

HTML e CSS

BCC481 Programação Web

Vander Luis de Souza Freitas

- **Hyper Text Markup Language (HTML);**
- Descreve a estrutura de uma página Web por uma série de elementos, descritos por tags, como “heading”, “paragraph”, etc.
- Os Browsers utilizam tais tags para renderizar o conteúdo da página.

Exemplo de estrutura de uma página:

```
<!DOCTYPE html>
<html>
  <head>
    <title>BCC481</title>
  </head>
  <body>
    ... conteúdo da página ...
  </body>
</html>
```

Exemplo de estrutura de uma página:

```
<!DOCTYPE html>
<html>
  <head>
    <title>BCC481</title>
  </head>
  <body>
    ... conteúdo da página ...
  </body>
</html>
```

Metadados que não aparecem na viewport do browser

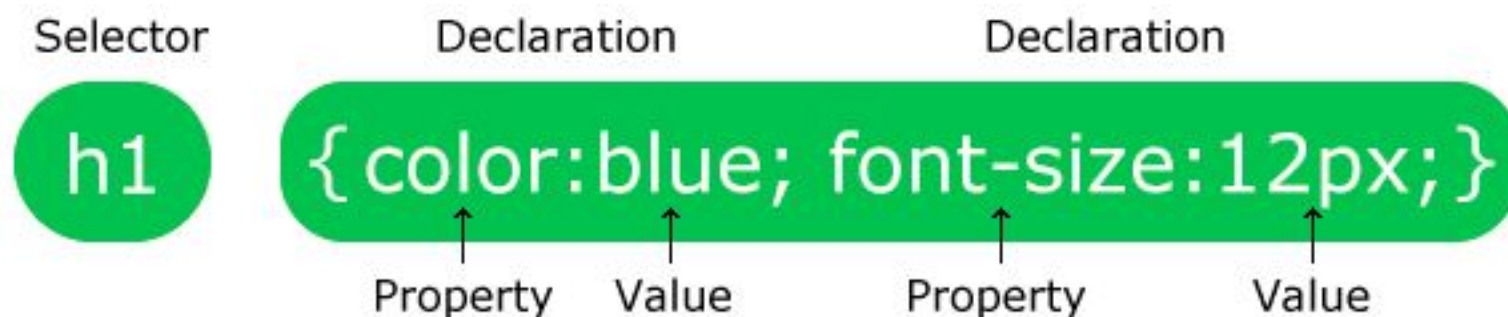
Aparece como nome da página, na aba do browser

Conteúdo que é renderizado na viewport do browser

CSS (Cascading Style Sheets) é uma linguagem que descreve o estilo de um documento HTML.

CSS descreve como os elementos devem ser apresentados.

Uma declaração consiste de um seletor e um bloco de declaração:



```
selector {  
    property: value;  
}
```

```
selector {  
    property: value;  
}
```

selector: Especifica os elementos HTML a estilizar

property: O nome do estilo CSS.

value: O valor para o estilo CSS.

Salvo em um arquivo do tipo: *arquivo.css*

Links úteis:

Mozilla Developer Network (MDN)

<https://developer.mozilla.org/en-US/docs/Web/CSS>

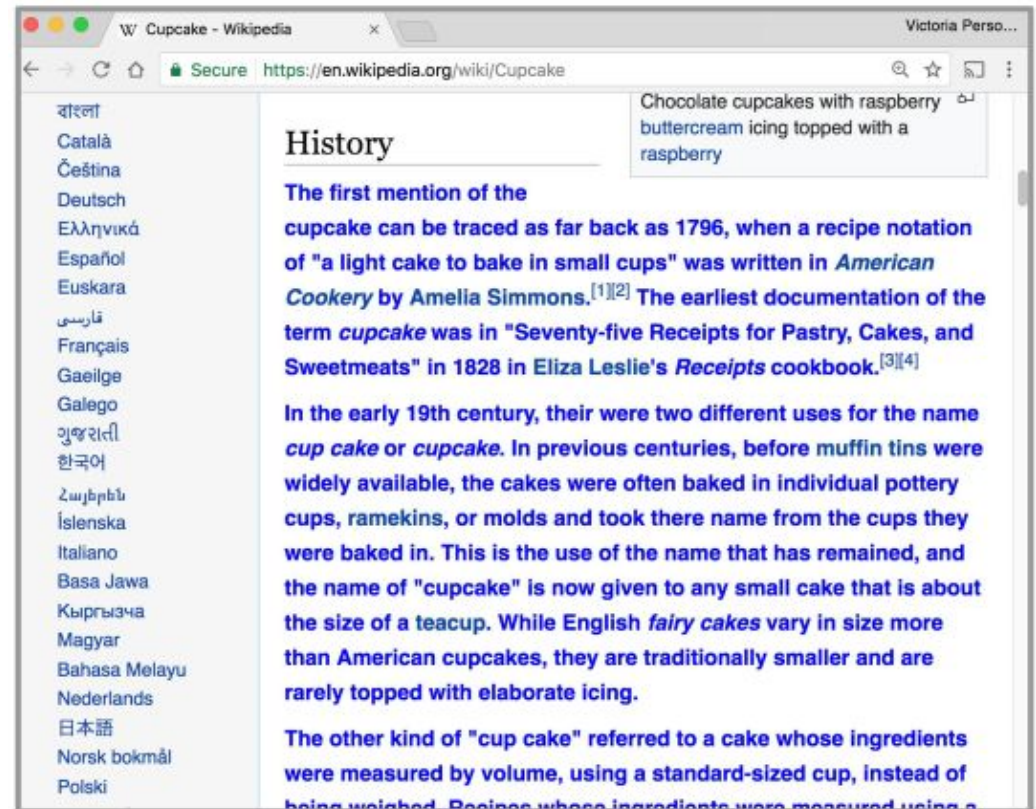
W3 Schools

<https://www.w3schools.com/css/default.asp>

Exemplo:

```
p {
  color: blue;
  font-weight: bold;
}
```

Todos os elementos <p> da página devem ser azuis e em negrito



Exemplo:

```
p {  
  color: red;  
  /* This is a single-line comment */  
  text-align: center;  
}
```

```
/* This is  
a multi-line  
comment */
```

Inserindo CSS no código HTML:

```
<!DOCTYPE html>
<html>
  <head>
    <title>BCC481</title>
    <link rel="stylesheet" href="filename.css" />
  </head>
  <body>
    ... conteúdo da página...
  </body>
</html>
```

Propriedades:

Existem mais de [500 propriedades CSS](#)! Algumas abaixo:

Font face (MDN)	<code>font-family: Helvetica;</code>
Font color (MDN)	<code>color: gray;</code>
Background color (MDN)	<code>background-color: red;</code>
Border (MDN)	<code>border: 3px solid green;</code>
Text alignment (MDN)	<code>text-align: center;</code>

notação
shorthand

Formas de definir cores em CSS:

140 nomes pré-definidos ([lista](#))

`color: black;`

[rgb\(\)](#) e [rgba\(\)](#)

`color: rgb(34, 12, 64);`

`color: rgba(0, 0, 0, 0.5);`

Valores [Hex](#):

`color: #00ff00;`

`color: #0f0;`

`color: #00ff0080;`

O “a” corresponde ao canal **alpha**, que representa um valor de **transparência**.

Prefira sempre formas mais descritivas:

- Nomes pré-definidos
- rgb / rgba
- Hex

Relembrando problema da aula anterior:



```
<!DOCTYPE html>
<html>
  <head>
    <title>BCC481</title>
  </head>
  <body>
    <h1>BCC481: Programação Web</h1>
    <strong>News:</strong><br/>
    Horários de atendimento disponibilizados<br/>
    Bora estudar!<br/>
    <br/>
    <a href="http://ufop.br"> UFOP </a>
  </body>
</html>
```

Estranhamente, o texto dentro de `<h1>` ficou em uma linha reservada, enquanto o texto dentro de `` não.

block vs inline

Tipos de elementos HTML

Cada elemento é categorizado pela especificação HTML dentre uma das três categorias a seguir:

- **block**: grandes blocos de conteúdo, que possuem altura (height) e largura (width)
`<div>`, `<p>`, `<h1>`, ``, ``, `<table>`
- **inline**: pequenas quantidades de conteúdo, sem altura or largura
`<a>`, ``, ``, ``
 - **inline block**: conteúdo inline com altura e largura
``
- **metadata**: informação sobre a página, geralmente invisível
`<title>`, `<meta>`

Elementos block

Exemplos:

`<div>`, `<p>`, `<h1>`, ``, ``, `<table>`

- Tomam toda a largura da página (**o fluxo é de cima pra baixo**)
- Tem altura e largura
- Podem ter elementos block ou inline como filhos



Elementos block

Exemplo block:



```
<!DOCTYPE html>
<html>
<head>
  <title>Block example</title>
</head>
<body>
  <h1>Sobre computadores</h1>
  <p>
    Gostam de <em>programar</em>
  </p>
</body>
</html>
```

Elementos block

Exemplo block:



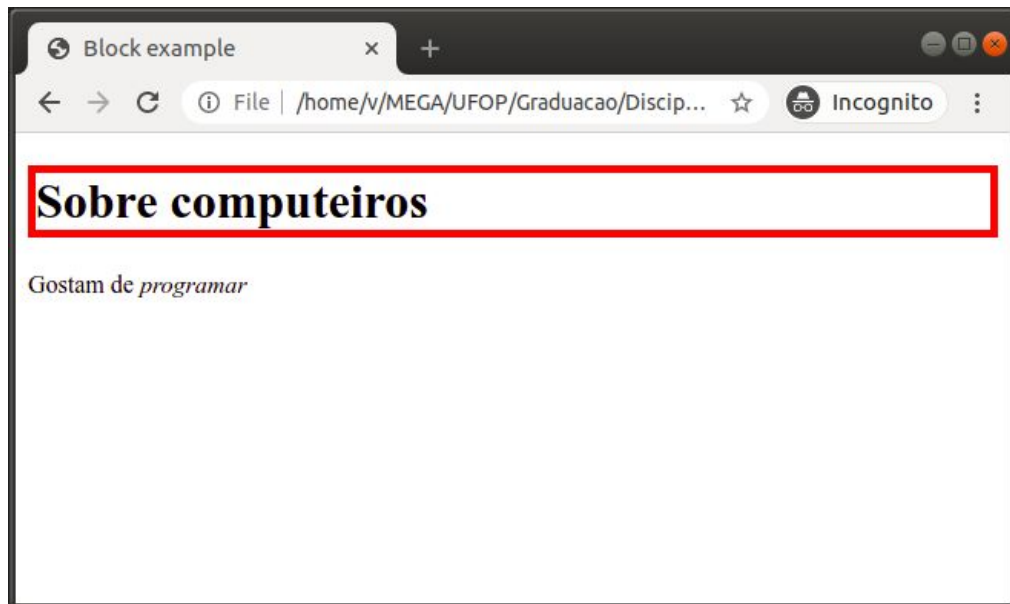
```
<!DOCTYPE html>
<html>
<head>
  <title>Block example</title>
</head>
<body>
  <h1>Sobre computadores</h1>
  <p>
    Gostam de <em>programar</em>
  </p>
</body>
</html>
```

Qual o resultado do código CSS abaixo nessa página?

```
h1 {
  border: 5px solid red;
}
```

Elementos block

Exemplo block:



`<h1>` é um elemento block, portanto se estende por toda a largura da página.

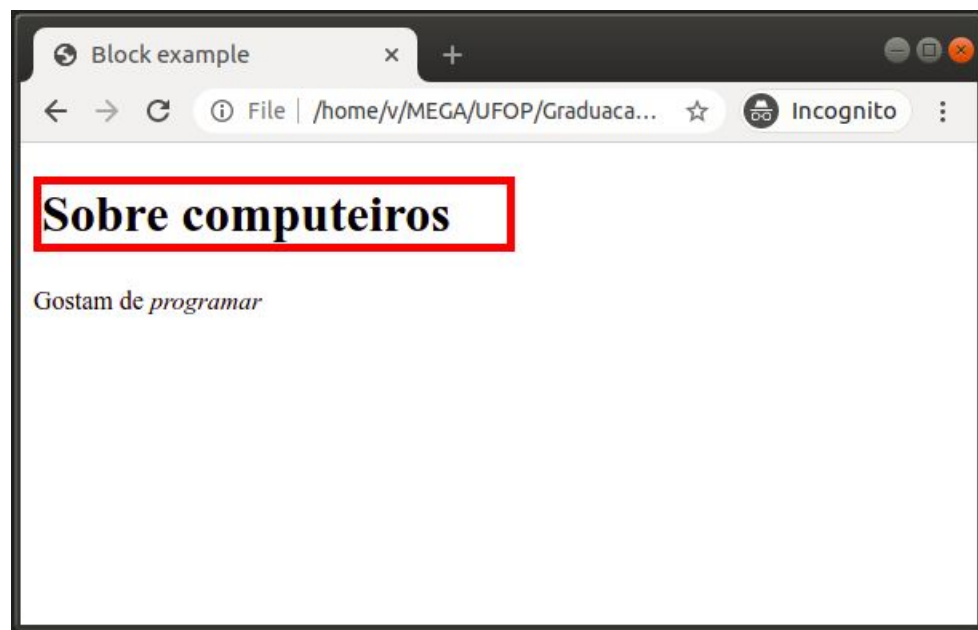
*Note que os elementos block `<p>` e `<h1>` fluem de cima pra baixo.

```
<!DOCTYPE html>
<html>
<head>
  <title>Block example</title>
  <style>
    h1 {
      border: 5px solid red;
    }
  </style>
</head>
<body>
  <h1>Sobre computadores</h1>
  <p>
    Gostam de <em>programar</em>
  </p>
</body>
</html>
```

Elementos block

Exemplo block:

E se adicionarmos uma largura para o elemento `<h1>`?



`<h1>` é do tipo block, portanto a largura pode ser modificada.

```
<!DOCTYPE html>
<html>
<head>
  <title>Block example</title>
  <style>
    h1 {
      border: 5px solid red;
      width: 50%;
    }
  </style>
</head>
<body>
  <h1>Sobre computadores</h1>
  <p>
    Gostam de <em>programar</em>
  </p>
</body>
</html>
```

Elementos inline

Exemplos:

`<a>`, ``, ``, ``

- Tomam apenas a largura necessária (da esquerda para a direita)
- **Não podem** ter largura e altura
- **Não podem** ter um elemento block como filho
- **Não podem** ser posicionados (por exemplo propriedades como float e position do CSS não se aplicam)
 - Ao invés, **deve-se posicionar seu elemento block** pai



Elementos inline

Exemplo inline:



```
<!DOCTYPE html>
<html>
<head>
  <title>Inline example</title>
</head>
<body>
  <strong>Recursos de Programação Web:</strong>
  <a href="https://developer.mozilla.org/en-US/">MDN</a>
  <a href="http://google.com">Google</a>
</body>
</html>
```

Elementos inline

Exemplo inline:

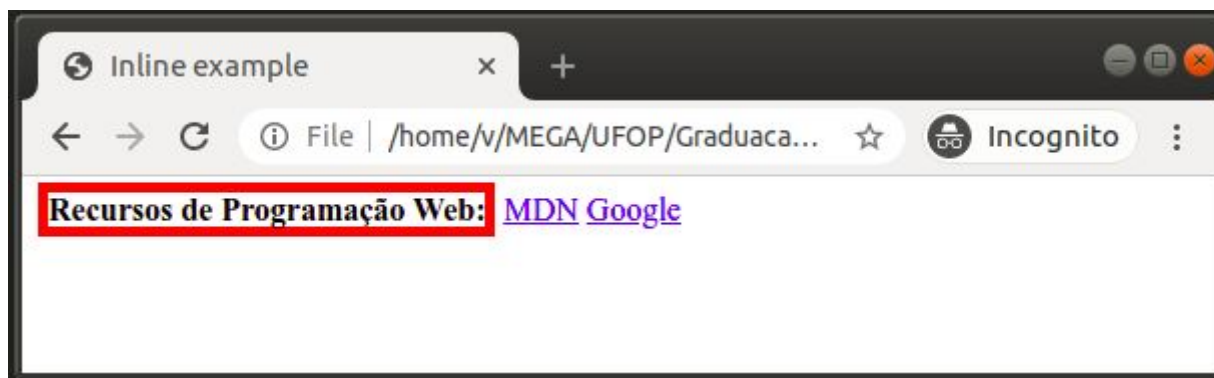


O que acontece se o elemento `` receber uma largura?

```
strong {  
    border: 5px solid red;  
    width: 1000px;  
}
```

Elementos inline

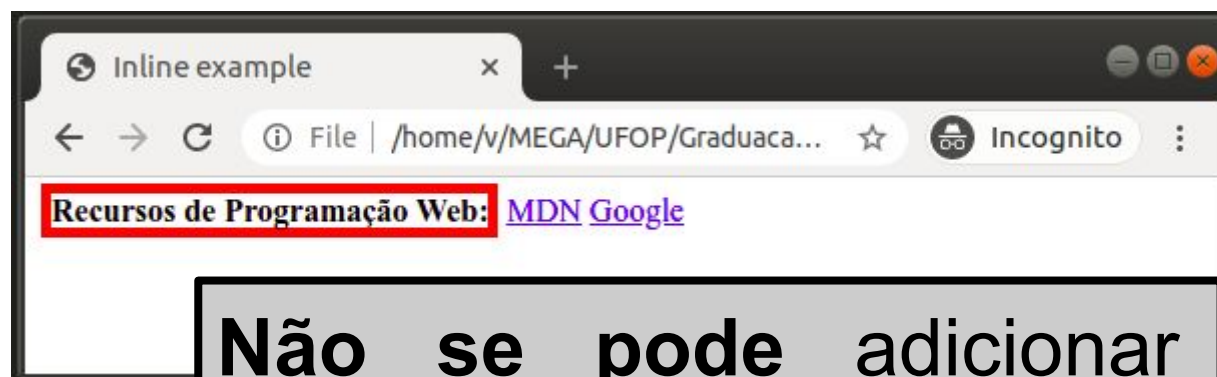
Exemplo inline:



```
<!DOCTYPE html>
<html>
<head>
  <title>Inline example</title>
  <style type="text/css">
    strong {
      border: 5px solid red;
      width: 1000px; /* não funciona; strong é inline! */
    }
  </style>
</head>
<body>
  <strong>Recursos de Programação Web:</strong>
  <a href="https://developer.mozilla.org/en-US/">MDN</a>
  <a href="http://google.com">Google</a>
</body>
</html>
```


Elementos inline

Exemplo inline:



Não se pode adicionar largura a elementos inline, portanto o comando width é ignorado.

```
<!DOCTYPE html>
<html>
<head>
  <title>Inline
  <style type="text/css">
    strong {
      border: 5px solid red;
      width: 1000px; /* não funciona; strong é inline! */
    }
  </style>
</head>
<body>
  <strong>Recursos de Programação Web:</strong>
  <a href="https://developer.mozilla.org/en-US/">MDN</a>
  <a href="http://google.com">Google</a>
</body>
</html>
```

Elementos inline-block

Exemplos:

``, ou qualquer elemento com `display: inline-block;`

- Tomam apenas a largura necessária (da esquerda para a direita)
- **Podem** ter largura e altura
- **Podem** ter um elemento block como filho
- **Podem** ser posicionados (por exemplo propriedades como float e position do CSS se aplicam)



Elementos inline-block

Exemplo inline:



```
<!DOCTYPE html>
<html>
<head>
  <title>Inline-block example</title>
  <style type="text/css">
    img {
      width: 50px;
    }
  </style>
</head>
<body>
  
  
  
  
  
</body>
</html>
```

Elementos inline-block

Exemplo inline:

- Elementos **inline-block** possuem altura, largura e fluem da esquerda para a direita, por isso que as imagens aparecem uma ao lado da outra.
- Note a largura da figura igual a 50px.



```
<!DOCTYPE html>
<html>
<head>
  <title>Inline-block example</title>
  <style type="text/css">
    img {
      width: 50px;
    }
  </style>
</head>
<body>
  
  
  
  
  
</body>
</html>
```

A propriedade display do CSS

O tipo de renderização padrão de um elemento pode ser alterado pela propriedade **display**. Exemplos:

```
p {  
    display: inline;  
}
```

```
a {  
    display: block;  
}
```

Valores possíveis para display:

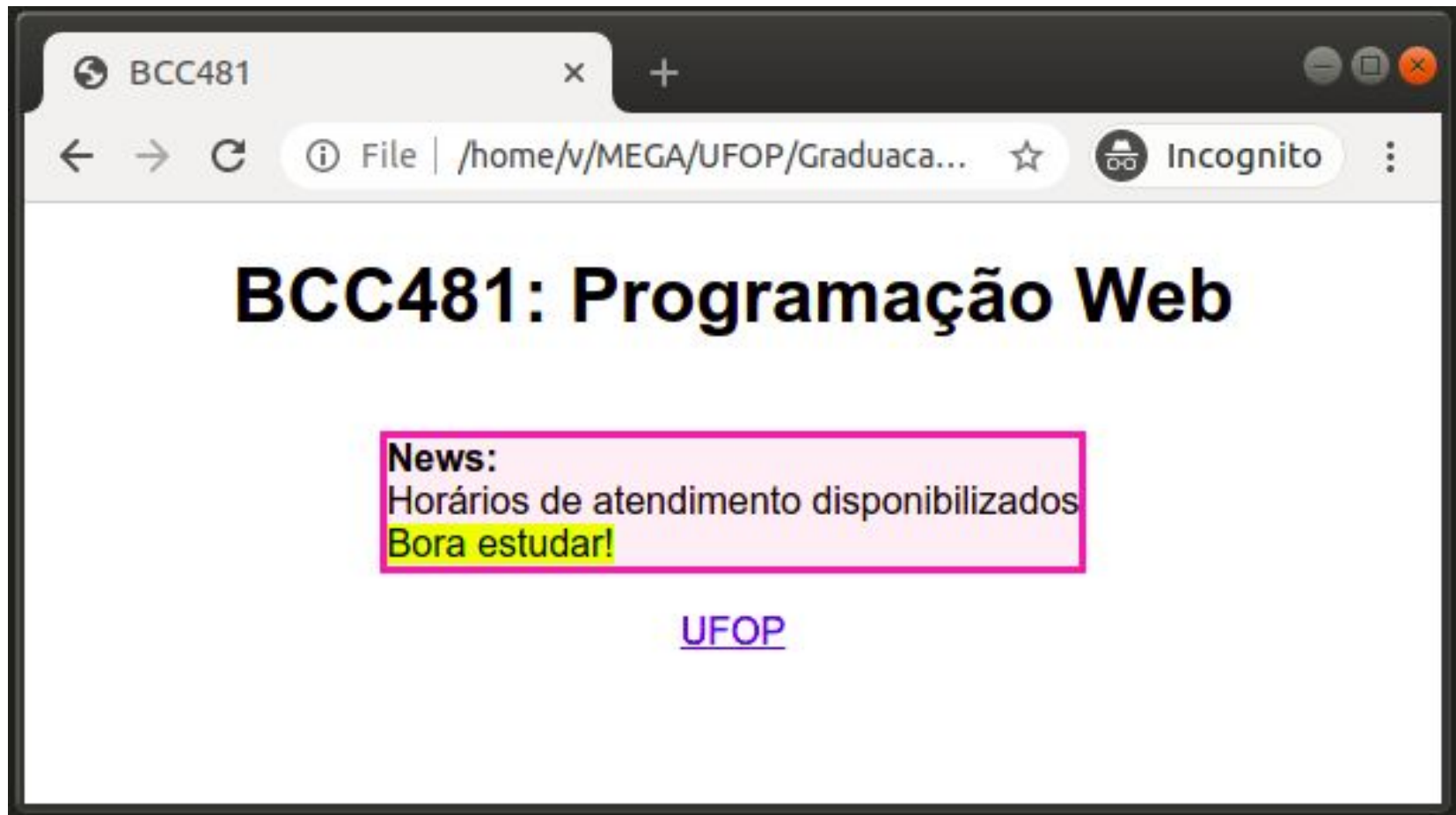
- block
- inline
- inline-block
- outros: [link](#)

- block: fluxo de cima para baixo (top-to-bottom); possui altura e largura
`<div>`, `<p>`, `<h1>`, ``, ``, `<table>`
- inline: fluxo da esquerda para a direita (left-to-right); não possui altura e largura
`<a>`, ``, ``, ``
- inline-block: fluxo da esquerda para a direita; possui altura e largura
``

Se teu CSS não estiver funcionando, verifique se você está aplicando propriedades de nível block em elementos inline.

Exemplo

- Crie a página a seguir, utilizando HTML e CSS:



Exemplo

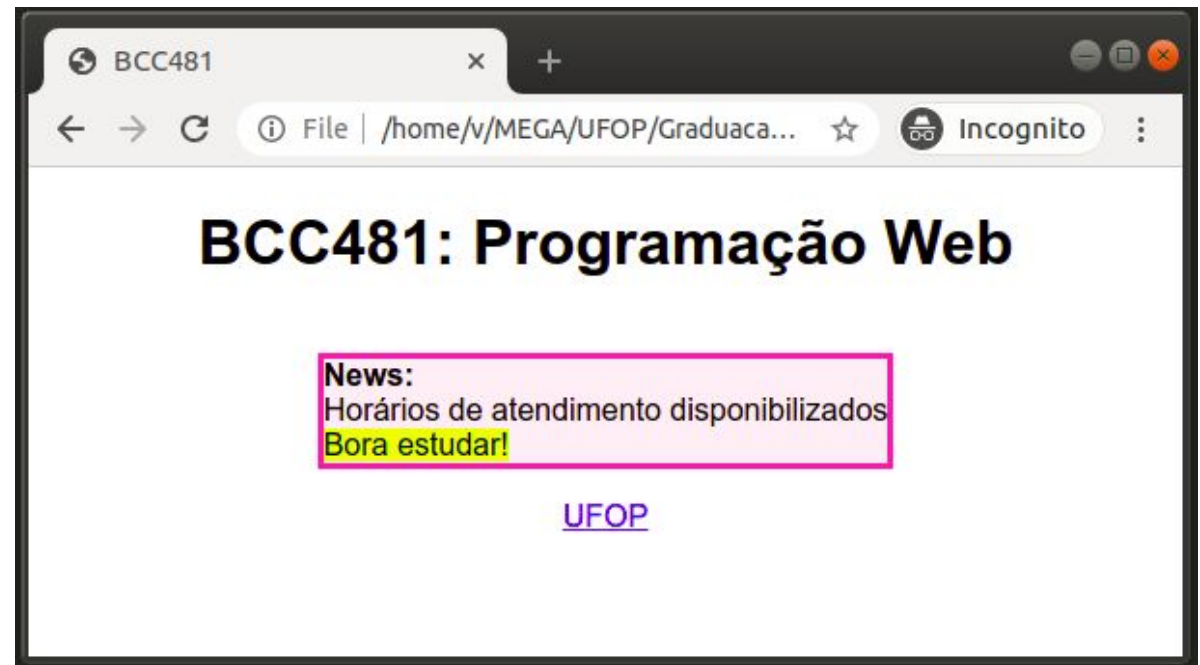
- Crie a página a seguir, utilizando HTML e CSS:

Font face: Helvetica

Border: hotpink 3px

Background color:
lavenderblush

- Box is **centrado**
- Header e link são **centrados**
- Conteúdo do box está **alinhado à esquerda**



Exemplo

Dado o código HTML abaixo, como seria o arquivo css apropriado?

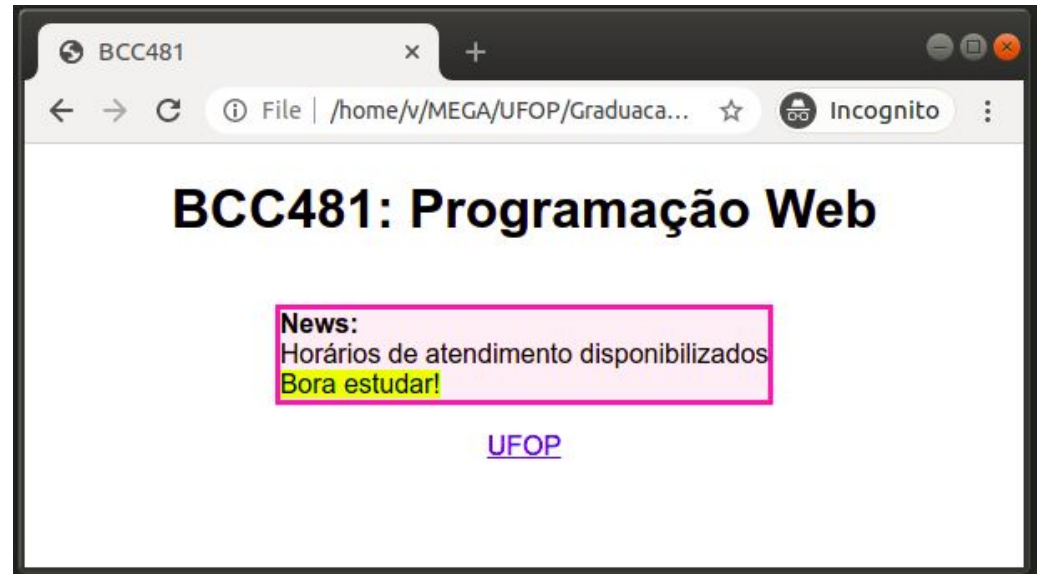
```
<!DOCTYPE html>
<html>
  <head>
    <title>BCC481</title>
    <link rel="stylesheet" href="css/css_file.css" />
  </head>
  <body>
    <h1>BCC481: Programação Web</h1>
    <p>
      <strong>News:</strong><br/>
      Horários de atendimento disponibilizados<br/>
      <span>Bora estudar!</span><br/>
    </p>
    <br/>
    <a href="http://ufop.br"> UFOP </a>
  </body>
</html>
```

Exemplo

```
body {  
    font-family: Helvetica;  
    text-align: center;  
}  
  
p {  
    border: 3px solid hotpink;  
    background-color: lavenderblush;  
    display: inline-block;  
    text-align: left;  
}  
  
span {  
    background-color: yellow;  
}
```

Exemplo

```
<!DOCTYPE html>
<html>
  <head>
    <title>BCC481</title>
    <link rel="stylesheet" href="css/css_file.css"
  />
</head>
<body>
  <h1>BCC481: Programação Web</h1>
  <p>
    <strong>News:</strong><br/>
    Horários de atendimento
    disponibilizados<br/>
    <span>Bora estudar!</span><br/>
  </p>
  <br/>
  <a href="http://ufop.br"> UFOP </a>
</body>
</html>
```



```
body {
  font-family: Helvetica;
  text-align: center;
}

p {
  border: 3px solid hotpink;
  background-color: lavenderblush;
  display: inline-block;
  text-align: left;
}

span {
  background-color: yellow;
}
```