

REVISÃO PROGRAMAÇÃO VISUAL

II UNIDADE

Professor: Marcos Brandão

Versão: 2026

Baseada nos conceitos de componentes visuais de páginas Web e nos tópicos fundamentais de CSS (Cascading Style Sheets), incluindo seletores, classes, IDs, cores, fontes, bordas, posicionamento, navegação, imagens e layout responsivo.

SUMÁRIO

1. Introdução à Programação Visual
 2. HTML e CSS
 3. Componentes Visuais de Páginas Web
 4. Seletores CSS
 5. Classes e IDs
 6. Cores e Fontes
 7. Bordas, Margens e Espaçamentos
 8. Imagens e Links
 9. Menus de Navegação
 10. Posicionamento de Elementos
 11. Pseudoclasses e Hover
 12. Layout Responsivo
 13. Boas Práticas de CSS
-

1. INTRODUÇÃO À PROGRAMAÇÃO VISUAL

A Programação Visual é a área responsável pela aparência e organização dos elementos de uma página Web.

Ela define:

- Cores

- Fontes
- Tamanhos
- Espaçamentos
- Posicionamento
- Layout das páginas

O principal recurso utilizado é o CSS (Cascading Style Sheets).

2. HTML E CSS

HTML

Responsável pela estrutura da página.

Exemplo:

```
<h1>Meu Site</h1>  
<p>Bem-vindo!</p>
```

CSS

Responsável pela aparência visual.

Exemplo:

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

O HTML define o conteúdo e o CSS define a apresentação visual.

3. COMPONENTES VISUAIS DE PÁGINAS WEB

Os principais componentes visuais são:

- Títulos
- Textos
- Imagens

- Menus
- Botões
- Formulários
- Tabelas
- Links

Esses elementos compõem a interface do usuário.

4. SELETORES CSS

Os seletores indicam quais elementos receberão formatação.

Seletor de Tag

```
p{  
  color:red;  
}
```

Aplica a todos os parágrafos.

Seletor Universal

```
*{  
  margin:0;  
}
```

Aplica a todos os elementos.

5. CLASSES E IDS

Classe (class)

Permite aplicar o mesmo estilo a vários elementos.

HTML:

```
<p class="destaque">Texto</p>
```

CSS:

```
.destaque{
  color:red;
}
```

ID

Identifica um elemento único.

HTML:

```
<h1 id="titulo">Site</h1>
```

CSS:

```
#titulo{
  color:blue;
}
```

Classes e IDs são recursos fundamentais para organização do CSS.

6. CORES E FONTES

Cor do texto

```
color: blue;
```

Cor de fundo

```
background-color: yellow;
```

Fonte

```
font-family: Arial;
```

Tamanho da fonte

```
font-size: 20px;
```

Negrito

font-weight: bold;

7. BORDAS, MARGENS E ESPAÇAMENTOS

Borda

border: 1px solid black;

Margem Externa

margin: 20px;

Espaçamento Interno

padding: 15px;

Esses recursos ajudam na organização visual da página.

8. IMAGENS E LINKS

Inserir imagem

HTML:

```

```

Alterar tamanho

CSS:

```
img{  
  width:300px;  
}
```

Link

HTML:

```
<a href="https://www.google.com">Google</a>
```

9. MENUS DE NAVEGAÇÃO

Os menus facilitam a navegação do usuário.

Exemplo:

```
<ul>  
<li>Home</li>  
<li>Produtos</li>  
<li>Contato</li>  
</ul>
```

O CSS pode ser utilizado para transformar listas em menus horizontais.

10. POSICIONAMENTO DE ELEMENTOS

Static

Posicionamento padrão.

```
position: static;
```

Relative

Posicionamento relativo.

```
position: relative;
```

Absolute

Posicionamento absoluto.

```
position: absolute;
```

Fixed

Elemento permanece fixo na tela.

```
position: fixed;
```

11. PSEUDOCASSES E HOVER

Permitem criar efeitos visuais.

Exemplo:

```
a:hover{  
  color:red;  
}
```

Quando o mouse passa sobre o link, sua cor muda.

12. LAYOUT RESPONSIVO

Um layout responsivo adapta-se a diferentes dispositivos:

- Computadores
- Tablets
- Smartphones

Benefícios:

- Melhor experiência do usuário
- Compatibilidade com dispositivos móveis
- Melhor aproveitamento da tela

O CSS é fundamental para criação de layouts responsivos.

13. BOAS PRÁTICAS DE CSS

- Utilizar arquivos CSS externos.

- Organizar seletores.
 - Utilizar classes para reutilização.
 - Evitar estilos inline.
 - Padronizar cores e fontes.
 - Utilizar nomes significativos para classes.
-