

Aula 15

CSS avançado

Mídia Queries e Design Responsivo



- ▶ Com a variedade de dispositivos disponíveis atualmente, o Design Responsivo não é apenas uma recomendação, mas uma necessidade. As Mídia Queries são uma ferramenta poderosa que permite aos desenvolvedores adaptar seus designs a qualquer tamanho de tela, garantindo que os usuários tenham a melhor experiência possível. Seja você um defensor do mobile-first ou do desktop-first, o importante é garantir que o site seja acessível e funcional em todos os dispositivos.



O que são Mídia Queries?

- ▶ Mídia Queries são ferramentas que permitem aos desenvolvedores aplicar estilos específicos com base nas características do dispositivo do usuário, como tamanho da tela, resolução, orientação (retrato ou paisagem) e outros.
- ▶ Elas são parte da especificação CSS3 e permitem que você defina condições para aplicar estilos específicos.

```
@media (max-width: 600px) {  
  body {  
    background-color: lightblue;  
  }  
}
```

- ▶ No exemplo acima, o corpo da página terá uma cor de fundo lightblue se a largura da tela for de 600 pixels ou menos.



Design Responsivo

- ▶ Com o aumento da diversidade de dispositivos e tamanhos de tela, tornou-se fundamental adaptar a interface do usuário para oferecer uma boa experiência de usuário em todos os dispositivos.
- ▶ Usar Mídia Queries permite adaptar o layout, fontes, imagens e outros elementos do design para diferentes tamanhos de tela.



Design Móvel Primeiro (Mobile-First) vs. Desktop Primeiro (Desktop-First)

- ▶ **Mobile-First:** O conceito por trás do "mobile-first" é projetar o site inicialmente para dispositivos móveis e depois adaptá-lo para telas maiores. A ideia é garantir que o site seja totalmente otimizado para a experiência móvel desde o início. Em termos de Mídia Queries, isso significa definir estilos padrão (para dispositivos móveis) e usar Mídia Queries para estilos de tela maiores.
- ▶ **Desktop-First:** Contrário ao Mobile-First, o Desktop-First envolve a criação de um design para telas grandes primeiro. Posteriormente, você adapta esse design para tamanhos de tela menores. Em termos de Mídia Queries, você define estilos padrão para desktop e usa Mídia Queries para ajustar estilos para telas menores.

Recomendações



- ▶ Para a maioria dos projetos modernos, o design mobile-first é recomendado porque:
 - O tráfego móvel tem crescido constantemente e, em muitos casos, supera o tráfego de desktop.
 - Começar pelo mobile ajuda a focar no essencial, dada a limitação de espaço.
 - Os sites geralmente carregam mais rapidamente em dispositivos móveis quando projetados com uma abordagem mobile-first.



Exemplos:

Adaptando o tamanho da fonte:

Suponhamos que você queira que o tamanho da fonte do seu título principal seja maior em dispositivos desktop, mas um pouco menor em dispositivos móveis.

```
/* Estilo padrão (desktop-first) */  
h1 {  
    font-size: 36px;  
}  
/* Adaptação para dispositivos com  
tela menor que 768px (por exemplo,  
tablets) */  
@media (max-width: 768px) {  
    h1 {  
        font-size: 28px;  
    }  
}  
/* Adaptação para dispositivos com  
tela menor que 480px (por exemplo,  
smartphones) */  
@media (max-width: 480px) {  
    h1 {  
        font-size: 24px;  
    }  
}
```



Exemplos:

Alterando a Disposição das Colunas:

Vamos imaginar um layout com duas colunas no desktop. No mobile, você quer que elas fiquem uma abaixo da outra.

```
/* Estilo padrão (desktop-first) */
```

```
.column {  
    float: left;  
    width: 50%;  
}
```

```
/* Adaptação para dispositivos móveis */
```

```
@media (max-width: 768px) {  
    .column {  
        float: none;  
        width: 100%;  
    }  
}
```




```
/* Estilo padrão (desktop-  
first) */
```

```
.sidebar {  
    display: block;  
}
```

Exemplos:

Ocultando Elementos em Dispositivos Móveis:

As vezes, pode haver elementos em seu site que não são necessários ou adequados para visualização em dispositivos móveis.

```
/* Escondendo a sidebar em  
dispositivos com tela menor  
que 768px */
```

```
@media (max-width: 768px) {  
    .sidebar {  
        display: none;  
    }  
}
```



```

```

Exemplos:

Adaptando Imagens:

Para economizar largura de banda e otimizar o carregamento da página em dispositivos móveis, você pode querer mostrar versões mais pequenas de suas imagens.

```
/* Estilo padrão (desktop-first)
*/
.responsive-image {
    width: 100%;
    height: auto;
}

/* Carregando uma imagem menor
para dispositivos com tela menor
que 768px */
@media (max-width: 768px) {
    .responsive-image {
        content: url('small-
image.jpg');
    }
}
```

Exemplos:

Mudando a Cor de Fundo com base na Orientação:

Suponha que você queira alterar a cor de fundo de um elemento com base na orientação do dispositivo (retrato ou paisagem).

```
/* Estilo padrão */  
.container {  
    background-color: lightblue;  
}  
  
/* Cor de fundo diferente para  
orientação paisagem */  
@media (orientation: landscape) {  
    .container {  
        background-color:  
lightcoral;  
    }  
}
```



Exercício

- ▶ Crie uma página com um container flexível e vários itens flexíveis dentro.
- ▶ Experimente diferentes propriedades e valores para ver como os itens se reorganizam e se adaptam.
- ▶ Tente replicar componentes comuns como barras de navegação, rodapés, galerias de imagens e cards usando apenas Flexbox.

Repositório

- ▶ Após realizar a criação do seu website, coloque ele no seu repositório, através do aplicativo do GitHub.

 · **Fecomércio**  **MT** · **Sindicatos** ·  **Senac** ·  **IPF-MT**

Sistema Comércio

Fecomércio  **MT** · **Sindicatos** ·  **Senac** ·  **IPF-MT**

Sistema Comércio