

IOS – Instituto de  
Oportunidade Social

HTML, CSS e JS – Pseudoclasses,  
Pseudoelementos e Media Query



- Compreender o uso de pseudoclasses/pseudoelementos;
- Aplicar o conceito de pseudoclasses e pseudoelementos nas folhas de estilo.
- Compreender o uso de At-Rules;
- Compreender o uso de Media Query;
- Aplicar diferentes formatações conforme o tamanho do Viewport

IOS – Instituto de  
Oportunidade Social

## Pseudoclasses e Pseudoelementos



## Pseudoclasses

Por diversas vezes, você está em uma página da internet e encontra elementos da página, por exemplo links, que mudam de cor no momento que passamos o mouse sobre eles. Isso pode ser feito utilizando pseudoclasses no CSS. Pseudoclassee é uma palavra-chave usada para aplicar um efeito especial a um seletor. A pseudoclassee seleciona elementos para aplicar uma determinada ação, propriedade etc. A sintaxe de pseudoclassee é:

```
seletor:pseudoclassee {  
  property: value;  
}
```

**OBS:** Na apostila há uma lista, a seguir vamos ver um exemplo:

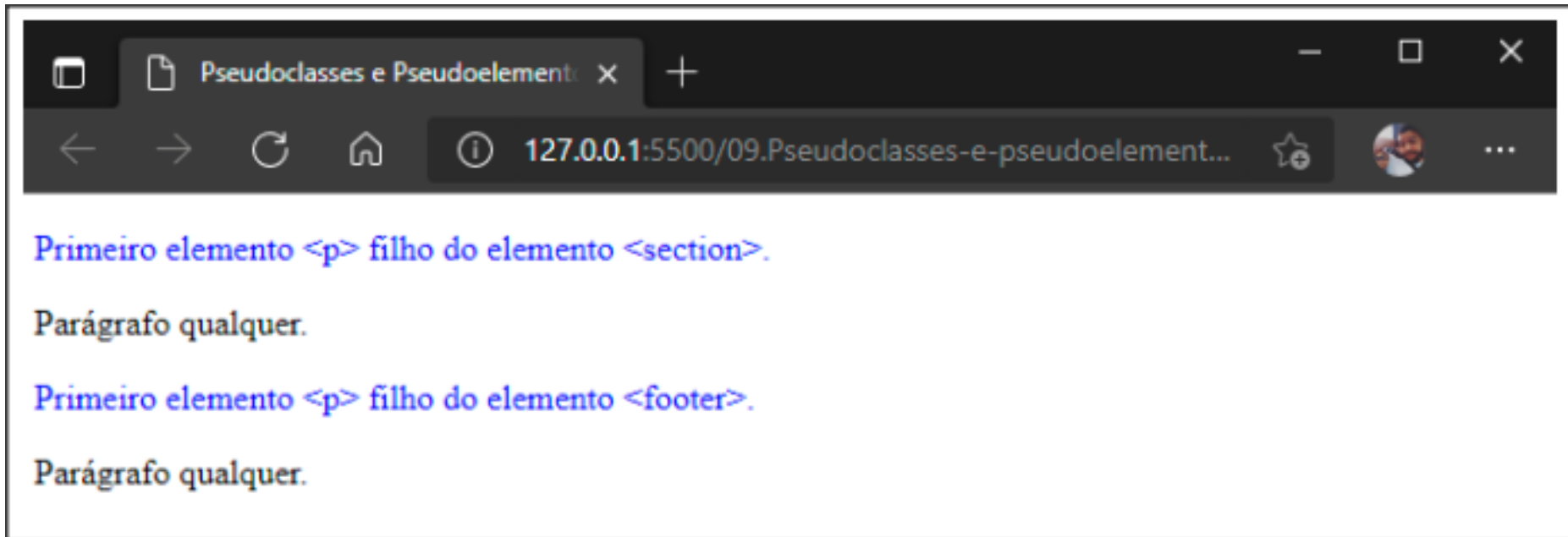
# Pseudoclasses e Pseudoelementos



```
<head>
  <meta charset="UTF-8">
  <title>Pseudoclasses</title>
  <style>
    p:first-child { color: blue; }
  </style>
</head>
<body>
  <section>
    <p>Primeiro elemento &lt;p> filho do elemento &lt;section>;</p>
    <p>Parágrafo qualquer.</p>
  </section>
  <footer>
    <p>Primeiro elemento &lt;p> filho do elemento &lt;footer>;</p>
    <p>Parágrafo qualquer.</p>
  </footer>
</body>
```

# Pseudoclasses e Pseudoelementos

Exemplo de página:



## Pseudoelementos

Pseudoelemento é uma palavra-chave usada para estilizar uma parte específica de um determinado elemento. Sua sintaxe é:

```
seletor::pseudoelemento {  
    property: value;  
}
```

Os principais pseudoelementos do CSS são:

| Seletor               | Exemplo         | Descrição do Exemplo  |
|-----------------------|-----------------|---|
| <b>::after</b>        | p::after        | Inserir um conteúdo depois de todo elemento <p>                     |
| <b>::before</b>       | p::before       | Inserir um conteúdo antes de todo elemento <p>                      |
| <b>::first-letter</b> | p::first-letter | Seleciona a primeira letra de todo elemento <p>                     |
| <b>::first-line</b>   | p::first-line   | Seleciona a primeira linha de todo elemento <p>                     |
| <b>::selection</b>    | p::selection    | Seleciona uma porção de um elemento que é selecionada pelo usuário. |

# Pseudoclasses e Pseudoelementos

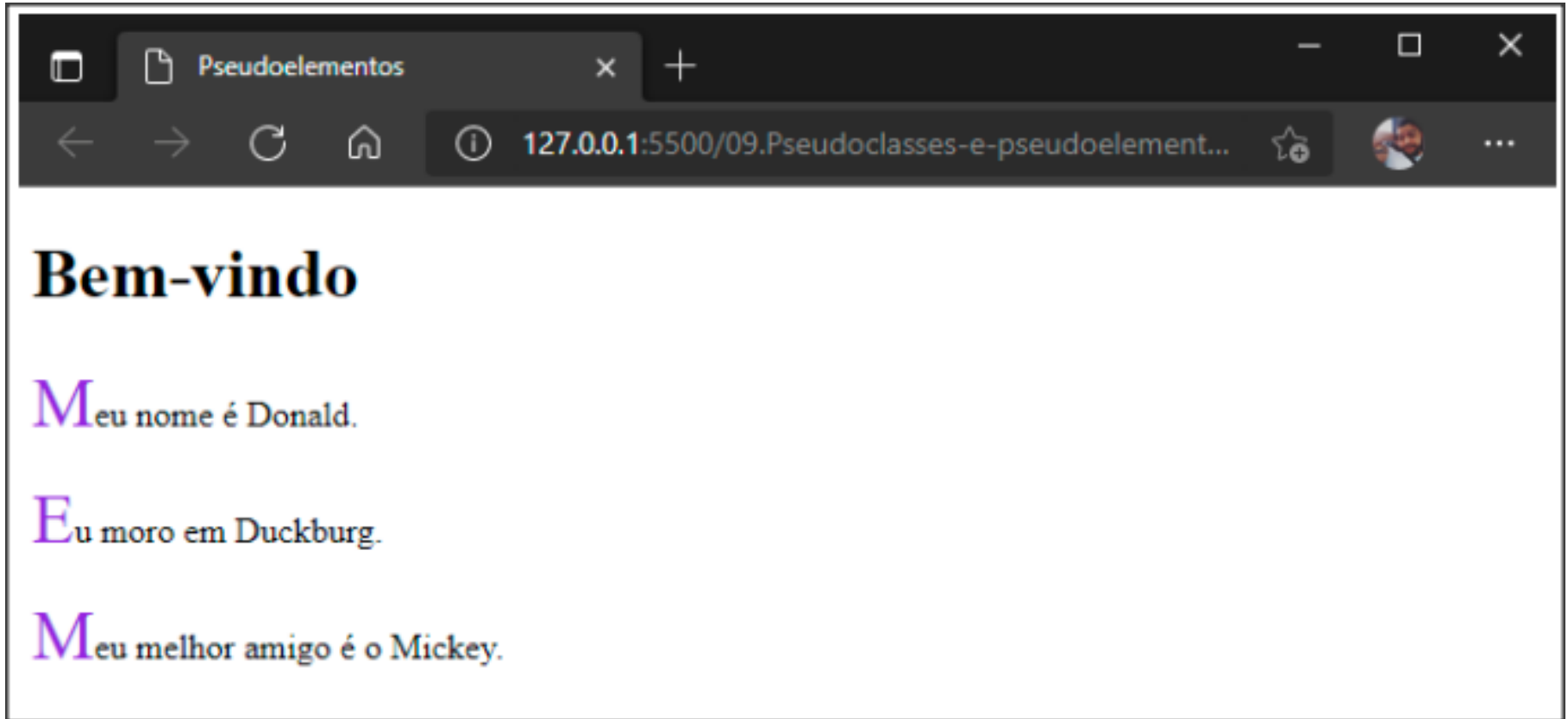


```
<head>
  <meta charset="UTF-8">
  <title>Pseudoelementos</title>
  <style>
    p::first-letter {
      font-size: 200%;
      color: #8A2BE2;
    }
  </style>
</head>
<body>
  <h1>Bem-vindo</h1>
  <p>Meu nome é Donald.</p>
  <p>Eu moro em Duckburg.</p>
  <p>Meu melhor amigo é o Mickey.</p>
</body>
```



# Pseudoclasses e Pseudoelementos

Exemplo de página:



IOS – Instituto de  
Oportunidade Social

Media Query



## At-rules

O CSS tem diversas regras, que são denominadas **At-Rules**, que são declarações com instruções de **como o CSS irá se comportar**. At-rules são iniciadas com o sinal de arroba (**@**), seguido por um identificador e terminando com um ponto-e-vírgula (**;**). Existem muitas at-rules regulares, que foram implementadas com seu identificador, cada uma com uma sintaxe diferente, por exemplo:

**@charset**: define o conjunto de caracteres (**codificação**) usado pelo CSS.

**@import**: informa ao CSS para incluir uma folha de **estilos externa**.

**@namespace** (en-US): informa ao CSS que todo o conteúdo deve ser considerado **prefixado** com um namespace XML.

Existe também um subconjunto de declarações, que pode ser usado como uma regra de uma folha de estilos, dentro de um grupo de regras condicional:

**@media:** um grupo de regras condicional que é aplicado ao conteúdo caso o **dispositivo** de exibição atenda com a condição de mídia definida (media query).

**@document** (en-US): um grupo de regras condicional que é aplicado ao conteúdo caso o **documento** atenda aos critérios da condição definidos.

**@page:** descreve o aspecto das mudanças do layout que serão aplicadas ao **imprimir** o documento

**@font-face:** descreve o aspecto de uma **fonte externa** que deve ser baixada.

**@supports** (en-US): um grupo de regras condicional que é aplicado ao conteúdo caso o **navegador** atenda aos critérios da condição definidos.

**@keyframes:** descreve o **aspecto das etapas intermediárias** em uma sequência de animação CSS. Ele é muito usado para declarar variáveis em CSS.

**@viewport:** descreve os aspectos da **área de visualização** para dispositivos com tela pequena.

**@counter-style:** define estilos de **contador específicos** que não fazem parte do conjunto predefinido de estilos.

**@property:** descreve o aspecto de **propriedades** e variáveis **personalizadas**.

**@color-profile:** permite que um **perfil de cores** seja definido para uso da função color().

## Media query

Atualmente, os sites devem se adaptar a diferentes mídias (computador, tablet, celulares, etc.) e é importante que a informação exibida no site seja enquadrada corretamente nas diferentes telas de exibição. Isso é o que chamamos de **web design responsivo**. Por exemplo, algumas larguras de tela comuns são:

Smartphones: 320px

Tablets: 768px

Netbook e modo de visualização landscape(paisagem) do tablet: 1024px

Desktop com tela mais larga: 1600px

Portanto **media query** é uma definição do tipo de mídia (como telas) e de uma expressão lógica que determina a resolução e orientação (Portrait [retrato] ou landscape [paisagem]) da tela de um dispositivo para exibição de um conteúdo. Quando a media query é verdadeira, direciona para a parte do código CSS.

Você pode fazer regras de acordo com o tipo de mídia e/ou tipo de características. Os tipos de mídia aceitos são:

| Valor         | Descrição   |
|---------------|---|
| <b>all</b>    | Default. Usada para todos os dispositivos de tipo de mídia                              |
| <b>print</b>  | Usada para impressora   |
| <b>screen</b> | Usada para telas de computadores, tablets, smartphones, etc.                            |
| <b>speech</b> | Usada para telas de leitura ( por exemplo: kindle), que “leem” a página para o usuário. |

**OBS:** Na página 118 da apostila há uma listagem das características da mídia que podem ser utilizadas.

## @media

A regra **@media** é usada para aplicar estilos com base no resultado de uma ou mais consultas de mídia e é capaz de testar características específicas de um dispositivo, ou seja, é usada em media query para aplicar diferentes estilos para diferentes tipos de mídias e dispositivos.

Ela pode ser usada para testar muitas coisas, tais como:

Largura e altura da **área de visualização**.

Largura e altura de um **dispositivo**.

**Orientação**, por exemplo, em tablets e celulares, temos retrato (portrait) ou paisagem (landscape).

**Resolução**.



Div price-box:

```
<div class="col">  
  <ul class="price-box">  
    <li class="header">Simples</li>  
    <li class="emph"><strong>R$ 5.99</strong> / ao Mês</li>  
    <li><strong>20GB</strong> Espaço em Disco</li>  
    <li><strong>10GB</strong> Transferência de dados</li>  
    <li><strong>2</strong> Domínios</li>  
    <li><strong>50</strong> Contas de Email</li>  
    <li><strong>2</strong> Contas de FTP</li>  
    <li><a href="#" class="btn">Cadastre-se</a></li>  
  </ul>  
</div>
```

Div price-box best:

```
<div class="col">  
  <ul class="price-box best">  
    <li class="header header-green">Padrão</li>  
    <li class="emph"><strong>R$ 15.99</strong> / ao Mês</li>  
    <li><strong>75GB</strong> Espaço em Disco</li>  
    <li><strong>50GB</strong> Transferência de dados</li>  
    <li><strong>10</strong> Domínios</li>  
    <li><strong>100</strong> Contas de Email</li>  
    <li><strong>Ilimitadas</strong> Contas de FTP</li>  
    <li><a href="#" class="btn">Cadastre-se</a></li>  
  </ul>  
</div>
```

Página Web antes de aplicar o CSS:



Aplicando alguns estilos a página:

```
@charset "utf-8";  
body {  
    font-family: "Tahoma", sans-serif;  
    background: #333;  
    color: #fff;  
}  
h2 {  
    text-align: center;  
}  
.col {  
    float: left;  
    width: 30%;  
    padding: 8px;  
    box-sizing: border-box;  
}
```


Página Web após aplicar alguns estilos:



## Ajustando o price-box:

```
.price-box {  
    background: #fff;  
    color: #555;  
    list-style-type: none;  
    border: 1px solid #555;  
    margin: 0;  
    padding: 0;  
}  
.price-box .header {  
    background-color: #111;  
    color: #fff;  
    font-size: 25px;  
}  
.price-box .header.header-green {  
    background: #4caf50;  
}
```

Página Web após aplicar os ajuste no price-box:



The screenshot shows a web browser window with the title "CSS3 site responsivo de tabela" and the address bar displaying "127.0.0.1:5500/10.Media-query/código-fonte-mate...". The main content area features a heading "CSS3 site responsivo de tabela de preços" and three pricing plans: "Simples", "Padrão", and "Super". The "Padrão" plan is highlighted with a green header. Each plan lists its price per month and various features like disk space, data transfer, domains, email accounts, and FTP accounts. A "Cadastre-se" link is provided at the bottom of each plan.

| Simples                     | Padrão                      | Super                         |
|-----------------------------|-----------------------------|-------------------------------|
| R\$ 5.99 / ao Mês           | R\$ 15.99 / ao Mês          | R\$ 25.99 / ao Mês            |
| 20GB Espaço em Disco        | 75GB Espaço em Disco        | 120GB Espaço em Disco         |
| 10GB Transferência de dados | 50GB Transferência de dados | 1000GB Transferência de dados |
| 2 Domínios                  | 10 Domínios                 | Ilimitados Domínios           |
| 50 Contas de Email          | 100 Contas de Email         | Ilimitadas Contas de mail     |
| 2 Contas de FTP             | Ilimitadas Contas de FTP    | Ilimitadas Contas de FTP      |
| <a href="#">Cadastre-se</a> | <a href="#">Cadastre-se</a> | <a href="#">Cadastre-se</a>   |

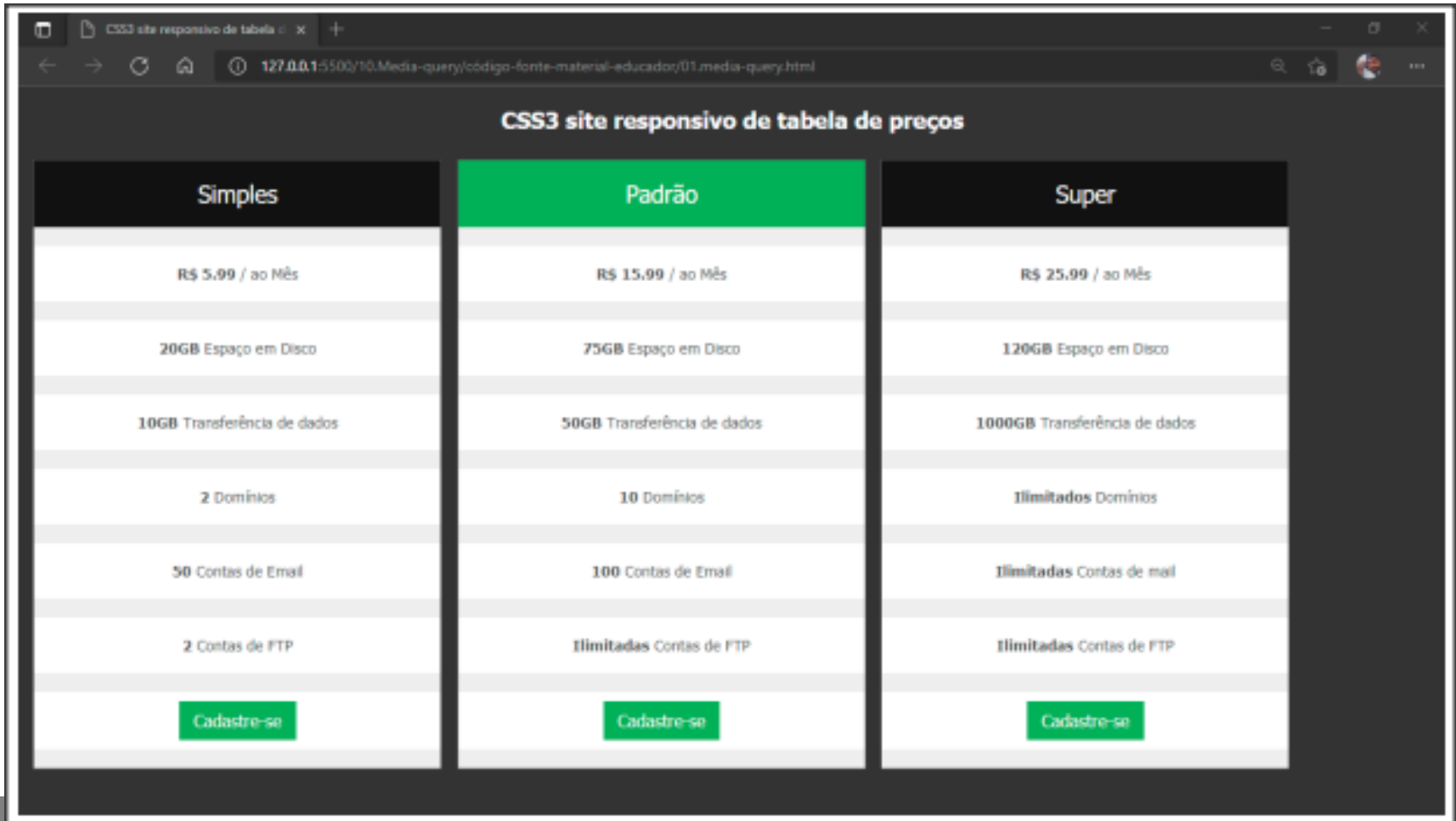
## Aplicando ajustes finais:

```
.price-box li {  
    padding: 20px;  
    border-bottom: 20px solid #eee;  
    text-align: center;  
}  
.btn {  
    background-color: #4caf50;  
    border: none;  
    color: #fff;  
    padding: 10px 15px;  
    text-align: center;  
    text-decoration: none;  
    font-size: 18px;  
}
```



# Media Query

Página Web após aplicar os ajuste finais:



| Simples                     | Padrão                      | Super                         |
|-----------------------------|-----------------------------|-------------------------------|
| R\$ 5.99 / ao Mês           | R\$ 15.99 / ao Mês          | R\$ 25.99 / ao Mês            |
| 20GB Espaço em Disco        | 75GB Espaço em Disco        | 120GB Espaço em Disco         |
| 10GB Transferência de dados | 50GB Transferência de dados | 1000GB Transferência de dados |
| 2 Domínios                  | 10 Domínios                 | Ilimitados Domínios           |
| 50 Contas de Email          | 100 Contas de Email         | Ilimitadas Contas de mail     |
| 2 Contas de FTP             | Ilimitadas Contas de FTP    | Ilimitadas Contas de FTP      |
| <a href="#">Cadastre-se</a> | <a href="#">Cadastre-se</a> | <a href="#">Cadastre-se</a>   |

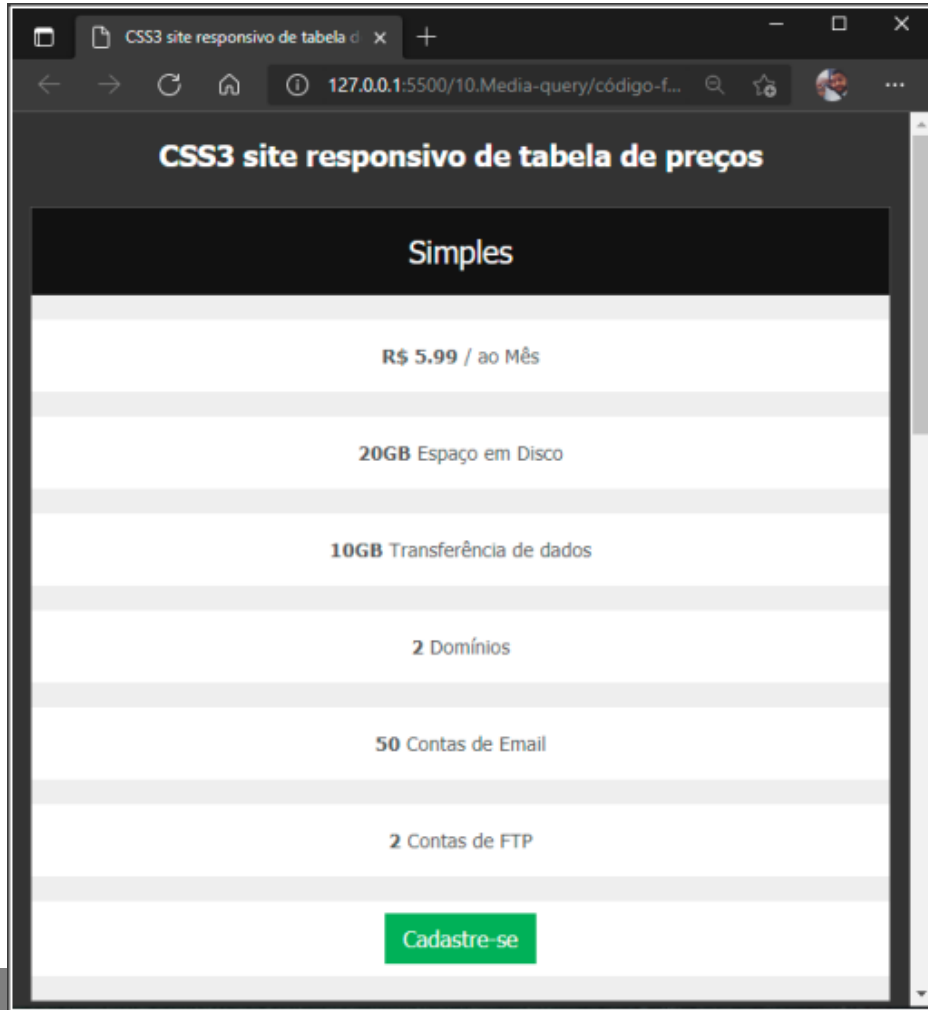
Tornando a página ajustável:

Enfim, vamos configurar a **media query**, para exibir as tabelas uma embaixo da outra, quando estivermos exibindo o site **em tela de até 700px**. Insira o código no arquivo CSS. Nesse caso vamos configurar a classe **.col** para usar a largura da área de visualização em 100%, para tela com até **700px de largura**.

```
@media(max-width:768px) {  
    .col {  
        width: 100%;  
    }  
}
```

# Media Query

Redimensionar o navegador para uma largura inferior a 700px:



IOS – Instituto de  
Oportunidade Social

## Exercícios



Montar uma página atendendo os seguinte critérios:

- Utilizar pelo menos 5 **pseudoclasses** (listagem pag 100)
- Utilizar pelo menos 3 **pseudoelementos**
- Utilizar alinhamento de listagens/elementos com **float** definindo uma porcentagem da largura (**width**). Ex: 30%,50%, etc
- Utilizar o Media Query **@media** para definir comportamento diferente por tamanho de tela, dessa forma a visualização deverá mudar se diminuir o espaço de visualização do navegador
- Publicar no GitHub e enviar no Moodle