



---

# Guia de Estudo: HTML e CSS – 15 Exercícios por Nível de Complexidade

---

## Nível Baixo (Iniciante)

---

### 1. Página de Perfil Simples



#### Enunciado:

Crie uma página com seu nome, uma imagem de perfil e uma breve descrição.



#### Passo a passo:

1. Crie `index.html` e adicione: `<h1>`, `<img>`, `<p>`.
2. Crie `style.css` e conecte com `<link>`.
3. Centralize o texto e use uma fonte diferente com `font-family`.



#### Saída esperada:

- Nome grande no topo, imagem abaixo, e uma descrição em parágrafo.
- Tudo centralizado e com uma fonte agradável.



[HTML Básico – MDN](#)

---

### 2. Lista de Compras



#### Enunciado:

Crie duas listas: uma com os itens de mercado e outra com os passos para preparar um prato.



#### Passo a passo:

1. Use `<ul>` para lista de compras e `<ol>` para o passo a passo.
2. Adicione itens com `<li>`.
3. Estilize com CSS para cores diferentes nas listas.



#### Saída esperada:

- Duas listas estilizadas, com itens bem visíveis.



[Listas HTML – W3Schools](#)



---

### 3. Parágrafo Estilizado



#### Enunciado:

Crie um parágrafo com um estilo visual agradável usando CSS.



#### Passo a passo:

1. Adicione `<p>` com texto.
2. No CSS, aplique: `background-color`, `color`, `font-family`, `padding`.



#### Saída esperada:

- Um parágrafo com fundo colorido, texto branco (ou escuro), fonte customizada e espaçamento interno.



[Texto com CSS – W3Schools](#)

---

---



### Nível Moderado

---

### 4. Navegação com Âncoras



#### Enunciado:

Crie um menu que leve o usuário a diferentes seções da mesma página.



#### Passo a passo:

1. Use `<nav>` com links `<a href="#secao1">`.
2. Crie `<section id="secao1">` para cada destino.
3. Estilize com cores e espaçamentos.



#### Saída esperada:

- Menu fixo no topo; ao clicar, a página rola para a seção correta.



[Âncoras – DevMedia](#)



---

## 5. Tabela de Horários



### Enunciado:

Crie uma tabela com os horários de aula da semana.



### Passo a passo:

1. Use `<table>`, `<tr>`, `<th>`, `<td>`.
2. Aplique bordas, `border-collapse`, e `text-align` com CSS.



### Saída esperada:

- Uma tabela clara, com colunas para dias e linhas para horários.



[Tabelas HTML – MDN](#)

---

## 6. Formulário de Contato



### Enunciado:

Crie um formulário com nome, e-mail, mensagem e botão de envio.



### Passo a passo:

1. Use `<form>` com `<input>`, `<textarea>`, `<button>`.
2. Adicione `required` para validação.
3. Estilize com `flex-direction: column`.



### Saída esperada:

- Campos organizados verticalmente e botão estilizado.
- Validação nativa ao tentar enviar vazio.



[Formulários – W3Schools](#)

---



---

## Nível Médio

---

### 7. Layout com Flexbox



#### Enunciado:

Crie um layout com três colunas lado a lado que se adaptam no celular.



#### Passo a passo:

1. Crie um `<div class="container">` com três `<div class="coluna">` dentro.
2. Use `display: flex` e `justify-content`.
3. Adicione `media queries` para empilhar no mobile.



#### Saída esperada:

- Três colunas horizontais no desktop, empilhadas no mobile.



[Flexbox – CSS Tricks](#)

---

### 8. Cartões de Produto



#### Enunciado:

Monte três cartões de produto com imagem, título, preço e botão.



#### Passo a passo:

1. Crie `divs` com conteúdo interno.
2. Estilize com `box-shadow`, `padding`, `border-radius`, `hover`.



#### Saída esperada:

- Cartões organizados com imagens e botões. Destaques visuais ao passar o mouse.



[Product Cards – W3Schools](#)



---

## 9. Cabeçalho Fixo com Scroll



### Enunciado:

Crie um cabeçalho fixo e conteúdo rolável com bastante texto.



### Passo a passo:

1. Use `<header>` com `position: fixed` e `z-index`.
2. Abaixo, adicione uma `<main>` com conteúdo extenso.
3. Use `padding-top` para evitar sobreposição.



### Saída esperada:

- Cabeçalho sempre visível, mesmo ao rolar.



[CSS Position – MDN](#)

---

## ● Nível Alto

---

## 10. Página Responsiva com Grid



### Enunciado:

Crie uma landing page com grid no desktop e layout flexível no celular.



### Passo a passo:

1. Crie seções estruturais: `<header>`, `<main>`, `<footer>`.
2. Use `display: grid` com `grid-template-areas`.
3. Adicione `media queries` para ajustar ao mobile.



### Saída esperada:

- Layout organizado no desktop, adaptado e empilhado no mobile.



[CSS Grid – CSS Tricks](#)



---

## 11. Tema Claro/Escuro



### Enunciado:

Adicione um botão que alterna entre modo claro e escuro com variáveis CSS.



### Passo a passo:

1. Declare variáveis `--bg-color`, `--text-color` no `:root`.
2. Crie duas classes: `.claro` e `.escuro`.
3. Use JavaScript para alternar a classe no `<body>`.



### Saída esperada:

- Um botão que muda dinamicamente as cores da página.



[Modo Escuro com CSS – Dev.to](#)

---

## 12. Portfólio com Animações



### Enunciado:

Crie um site com animações suaves ao rolar e interagir com elementos.



### Passo a passo:

1. Crie seções como "Sobre", "Projetos", "Contato".
2. Use `@keyframes`, `transition`, `hover`, e `scroll-behavior`.
3. Aplique efeitos ao rolar e ao passar o mouse.



### Saída esperada:

- Elementos aparecendo suavemente ao rolar.
- Animações sutis nos botões e cards.



[Animações CSS – MDN](#)



---

## 15. Projeto Final: Mini Portfólio com HTML e CSS

---

### Enunciado:

Crie um mini site de portfólio fictício aplicando todos os conceitos vistos anteriormente. O site deve conter:

- Página inicial com cabeçalho fixo
- Seção de perfil
- Lista de serviços
- Tabela de disponibilidade
- Formulário de contato
- Galeria com carrossel
- Tema claro/escuro
- Layout responsivo (grid ou flex)
- Animações simples de entrada
- Footer

---

### Estrutura de Pastas:

```
swift
CopiarEditar
15-projeto-final/
├── index.html
├── style.css
├── script.js
├── /img/
│   ├── perfil.jpg
│   ├── projeto1.jpg
│   └── projeto2.jpg
```