

# CSS - CASCADING STYLE SHEETS

Introdução e Fundamentos

**Isa Firmino**

Dev Full stack  
Alpha Lumen



**Lay Lima**

Dev Full stack  
Alpha Lumen

# LEMBRETES DA AULA

Aqui estão algumas regras e lembretes para ter em mente.

**1**

Silencie seu microfone para não atrapalhar o entendimento de outros alunos.

**2**

Ligue sua câmera.

**3**

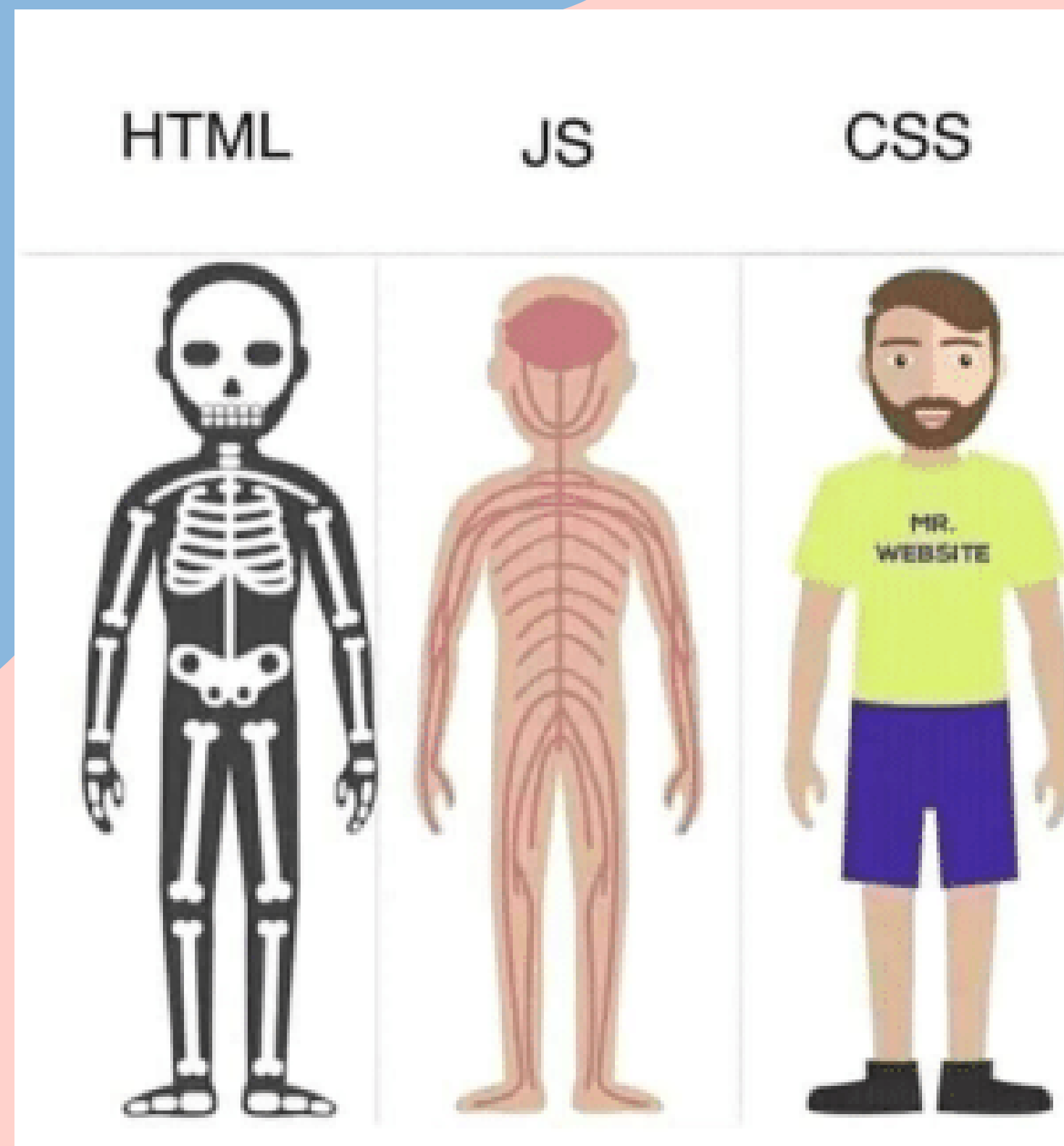
Perguntas serão respondidas ao final da aula.

# TÓPICOS DA AULA DE HOJE

Nossa agenda desta tarde.

- O que é o CSS e sua sintaxe;
- Conceito de cascata;
- Conceito de Herança;
- Tipos de estilo e ordem de prioridade;
- Seletores Id, class e tag.
- Demonstração prática;

# O QUE É?



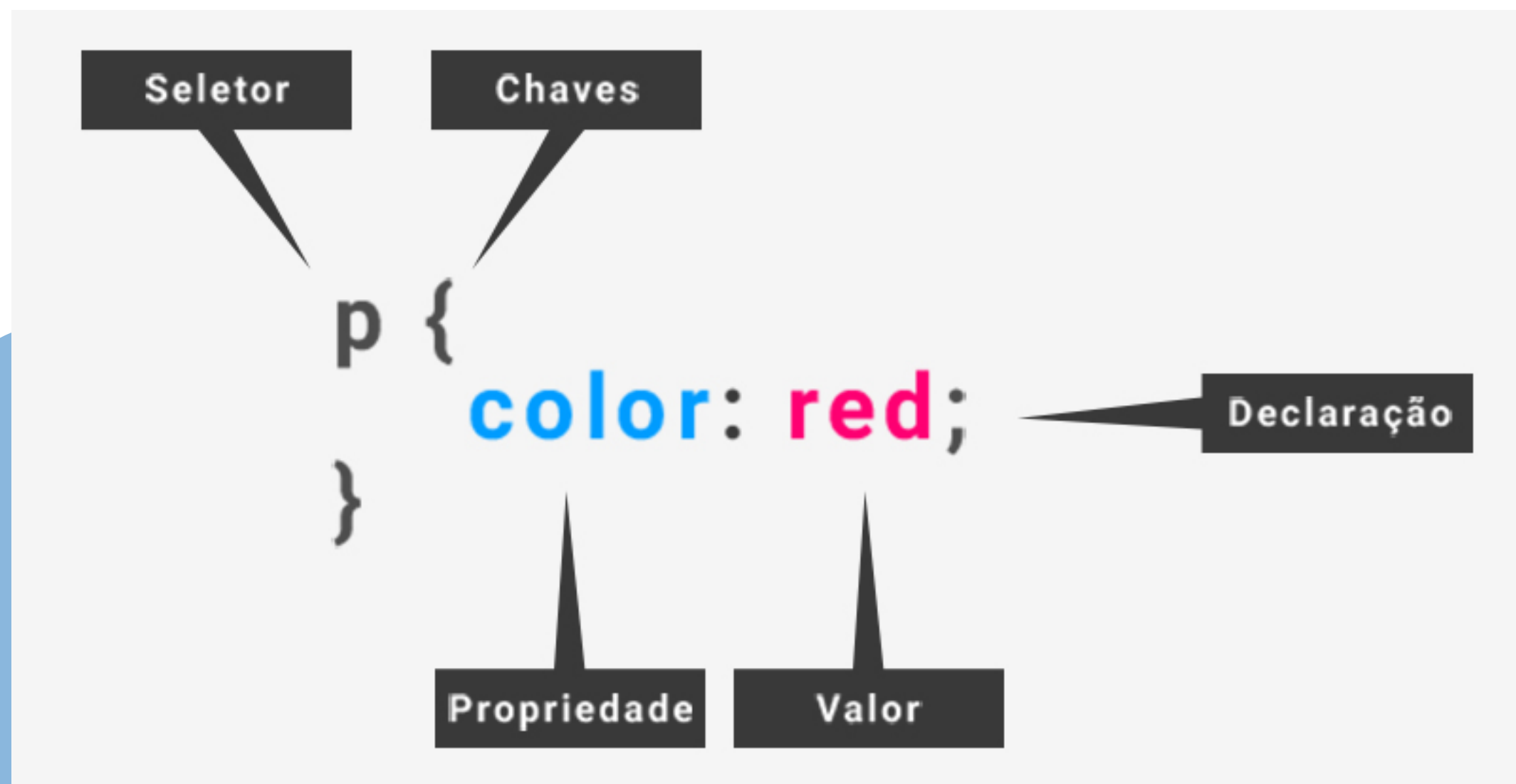
## O que é o CSS?

Podemos fazer uma breve comparação...

## Como usamos?

Iremos ver a seguir...

# SINTAXE



# CONCEITO DE CASCATA

CSS significa Cascading Stylesheets  
(Folhas de Estilo em Cascata).

A cascata é o a forma do CSS resolver  
conflitos quando várias regras de CSS  
se aplicam a um elemento HTML.



# EXEMPLO

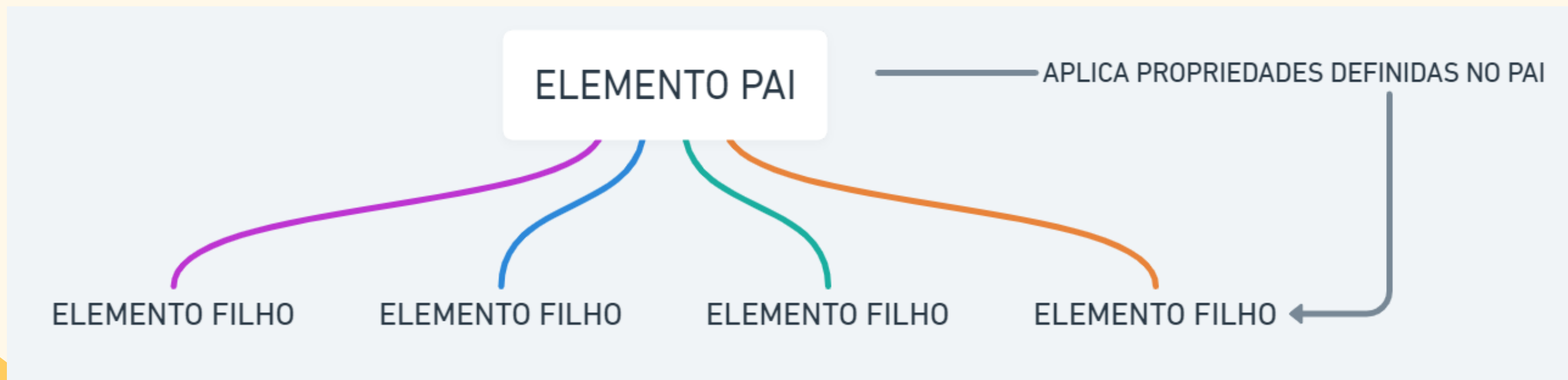
É por isso que o texto do botão estilizado com o seguinte CSS será azul.

```
button {  
  color: red;  
}
```

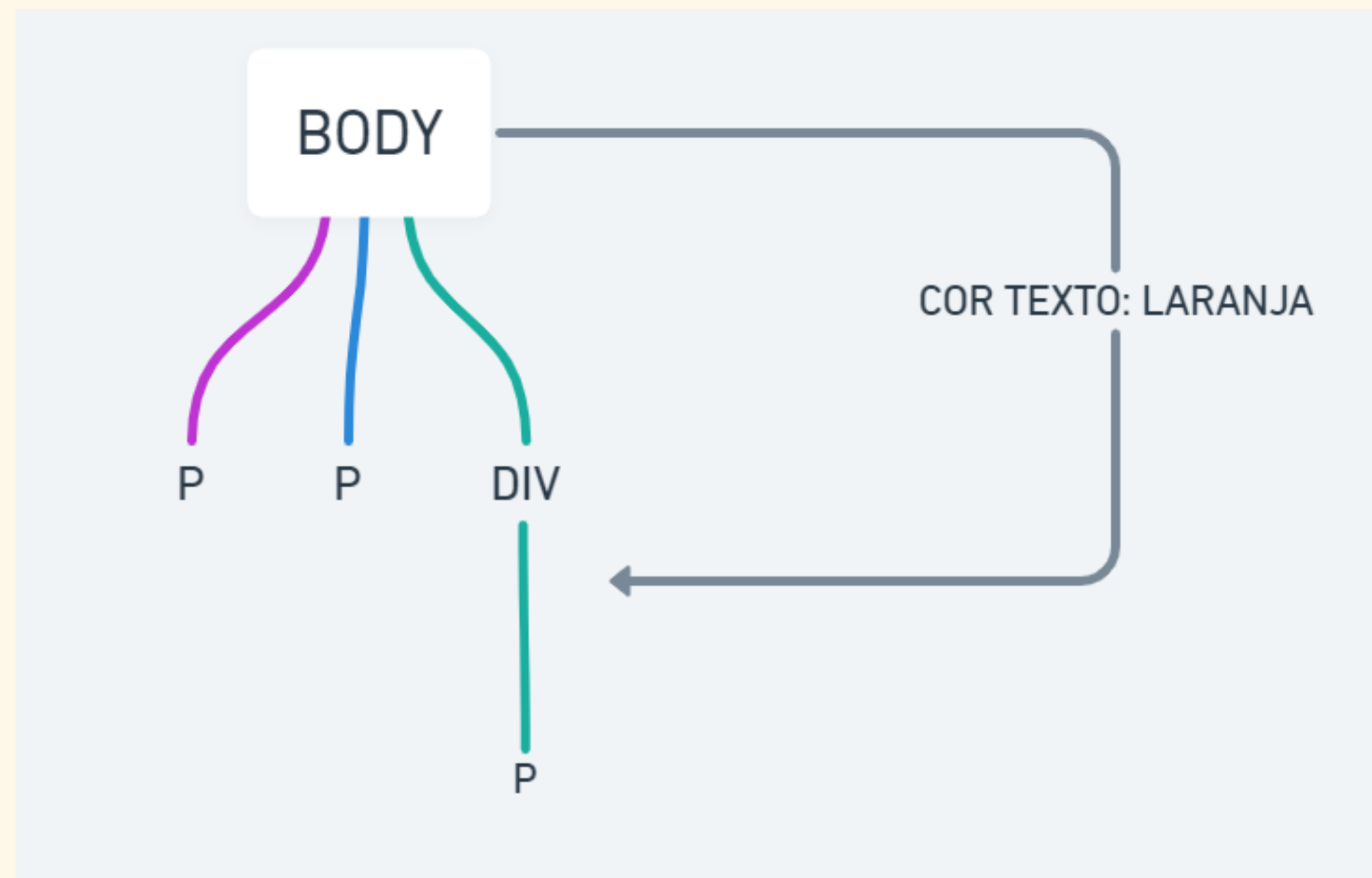
```
button {  
  color: blue;  
}
```

# CONCEITO DE HERANÇA

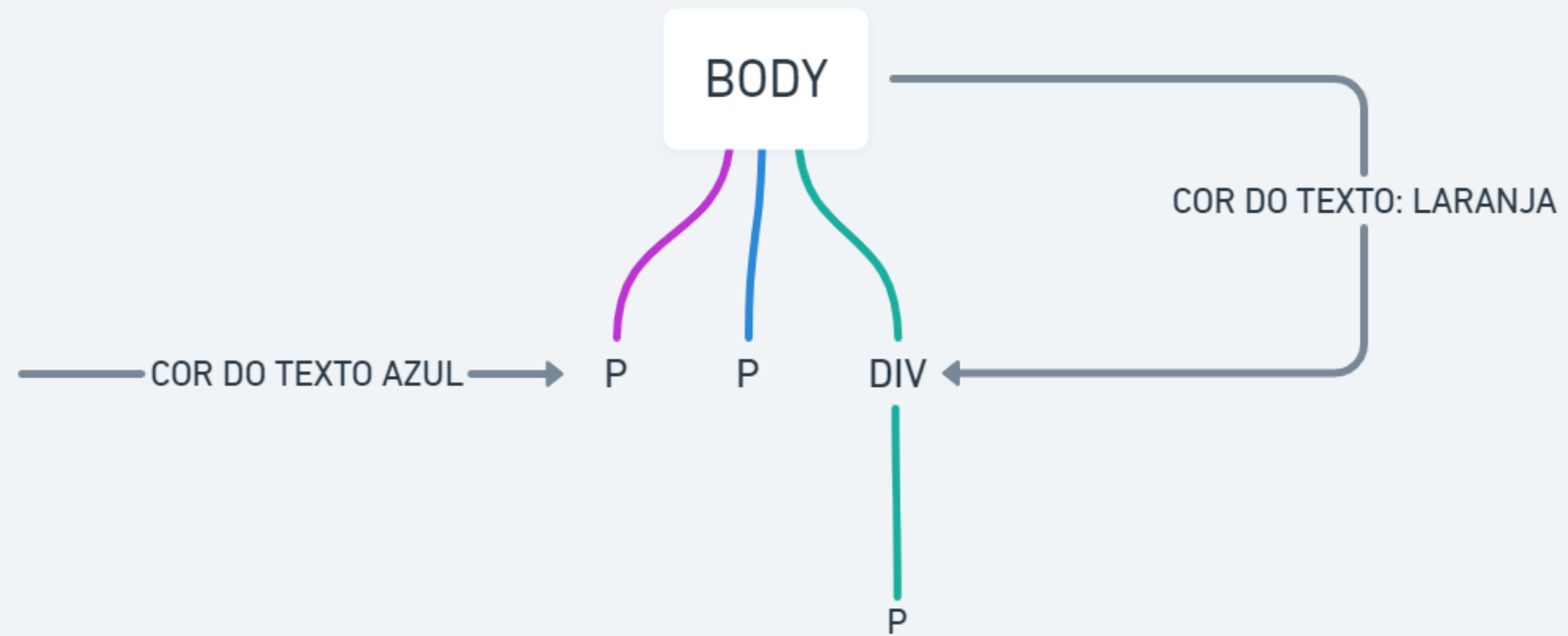
Os filhos herdam as mesmas propriedades dos seus pais, a não ser que uma regra seja definida especificamente para eles.







# MAS...



jsfiddle

# MANEIRAS DE INSERÇÃO E ORDEM DE PRIORIDADE

- Inline
- Internal
- External

# INLINE

É inserido diretamente da tag de abertura HTML.

```
<!-- Inline CSS Styles -->
```

```
<h1 style="font-weight: bold; font-size: 42px;">
```

```
I will be bold and 42 pixels!
```

```
</h1>
```

```
<p style="color: blue;">I will be blue!</p>
```

# INTERNAL

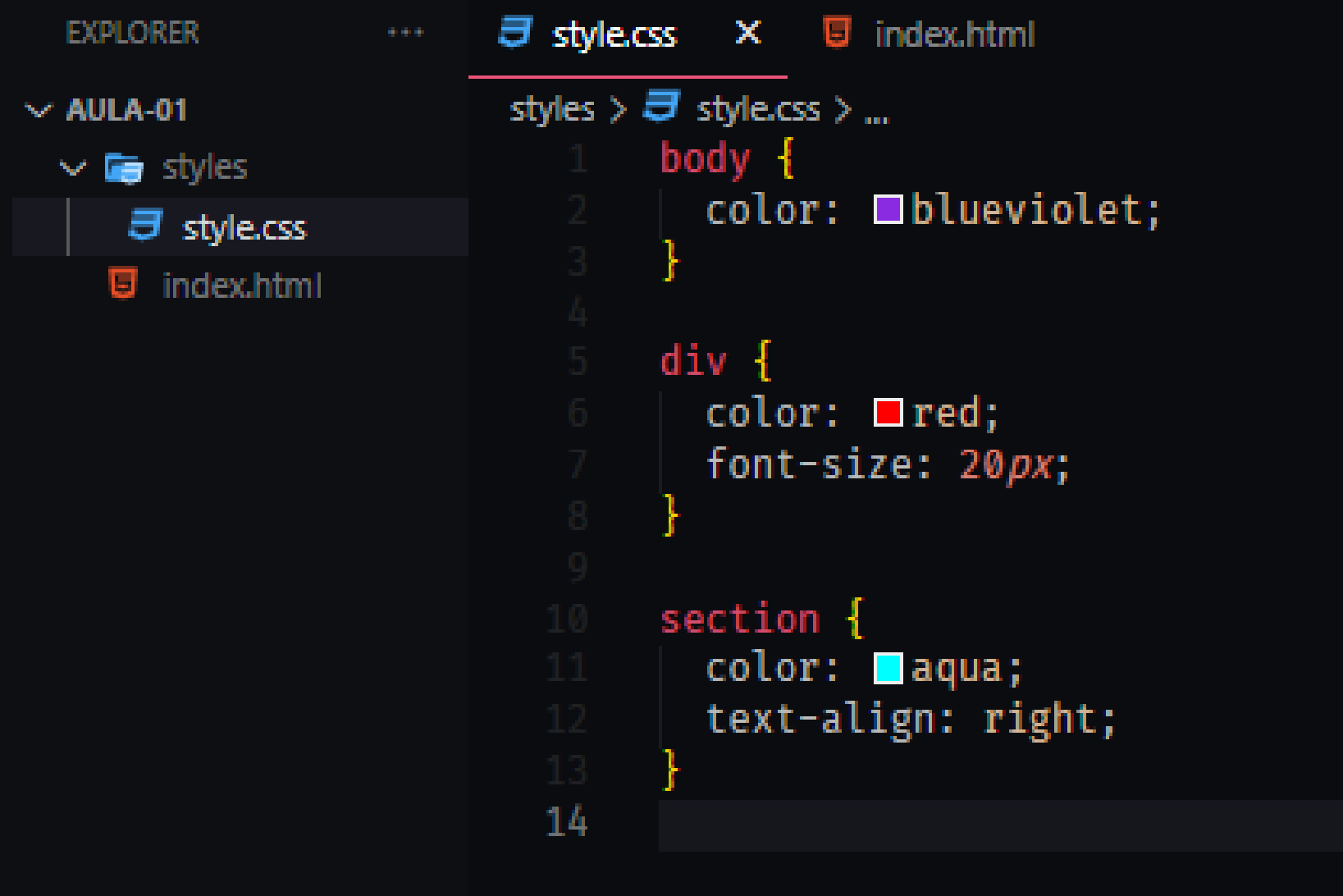
Utiliza a tag HTML style diretamente no arquivo html.

```
6 <meta name= viewport content= width=device width, initial
7 <title>Aula 1</title>
8 <link rel="stylesheet" href="styles/style.css" />
9 </head>
10
11 <style>
12   body {
13     color: ■ orange;
14   }
15 </style>
16
17 <body>
18   <p style="color: ■ green">Propriedades herdadas</p>
19
```

# EXTERNAL

Utiliza uma folha de estilo externa com a extensão .css. Esse arquivo será integrado com a tag Link no html, pelo atributo href.

```
<title>Aula 1</title>
<link rel="stylesheet" href="styles/style.css" />
</head>
```



The screenshot shows the Visual Studio Code interface. On the left, the 'EXPLORER' sidebar displays a project named 'AULA-01' with a subfolder 'styles' containing 'style.css' and 'index.html'. The main editor area shows the 'style.css' file with the following CSS rules:

```
1  body {
2    color: blueviolet;
3  }
4
5  div {
6    color: red;
7    font-size: 20px;
8  }
9
10 section {
11   color: aqua;
12   text-align: right;
13 }
14
```

# ORDEN DE PRIORIDADE

INLINE > INTERNAL > EXTERNAL

«/»

CSS INLINE TEM A PRIORIDADE  
MAIS ALTA.

«/»

EM SEGUIDA VEM  
INTERNAL/EMBEDDED

{ }

SEGUIDO DE EXTERNAL, O QUAL  
TEM ÚLTIMA PRIORIDADE.

.JS

OBS.: O ESTILO INSERIDO NO  
JAVASCRIPT FUNCIONA COMO  
INLINE E SOBRESCREVE ESTILOS  
DE PROPRIEDADES DEFINIDAS  
ANteriormente.

# SELETOR POR TAG/ELEMENTO

Usado para selecionar elementos por seus nomes de tags HTML. Ex: p, div, section, h1, h2, h3...

```
h1 {  
  color: blue;  
}
```



# SELETORES DO TIPO CLASS E ID

## CLASS

Um elemento pode ter mais de uma classe e essa classe pode se repetir em mais de um elemento.

## ID

É único, não pode se repetir e nem ter mais de um.

# TIPO CLASS

Atribuição no html pelo atributo **class** e identificado no css por um seletor iniciado por um "."

```
<span class="classeElementosSpan">  
  PRIMEIRO TEXTO  
</span>  
  
<span class="classeElementosSpan">  
  SEGUNDO TEXTO  
</span>
```


```
.classeElementosSpan {  
  font-size: 30px;  
}
```

JSfiddle

# TIPO ID

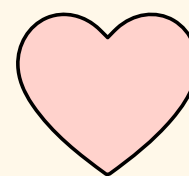
Atribuição no html pelo atributo id e identificado no css por um seletor iniciado por um "#"

```
<h1 id="identificador">  
  EXEMPLOS DE SELETORES  
</h1>
```

```
#identificador {  
  color:  pink;  
}
```

JSfiddle

# CONCLUSÃO E DEMONSTRAÇÃO



OBRIGADA PELA ATENÇÃO!