label>

<input type="text" id="usuario" name="usuario">

<label for="senha">Senha:</label>

<input type="password" id="senha" name="senha">

<button type="submit">Entrar</button>

</form>

**O que acontece aqui:**

* O id="usuario" permite que o <label for="usuario"> saiba a qual input ele pertence.
* O name="usuario" faz com que, ao enviar o formulário, o servidor receba algo assim:

usuario=valorDigitado&senha=valorDigitado

* id → único, usado para **identificação no HTML, CSS e JavaScript**.
* name → serve para **enviar os dados** ao servidor em formulários.