

INSTITUTO FEDERAL

Paraná

Campus Paranaguá

Media Queries

Desenvolvimento Web

Prof. Diego Stiehl

Web Design Responsivo

- Ato de projetar uma página web que seja capaz de “responder” às características do dispositivo ao qual é servida
 - Expandir
 - Contrair
 - Mudar de lugar
 - Redimensionar
 - ...

Web Design Responsivo

- Adaptar conteúdo com a finalidade de acomodá-lo de maneira usável e acessível ao local onde é visualizado
 - Smartphone
 - Tablet
 - Desktop
 - Leitor de tela
 - Search engine
 - ...

Como?

- Mas como vou atingir um design responsivo?
 - Código HTML5
 - Semântico
 - Bem estruturado
 - Código CSS3
 - Bem estruturado
 - Devidamente “cascadeado”
 - Utilizando Media Queries
 - Código JavaScript
 - Compatibilidade com browsers antigos

Ingredientes

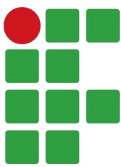
- O design responsivo (web) é composto pela combinação de três “ingredientes técnicos”
 - Grids fluidos
 - Imagens e mídias flexíveis
 - Media queries

Media Queries

- *Media Query*
 - Tradução: Consulta de Mídia
- Permite a especificação de regras CSS somente sob determinadas condições
 - Condições baseadas nas características do local onde a página é renderizada
- Funcionam como uma condição **if**

Marcação HTML

- Sugestão da W3C:
 - “Uma marcação HTML para todos os tipos de mídia”
- A ideia é não modificar o conteúdo (HTML) pensando em atingir um comportamento ideal em diferentes dispositivos:
 - Desktop, smartphone, TV, impressora, ...



CSS Básico

- Devemos resolver toda responsividade que for possível apenas com CSS básico
 - Sem Media Queries
- Aplicar as Media Queries a partir do momento que necessitar comportamentos diferentes em diferentes tipos de mídia

Tipos de Mídia

- A CSS3 aceita 4 palavras reservadas para diferentes tipos de mídia
 - screen
 - Para renderização em uma tela (monitor)
 - print
 - Para impressão
 - speech
 - Para sintetizadores de fala
 - all
 - Vale para todos os anteriores

Definindo uma Media Query

- Via CSS (direto no arquivo já utilizado)

```
@media condição {  
    /*  
        Bloco com regras CSS normais.  
        Serão aplicadas somente se a  
        condição for satisfeita.  
    */  
}
```

- Via HTML
 - Vai importar somente se a condição for satisfeita

```
<link rel="stylesheet" href="..." media="condição">
```

Condição

- Segundo a W3C
 - “A Media Query é uma sentença formada por um tipo de mídia e zero ou mais expressões destinadas a verificar as condições de ocorrência de uma determinada característica naquele tipo de mídia.”
- Se a condição for verdadeira:
 - Aplica todos estilos do bloco

Exemplos

- CSS para ser renderizado apenas em telas

```
@media screen {  
    /*  
        Bloco com regras CSS normais.  
    */  
}
```

```
<link rel="stylesheet" href="..." media="screen">
```

Aplicando Condição

- Para aplicar uma condição, usamos o operador and
 - Funciona como o && das linguagens de programação
- Toda a condição precisa ser satisfeita para que o bloco CSS seja aplicado
 - Senão o bloco é descartado



Exemplo

```
@media screen and (min-width: 400px) {  
  /*  
    Aplica regras somente se a "tela"  
    tiver NO MÍNIMO 400px de largura.  
    Tela = Largura do Viewport (browser).  
  */  
  body {  
    background-color: #f00;  
  }  
}  
@media print and (max-height: 20cm) {  
  /*  
    Somente se a "folha" tiver NO MÁXIMO 20cm de altura.  
  */  
  body {  
    background-color: #0f0;  
  }  
}
```

Encadeamento

- Podemos encadear os operadores
 - Quantos forem necessários
- Exemplo:

```
@media screen and (min-width: 400px) and (max-width: 799px) {  
  /*  
    Aplica regras somente se a tela  
    tiver NO MÍNIMO 400px e NO MÁXIMO 799px de largura  
  */  
  body {  
    background-color: #ff0;  
  }  
}
```

Atributos

- Atributos que podem ser verificados

width

overflow-block

pointer

height

overflow-inline

hover

aspect-ratio

color

any-pointer

orientation

color-gamut

any-hover

resolution

color-index

light-level

scan

display-mode

scripting

grid

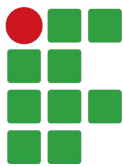
monochrome

update

inverted-colors

Documentação

- Alguns atributo aceitam...
 - Prefixos min- e max-
 - Apenas números inteiros
 - Apenas palavras-chave específicas
- Verificar documentação para mais detalhes:
 - <https://www.w3.org/TR/css3-mediaqueries>



E agora?

- Tá jóia! Saquei!
 - Mas como vou aplicar isso tudo no site que estou construindo?
- Mobile First ← **Forma mais indicada!**
 - Foque em criar o site pensando em “ficar bonito” em dispositivos móveis e adapte aos maiores
- Desktop First
 - Somente se a sua aplicação já existir e você quiser adaptá-la para dispositivos móveis

Variações

- Quantas variações responsivas do meu layout eu preciso oferecer?
 - Varia para cada projeto
 - Varia para dispositivos que deseja atingir
- Um padrão utilizado atualmente:
 - width < 576px ← Extra small
 - width >= 576px e < 768px ← Small
 - width >= 768px e < 992px ← Medium
 - width >= 992px e < 1200px ← Large
 - width >= 1200px ← Extra large

Breakpoints

- Estas variações são conhecidas como **breakpoints**
 - Pontos de quebra do layout
 - Se eu for expandindo minha tela, em que medida de largura o layout passará a se comportar de forma diferente?

Estilos em Cascata

- Tomar cuidado com a **precedência**

```
body {  
  background-color: #f00;  
}  
  
@media screen and (min-width: 992px) {  
  body {  
    background-color: #00f;  
  }  
}  
  
@media screen and (min-width: 768px) {  
  body {  
    background-color: #0f0;  
  }  
}
```

Uma tela com **1000px** de largura, por exemplo, irá aplicar:

ESTE
DEPOIS ESTE
E DEPOIS ESTE

Porque ela se encaixa em todas as condições.

Ordem correta

```
body {  
    background-color: #f00;  
}
```

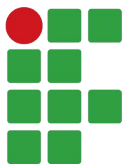
```
@media screen and (min-width: 768px) {  
    body {  
        background-color: #0f0;  
    }  
}
```

```
@media screen and (min-width: 992px) {  
    body {  
        background-color: #00f;  
    }  
}
```

Aplica a cor VERMELHA
para qualquer caso.

Caso a tela tenha mais
que 768px de largura,
troca para cor VERDE

Mas caso a tela tenha
mais que 992px de
largura, troca para cor
AZUL



Viewport

- Normalmente browsers móveis realizam um “zoom out” nos sites
 - Pode prejudicar a visibilidade e usabilidade
- Para melhor suporte a dispositivos móveis, utilizar a meta tag de viewport

```
<meta name="viewport"  
      content="width=device-width, initial-scale=1">
```

HTML Semântico

- **HTML semântico** é nada mais que escrever documentos HTML que sejam legíveis tanto para humanos quanto para computadores
- Motivo
 - Melhor apresentação e organização do código
 - Facilita a leitura
 - Melhora o ranqueamento nos motores de busca
 - SEO (Search Engine Optimization)

HTML5 Semântico

- A versão 5 do HTML criou algumas **tags com semântica**
- São DIVs com significado
- Devemos utilizá-las onde fizer sentido
 - DIVs ainda são válidas
- O Google vai gostar mais do nosso site

