

**Professores:**

Celso Giusti

Daniel Manoel Filho

Isadora Dias Afonso

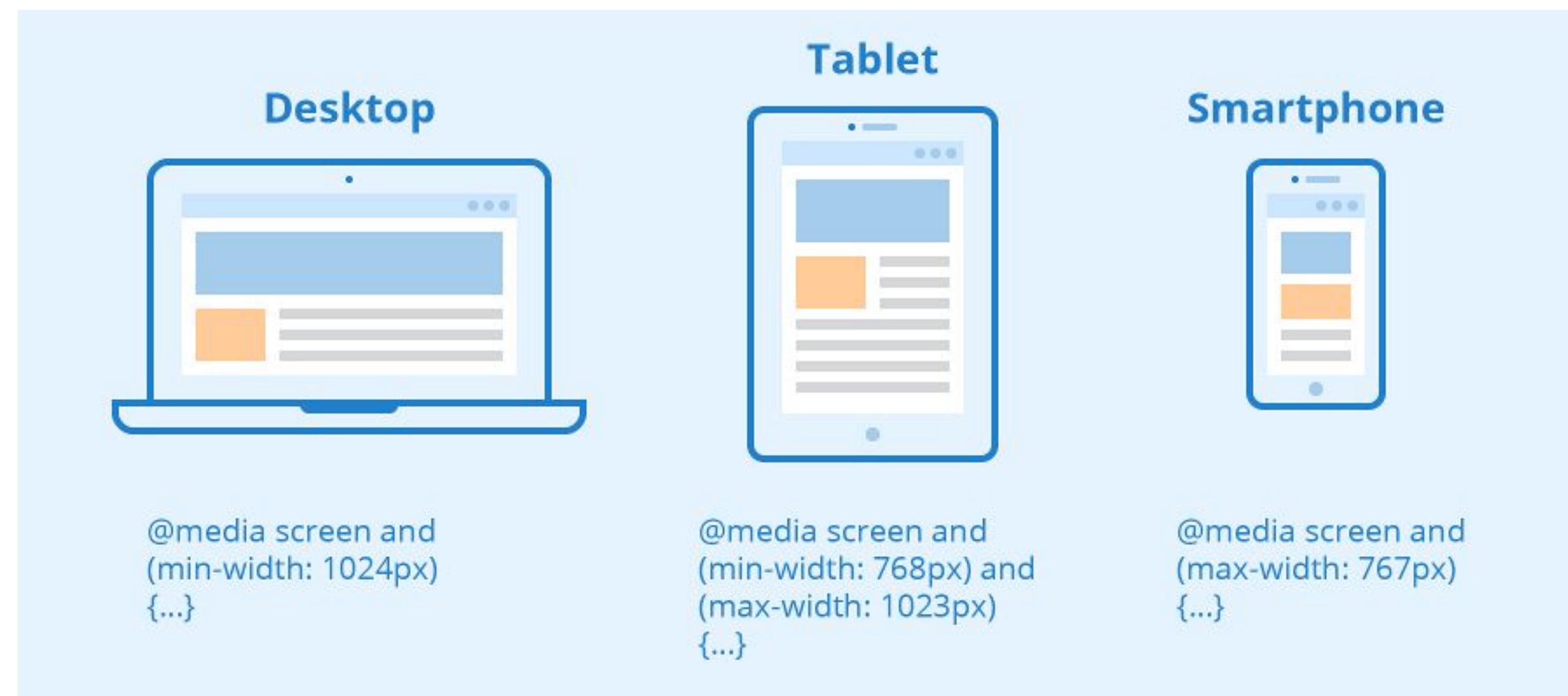
Marlon Palata Fanger Rodrigues

# Design Responsivo



# O que é Design Responsivo?

É uma abordagem de design e desenvolvimento web que cria sites que fornecem uma experiência de **visualização ideal** em **diversos dispositivos** (celulares, tablets e desktops).



# Aplicação: A Tag Viewport

**O primeiro passo para a responsividade** Sem essa linha de código no seu **<head>**, o navegador do celular vai tentar "espremer" o site de desktop na tela pequena, deixando as letras minúsculas.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

- **width=device-width**: Diz que a largura do site deve ser a mesma da tela do aparelho.
- **initial-scale=1.0**: Define o zoom inicial como padrão.

# Media Queries

É um **recurso do CSS** que permite aplicar estilos apenas se certas condições forem atendidas (como a largura da tela ser menor que 600px).

## Estrutura Básica:

```
@media (condição) {  
    /* O código CSS aqui dentro só funciona se a condição for verdade */  
}
```

**Ponto Chave:** Ela permite que você mude cores, tamanhos e, principalmente, a estrutura do seu **Grid** dependendo do espaço disponível.

# Tipos de Mídia (Media Types)

**Para onde o estilo vai?** Antes de definir tamanhos, indicamos para qual meio de saída o estilo se aplica:

- **all**: Valor padrão. Vale para todos os tipos de dispositivos.
- **print**: Estilos que só funcionam quando o usuário imprime a página em papel.
- **screen**: Estilos focados em telas digitais (computadores, tablets e celulares).

# Exemplo Básico Media Query

**Mudança de Estado** O exemplo mais simples é mudar a cor de fundo do site quando a tela diminui para o tamanho de um celular:

```
/* Cor padrão para Computador */  
body { background-color: white; }  
  
/* Quando a tela for menor que 600px */  
@media (max-width: 600px) {  
    body { background-color: lightblue; }  
}
```

O navegador lê o CSS normal e, quando chega na **@media**, ele faz uma pergunta: "A tela é menor que tal valor?". Se sim, ele aplica o que está lá dentro, **substituindo** o que veio antes.

# Operadores Lógicos em Media Queries

**Refinando a busca** Podemos combinar condições para sermos mais específicos:

- **and**
- **not**
- **only**
- **vírgula (,)**

# Operador Lógico: AND (E)

O operador **and** combina várias condições. O CSS só será aplicado se **todas** as condições forem verdadeiras.

## Aplicação (Foco em Tablets):

```
/* Aplicar apenas se a tela estiver entre 601px E 1024px */
@media screen and (min-width: 601px) and (max-width: 1024px) {
    .vitrine-grid {
        grid-template-columns: 1fr 1fr; /* Muda para 2 colunas no Tablet */
    }
}
```

**Explicação:** O estilo só será aplicado se a **tela** tiver no mínimo **601px** E no máximo **1024px**. É ideal para isolar o layout de tablets.

# Operador Lógico: VÍRGULA (OR)

A **vírgula** funciona como o operador lógico "OU". Se **qualquer uma** das condições for verdadeira, o estilo é aplicado.

## Aplicação (Mobile ou Orientação Retrato):

```
/* Aplica se a tela for menor que 600px OU se estiver em pé */
@media (max-width: 600px), (orientation: portrait) {
    .vitrine-grid { grid-template-columns: 1fr; }
}
```

**Explicação:** O Grid vira 1 coluna se a tela for menor que 600px OU se o celular estiver "em pé" (portrait).

# Operador Lógico: ONLY

O operador **only** serve para impedir que navegadores muito antigos (que não entendem Media Queries) apliquem os estilos de forma errada.

## Aplicação:

```
/* Apenas para telas que suportam Media Queries */  
@media only screen and (max-width: 480px) {  
    h1 { font-size: 1.2rem; }  
}
```

**Explicação:** Garante que "apenas" dispositivos modernos processem a regra.

# Operador Lógico: NOT (NÃO)

O operador **not** nega uma condição. Ele aplica o estilo em todos os casos, **exceto** naquele que você especificou.

## Aplicação (Escondendo na Impressão):

```
/* Aplica se não for uma tela (print) */  
@media not screen {  
    #menu,  
    .btn-pedido {  
        display: none;  
    }  
}
```

**Explicação:** O menu e o botão serão escondidos se o dispositivo **NÃO** for uma tela (ou seja, sumirão na impressão).

# A Ordem Sequencial do CSS (Cascata)

**Como organizar o seu CSS** O CSS é lido de cima para baixo. Se você usa o modelo **Desktop-First** (começa pelo computador e vai diminuindo), as regras devem seguir a ordem decrescente:

- 1. CSS Geral:** Estilo base para Desktop (telas grandes).
- 2. Media Query Tablet:** (Ex: max-width: 1024px).
- 3. Media Query Mobile:** (Ex: max-width: 600px).

**Atenção:** Colocar a regra de 600px antes da de 1024px pode fazer com que a regra maior anule a menor. O **menor breakpoint sempre fica por último no arquivo.**

# Dicas de Organização Professional

- 1. Agrupamento:** Tente colocar todas as suas Media Queries no final do arquivo CSS.
- 2. Breakpoints Padrão:** Use medidas comuns de mercado:
  - **480px** (Celulares)
  - **768px** (Tablets)
  - **1024px** (Laptops/Tablets Grandes)
- 3. Somente Mudanças:** Não repita todo o código dentro da Media Query. Só coloque as propriedades que precisam ser alteradas (como o número de colunas do Grid).

# Links de Estudo e Documentação

Para aprofundar os estudos em Media Queries e Responsividade, utilize as fontes oficiais:

- **MDN Web Docs – Media Queries:** Guia completo e detalhado em português.

[https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS\\_media\\_queries/Using\\_media\\_queries](https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS_media_queries/Using_media_queries)

- **W3Schools – RWD Breakpoints:** Exemplos práticos dos pontos de quebra mais comuns.

[https://www.google.com/search?q=https://www.w3schools.com/css/css\\_rwd\\_breakpoints.asp](https://www.google.com/search?q=https://www.w3schools.com/css/css_rwd_breakpoints.asp)

- **Google Developers – Responsive Web Design:** Curso gratuito do Google sobre fundamentos de design moderno.

<https://web.dev/learn/design/>

# Resumo

- Design Responsivo** = Um site que se molda a qualquer tela.
- Viewport** = Tag Meta que habilita a responsividade no mobile.
- Media Types** = **all**, **print**, **screen**.
- Operadores** = **and** (e), **not** (não), **only** (apenas) e , (ou).
- Sequência** = No CSS, as telas menores vêm por último.

## ATIVIDADE

# Portal EcoCycle

1

## Ativação da Responsividade

**Contexto:** Portal EcoCycle (Notícias de Ecologia)

O portal EcoCycle precisa sair do layout de coluna única e se tornar um portal de notícias profissional. Para isso, utilizaremos **CSS Grid** para organizar os cards e **Media Queries** para que o site se adapte a tablets e celulares. Além disso, incluiremos conteúdos multimídia éticos.

Confira se a **Tag Viewport** se encontra no seu arquivo **index.html**. Caso não esteja inserida, inclua ela dentro da tag **<head>**

# Portal EcoCycle

2

## Estruturando com CSS Grid (Desktop)

Contexto: Portal EcoCycle (Notícias de Ecologia)

- 1. Multiplicação de Conteúdo:** No seu HTML, dentro da seção **.lista-noticias**, crie pelo menos **3 cards** de notícias para que possamos ver a grade funcionando.
- 2. Definição da Grade:** No CSS, transforme a classe **.lista-noticias** em um container **Grid** com 3 colunas iguais (1fr).
- 3. Ajuste de Card:** Remova o **width: 320px** fixo e o **margin: 20px auto** do **.card-noticia**, permitindo que o Grid controle o tamanho dos cards.

# Portal EcoCycle

3

## Multimídia e Ética Visual

Contexto: Portal EcoCycle (Notícias de Ecologia)

4. **Imagens:** Utilize imagens de bancos gratuitos (Pexels/Unsplash) e garanta que elas tenham o atributo **alt** preenchido.
5. **Vídeo Responsivo:** Adicione um novo card contendo um **<iframe>** do **YouTube** sobre sustentabilidade. Utilize uma **div** auxiliar para garantir que o vídeo seja fluido e não "estoure" a tela.

# Portal EcoCycle

4

**Operador AND (Ajuste para Tablets)**

**Contexto: Portal EcoCycle (Notícias de Ecologia)**

6. Crie uma Media Query para telas entre **768px** e **1024px**.
7. Neste intervalo, o Grid das notícias deve mudar de 3 colunas para **2 colunas**.

# Portal EcoCycle

5

**Operador VÍRGULA (Ajuste Mobile/Portrait)**

**Contexto: Portal EcoCycle (Notícias de Ecologia)**

8. Crie uma Media Query para telas menores que **600px** OU em modo **Portrait** (em pé).
9. Nesta condição:
  - a. O Grid deve exibir apenas **1 coluna**.
  - b. Os links do menu no cabeçalho devem ficar um abaixo do outro (**display: block**).
  - c. Esconda a descrição (**p**) dos cards para economizar espaço vertical.

## ATIVIDADE

# Portal EcoCycle

6

## Organização e Cascata

**Contexto:** Portal EcoCycle (Notícias de Ecologia)

10. Certifique-se de que as Media Queries estejam no **final do arquivo CSS**, seguindo a ordem: Desktop -> Tablet -> Mobile.

# Referências

SENAI-SP. **Desenvolvimento de Sistemas: Front-end.** São Paulo: Editora SENAI, 2023. MDN WEB DOCS.

**CSS Media Queries.** Disponível em:

[https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS\\_media\\_queries](https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS_media_queries).

W3C. **Media Queries Level 4.** Disponível em: <https://www.w3.org/TR/mediaqueries-4/>.

MARCOTTE, Ethan. **Responsive Web Design.** A Book Apart, 2011.



## ESCOLA SENAI ÍTALO BOLOGNA

Av. Goiás, 139

### Telefone

(11) 2396-1999

### Instagram

@senai.itu

### Facebook

/senaisp.itu

### Site

<https://sp.senai.br/unidade/itu/>