

CSS



CSS

É a terceira mais nova versão das famosas **Cascading Style Sheets** (ou simplesmente CSS), pela qual se define estilos para um projeto web (página de internet).

Através de efeitos de transição, imagem, imagem de fundo/background e outros, pode-se criar estilos únicos para seus projetos web, alterando diversos aspectos de design no layout da página.

A principal função do CSS3 é abolir as imagens de plano/Background de fundo, implementar bordas arredondadas, apresentar transições e efeitos para criar animações de vários tipos.

CSS

Capacidades do CSS3

Construir animações tanto em 2 como em 3 dimensões:

- Os efeitos de rotação;
- Movimento; e
- Transição.

CSS

Uso do CSS na tag do HTML

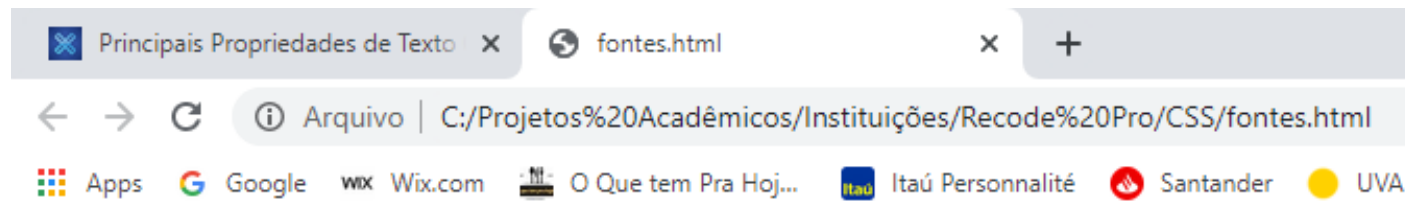
Exemplos:

`<p style="color: blue">Parágrafo com fonte azul</p>`

`<p style="color: red">Esse outro parágrafo não é azul</p>`

`<p style="color: red; font-family: arial">O carro é veloz</p>`

`<p style="color: green; font-size: 30px; font-family: fantasy">O caminhão não chegou</p>`



Parágrafo com fonte azul

Esse outro parágrafo não é azul

O carro é veloz

O caminhão não chegou

Atividade Prática 1

Duração da atividade é de 20 minutos.

Criar um código HTML com CSS de nome texto1.html

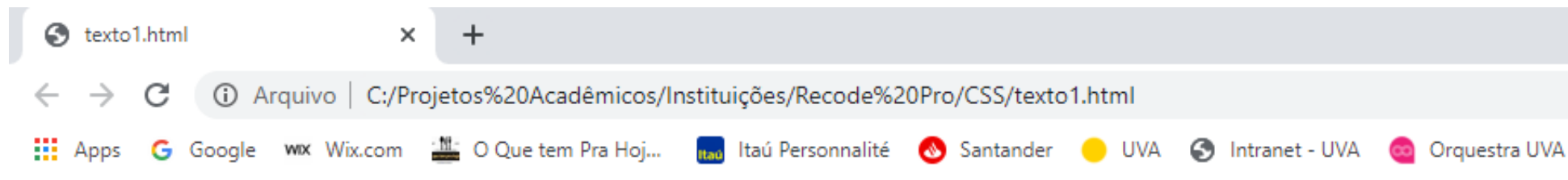
1. Criar um título na tag <head> de nome “Iniciando os exercícios com texto”.
2. Criar um título na tag <body> de nome “Estilos de Textos”, usar a cor gray.
3. Criar um parágrafo de com a informação “O mar quando está na cor azul me faz acalmar”, usar a cor azul.
4. Criar um parágrafo de com a informação “Adoro uma picanha mal passada”, usar a cor vermelha, usar a fonte de tamanho 20px.
5. Criar um parágrafo de com a informação “Amanhã volto ao trabalho”, usar a cor amarela, usar a fonte do tipo arial.
6. Criar um parágrafo de com a informação “O salário mínimo teve aumento”, usar a cor verde, , usar a fonte de tamanho 20px, usar a fonte do tipo fantasy.

Atividade Prática 1

Gabarito do exercício 1:

```
<html>
  <head>
    <title>Iniciando os exercícios com texto</title>
  </head>
  <body>
    <h1 style="color: gray">Estilos de Textos</h1>
    <p style="color: blue">O mar quando está na cor azul me faz acalmar</p>
    <p style="color: red; font-size: 20px">Adoro uma picanha mal passada</p>
    <p style="color: yellow; font-family: arial">Amanhã volto ao trabalho</p>
    <p style="color: green; font-size: 30px; font-family: fantasy">O salário mínimo teve
    aumento</p>
  </body>
</html>
```

Atividade Prática 1



Iniciando os exercícios com texto

Estilos de Textos

O mar quando está na cor azul me faz acalmar

Adoro uma picanha mal passada

Amanhã volto ao trabalho

O salário mínimo teve aumento

Atividade Prática 2

Duração da atividade é de 30 minutos.

Criar um código HTML com CSS de nome texto2.html

1. Todas as informações de título e parágrafo ficam por sua conta.
2. Criar um título na tag <head>.
3. Criar um outro título na tag <head> usar a cor preta e a fonte de tamanho 30px.
4. Criar um título na tag <body>, usar a cor red e a fonte de tamanho 30px.
 1. Criar um parágrafo com uma linha, usar a cor laranja.
 2. Criar um outro parágrafo com duas linhas, usar a cor vermelha (usar
)
5. Criar um título na tag <body>, usar a cor verde e a fonte de tamanho 20px.
 1. Criar um parágrafo com quatro linhas, usar a cor amarela, usar a fonte de tamanho 10px, usar a fonte do tipo fantasy.
 2. Criar um outro parágrafo com duas linhas, usar a cor vermelha
6. Criar um parágrafo, usar a cor marron, usar a fonte do tipo arial.

Atividade Prática 2

Gabarito do exercício 2: (feito pelo aluno Raphael)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Exercício 2</title>
</head>

<body>

  <h1 style="color: black; font-size: 30px;">Música: Indios</h1>

  <h1 style="color: red; font-size: 30px;">Legião Urbana</h1>

  <p style="color: orange;">Quem me dera ao menos uma vez Ter de volta todo o ouro que entreguei a quem</p>

  <p style="color: red;">Conseguiu me convencer que era prova de amizade <br>
```

Atividade Prática 2

Se alguém levasse embora até o que eu não tinha</p>

<h1 style="color: green; font-size: 20px;">Quem me dera ao menos uma vez
Esquecer que acreditei que era por brincadeira</h1>

<p style="color: yellow; font-size: 10px; font-family: fantasy;">Que se cortava sempre um pano de chão

De linho nobre e pura seda

Quem me dera ao menos uma vez

Explicar o que ninguém consegue entender</p>

<p style="color: red;">Que o que aconteceu ainda está por vir

E o futuro não é mais como era antigamente</p>

<p style="color: brown; font-family: Arial;"> Quem me dera ao menos uma vez Provar que quem tem
mais do que precisa ter</p>

</body>

</html>

CSS

Propriedades CSS para trabalhar com lista

list-style → Maneira abreviada para definir DUAS propriedades.

Sintaxe: `list-style: list-style-type list-style-position list-style-image | initial | inherit;`

Exemplo:

```
ul {  
    list-style: square inside url("sqpurple.gif");  
}
```

CSS

list-style-position:

Sintaxe: list-style-position: inside | outside | initial | inherit;

Exemplo:

```
ul.a {  
    list-style-position: outside;  
}
```

```
ul.b {  
    list-style-position: inside;  
}
```

CSS

Valores válidos para propriedades CSS da lista

list-style-image:

Sintaxe: list-style-image: none | *url* | initial | inherit;

Exemplo:

```
ul {  
    list-style-image: url('sqpurple.gif');  
}
```

CSS

Propriedades CSS para trabalhar com lista

list-style-type → Define o tipo do marcador da lista;

Sintaxe: list-style-type: *value*;

Exemplo:

```
ul.a {list-style-type: circle;}
```

```
ul.b {list-style-type: square;}
```

```
ol.c {list-style-type: upper-roman;}
```

```
ol.d {list-style-type: lower-alpha;}
```

CSS

list-style-type:

1. none: sem marcador
2. disc: círculo (bolinha cheia)
3. circle: circunferência (bolinha vazia)
4. square: quadrado cheio
5. decimal: números 1, 2, 3, 4, ...
6. decimal-leading-zero: decimal com 0 no início
7. lower-roman: romano minúsculo i, ii, iii, iv, ...
8. upper-roman: romano maiúsculo I, II, III, IV, ...
9. lower-alpha: letra minúscula a, b, c, d, ...
10. upper-alpha: letra maiúscula A, B, C, D, ...
11. lower-greek

Atividade Prática 3

Exercício 3:

Duração da atividade é de 40 minutos.

Criar um código HTML com CSS de nome texto3.html

1. Todas as informações de título e parágrafo ficam por sua conta.
2. Criar um título na tag <head>.
3. Criar seis título na tag <body>, usar a cor de sua escolha, a fonte de tamanho de sua escolha, usar a fonte de tipo de sua escolha.
4. Os títulos ímpares deverão estar em itálico e os títulos pares deverão estar sublinhado.
5. O primeiro título deverá ter uma imagem abaixo dele.
6. O segundo título deverá ter um parágrafo onde a formatação fica por sua conta.
7. O terceiro parágrafo deverá ter um texto com duas linhas, usar a cor rosa, usar a fonte de tamanho 15px, usar a fonte do tipo fantasy.
8. Após o terceiro parágrafo deve ter uma lista não ordenada de 5 itens.
9. Após o item 8 deve ter uma lista ordenada de 6 itens.
10. Após o item 9 deve ter uma tabela com quatro linhas e três colunas. Você define as informações.

Atividade Prática 3

Gabarito do exercício 3 (feito pelo aluno Marcus Vinicius):

```
<!DOCTYPE html>
<html lang="pt-br">

<head>
  <meta charset="UTF-8">
  <title>Exercício 3</title>
</head>

<body>
  <h1 style="color: skyblue; font-size: 20px; font-family: Verdana, Geneva, Tahoma, sans-serif; font-style: italic;">E viver</h1>
  
  <h2 style="color: blue; font-size: 18px; font-family: Geneva, Tahoma, sans-serif; text-decoration: underline;">E cantar</h2>
  <p style="color: gray; background-color: skyblue; font-style: italic; font-size: 22px;">Tão natural quanto a luz do dia</p>
```

Atividade Prática 3

```
<h3 style="color: skyblue; font-size: 16px; font-family: Tahoma, sans-serif; font-style: italic;">Não  
importa qual seja o dia</h3>
```

```
<p style="color: pink; font-size: 15px; font-family: fantasy;">Mas que preguiça boa<br>
```

```
Me deixa aqui à toa</p>
```

```
<h4 style="color: blue; font-size: 14px; font-family: Verdana, Geneva, Tahoma, sans-serif; text-  
decoration: underline;">
```

```
Vamos viver</h4>
```

```
<ul style="list-style-type: square;">
```

```
<li>Tão natural quanto a luz do dia</li>
```

```
<li>Mas que preguiça boa</li>
```

```
<li>Me deixa aqui à toa</li>
```

```
<li>Hoje ninguém vai estragar meu dia</li>
```

```
<li>Só vou gastar energia pra beijar sua boca</li>
```

```
</ul>
```

Atividade Prática 3

```
<h5 style="color: skyblue; font-size: 12px; font-family: sans-serif; font-style: italic;">Vadiar</h5>
```

```
<ol style="list-style-type: upper-alpha;">
```

```
<li>Fica comigo então</li>
```

```
<li>Não me abandona, não</li>
```

```
<li>Alguém te perguntou como é que foi seu dia?</li>
```

```
<li>Uma palavra amiga, uma notícia boa</li>
```

```
<li>Isso faz falta no dia a dia</li>
```

```
<li>A gente nunca sabe quem são essas pessoas</li>
```

```
</ol>
```

```
<h6
```

```
style="color: blue; font-size: 20px; font-family: Verdana, Geneva, Tahoma, sans-serif; text-decoration: underline;">
```

```
O que importa é nossa
```

```
alegria</h6>
```

Atividade Prática 3

```
<tr>
```

```
  <td>Fica</td>
```

```
  <td>comigo</td>
```

```
  <td>então</td>
```

```
</tr>
```

```
<tr>
```

```
  <td>Não</td>
```

```
  <td>me</td>
```

```
  <td>abandona</td>
```

```
</tr>
```

```
<tr>
```

```
  <td>Uma</td>
```

```
  <td>palavra</td>
```

```
  <td>amiga</td>
```

```
</tr>
```

Atividade Prática 3

```
<tr>
  <td>uma</td>
  <td>notícia</td>
  <td>boa</td>
</tr>
</table>
</body>

</html>
```

CSS

Modelo de referência para trabalhar com o CSS

- Elemento;
- Id; e
- Classes.

Formas de uso do CSS

1. Como propriedade de um elemento HTML;
2. Como uma tag de HTML;e
3. Como um arquivo separado do projeto.

CSS

O CSS tem:

- Sintaxe simples;
- Utiliza uma série de palavras em inglês para especificar os nomes de diferentes propriedade de estilo de uma página.

Uma instrução CSS consiste em um **seletor** e um **bloco de declaração**.

Cada declaração contém uma **propriedade** e um **valor**, separados por dois pontos (:).

Cada declaração é separada por ponto e vírgula (;).

CSS

Selector (seletor) → é a referencia para o elemento (tag) que queremos utilizar. No exemplo acima, utilizamos "**p**" para estilizar os elementos de parágrafo (<**p**>).

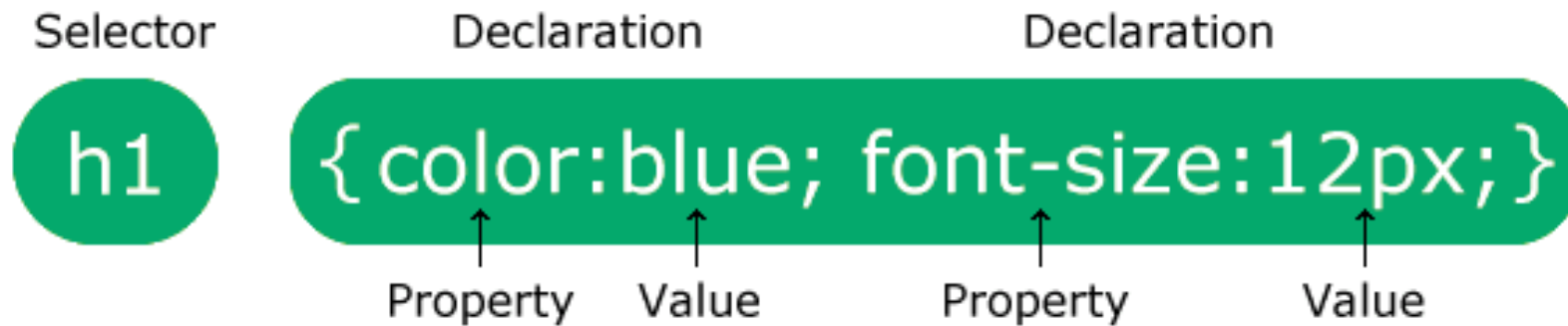
Property (propriedade) → é o atributo do elemento que será alterado e estilizado. Nesse caso , iremos modificar sua **cor** (color).

Property value (Valor da propriedade) → é o valor da propriedade, no exemplo acima, informamos que o parágrafo **p** terá a cor **vermelha**.

Declaration (Declaração) → é a junção da propriedade com o valor, lembre-se que ela **sempre deve terminar com ";"**.

CSS

Sintaxe:



Exemplo: todos os elementos `<p>` serão alinhados ao centro, com uma cor de texto vermelha:

```
p {
  color: red;
  text-align: center;
}
```

CSS

Seletor do tipo elemento

O seletor do tipo **elemento**.

Para usar neste formato é preciso que haja a tag `<style> .. </style>`

A declaração deve acontecer na tag `<head> .. </head>`, toda a formatação feita surtirá como efeito nas formatações abaixo:

- `h1 {/* estilo para todo o título */}`
- `p {/* estilo para todos os parágrafos */}`
- `body {/* estilo para todo o corpo da pagina}`
- `img {/* estilo para todas as imagens */}`
- `ul li {/* estilo para todos os itens da lista */}`

CSS

Uso do CSS como uma propriedade da tag do HTML

Este efeito funciona para o que está para a tag <head> em diante.
Neste formato, o seletor "**p**" representa o elemento <p>.

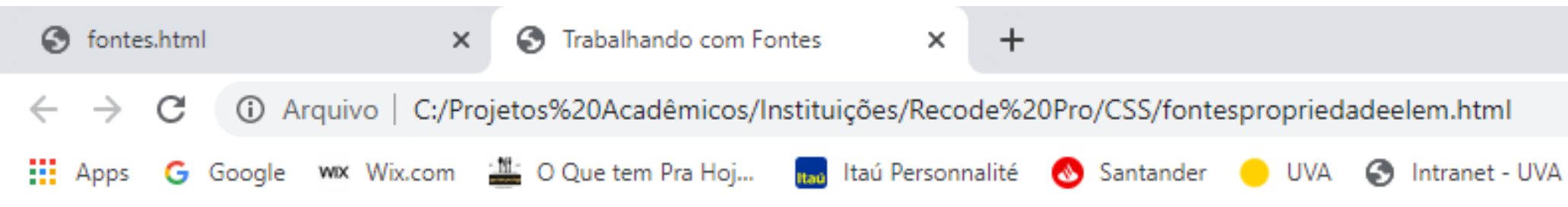
Exemplo com uso de texto:

```
<html>
  <head>
    <title>Trabalhando com Fontes</title>
    <style>
      p {color: green; font-size: 30px}
    </style>
  </head>
  <body>
    <p>Parágrafo com fonte</p>
    <p>Esse outro parágrafo</p>
    <p>O carro é veloz</p>
  </body>
</html>
```



Reflete em tudo que está
no <body>

CSS



Parágrafo com fonte

Esse outro parágrafo

O carro é veloz

CSS

Propriedades CSS para trabalhar com texto

color → muda a cor do texto.

Exemplo:

```
<h1 style="background-color:rgb(255, 99, 71);">Olá</h1>
```

```
<h1 style="background-color:#ff6347;">Olá</h1>
```

```
<h1 style="background-color:hsl(9, 100%, 64%);">Olá</h1>
```

Tabela de cores: <https://erikasarti.com/html/tabela-cores/>

CSS

Propriedades CSS para trabalhar com texto

font-family → define uma lista de fontes para ser utilizada no elemento (e seus herdeiros). Os nomes das fontes são separados por vírgulas, na ordem da preferida para a menos preferida. Fontes que possuem mais que uma palavra como nome tem que estar entre aspas.

Exemplo:

font-family: "Trebuchet MS", Arial, Tahoma, sans-serif;. Se o usuário tiver a fonte Trebuchet MS, o navegador irá usá-la; caso contrário, Arial; caso contrário, Tahoma; e se não houver nenhuma destas fontes instaladas, a fonte padrão será a sans-serifa.

CSS

font-size → muda o tamanho do texto. Podemos utilizar unidades como px (pixels), em (unidade relativa), %, pt (pontos), entre outras.

Exemplo: font-size: 20px.

font-weight → muda o peso da fonte. Valores incluem bold e normal.

Exemplo: font-weight: bold;

font-style → muda o estilo da fonte. Valores incluem italic e normal.

Exemplo: font-style: italic;

CSS

text-decoration → muda a decoração do texto. Valores incluem underline e none (ex: para retirar o sublinhado de links).

Exemplo: text-decoration: underline;

line-height → muda a altura de cada linha, ou *entrelinha*. É comum utilizar a unidade em, pois assim deixamos a entrelinha proporcional ao tamanho da fonte.

Exemplo: line-height: 1.5em; (1.5 vezes o tamanho da fonte).

text-indent → muda a indentação da primeira linha de um bloco de texto.

Exemplo: text-indent: 2em; (indentação de 2 vezes o tamanho da fonte na primeira linha do texto).

CSS

line-height → muda a altura de cada linha, ou *entrelinha*. É comum utilizar a unidade em, pois assim deixamos a entrelinha proporcional ao tamanho da fonte.

Exemplo: line-height: 1.5em; (1.5 vezes o tamanho da fonte).

text-indent → muda a indentação da primeira linha de um bloco de texto.

Exemplo: text-indent: 2em; (indentação de 2 vezes o tamanho da fonte na primeira linha do texto).

word-spacing → determina o espaçamento entre palavras.

Exemplo: word-spacing: 5px;

CSS

text-transform → muda a "caixa" do texto. Valores incluem uppercase (caixa-alta), lowercase (caixa-baixa) e capitalize (primeira letra maiúscula e o restante minúsculas).

Exemplo: text-transform: uppercase;

font-variant → muda a variante da fonte. Valores incluem normal e small-caps (versalete).

Exemplo: font-variant: small-caps;

letter-spacing → determina o espaçamento entre letras.

Exemplo: letter-spacing: 2px;

Atividade Prática 4

Duração da atividade é de 20 minutos.

Criar um código HTML com CSS de nome texto4.html

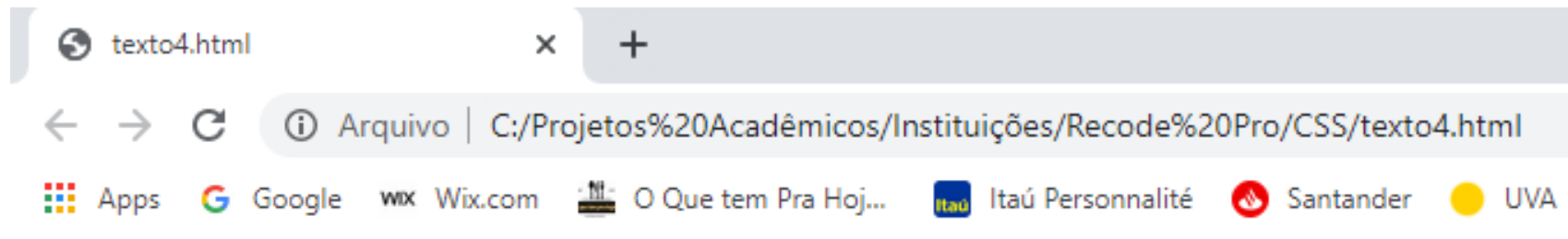
1. Criar um título na tag <head> com a informação “Exercícios com textos”, usar a cor cornflowerblue, a fonte de tamanho 20px , deve estar em negrito.
2. Criar uma configuração para atender a todos os parágrafos abaixo com as seguintes configurações: usar a cor firebrick, usar a fonte de 40px, usar a fonte de tipo sans-serif.
3. Criar um parágrafo na tag <body> com a informação “Existem muitas possibilidades de fontes”.
4. Criar um parágrafo na tag <body> com a informação “As opções são diversas”.
5. Criar um parágrafo na tag <body> com a informação “Existem muitas cores para se trabalhar”.
6. Colocar uma linha de separação após o item 4.
7. Criar um parágrafo na tag <body> com a informação “Ainda há muitas opções para ver”.
8. Criar um parágrafo na tag <body> com a informação “Existem várias possibilidades”.
9. Colocar uma linha de separação após o item 7.

Atividade Prática 4

Gabarito do exercício 4:

```
<html>
  <head>
    <title>Exercícios com textos</title>
    <style>
      p {color: firebrick; font: 40px; font-family: sans-serif;}
    </style>
  </head>
  <body>
    <h1 style="color: cornflowerblue; font-size: 20px; font-weight: bold">Exercícios
    com textos</h1>
    <p>Existem muitas possibilidades de fontes</p>
    <p>As opções são diversas</p>
    <p>Existem muitas cores para se trabalhar</p>
    <hr>
    <p>Ainda há muitas opções para ver</p>
    <p>Existem várias possibilidades</p>
    <hr>
  </body>
</html>
```

Atividade Prática 4



Exercícios com textos

Existem muitas possibilidades de fontes

As opções são diversas

Existem muitas cores para se trabalhar

Ainda há muitas opções para ver

Existem várias possibilidades

Atividade Prática 5

Duração da atividade é de 30 minutos.

Criar um código HTML com CSS de nome texto5.html

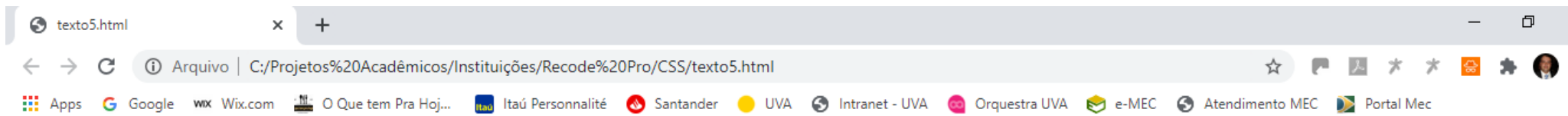
1. Criar um título na tag <body> com a informação “Exercícios com textos Avançados”, usar a cor #ff00ff, a fonte de tamanho 30px , deve estar em negrito.
2. Criar um outro título na tag <body> com a informação “Temas Inovadores”, usar a cor #ffff00, a fonte de tamanho 20px , deve estar em itálico.
3. Criar uma configuração para atender a todos os parágrafos abaixo com as seguintes configurações: usar a cor coral, usar a fonte de 10px, usar a fonte de tipo fantasy e serif.
4. Criar uma segunda configuração para atender a todos os parágrafos abaixo com as seguintes configurações: usar texto decorado com underline, usar texto de alinhamento no centro.
5. Criar um parágrafo na tag <body> com a informação “Veraneio Vascaína”.
6. Criar um parágrafo na tag <body> com a informação “Vem dobrando a esquina”.
7. Colocar uma linha de separação após o item 6.
8. Colocar uma linha de separação após o item 7.
9. Criar um parágrafo na tag <body> com a informação “Esta vida me maltrata”.
10. Criar um parágrafo na tag <body> com a informação “Estou virando psicopata”.
11. Colocar uma linha de separação após o item 10.

Atividade Prática 5

Gabarito do exercício 5:

```
<html>
  <head>
    <style>
      p {color: coral; font: 10px; font-family: fantasy, serif}
      p {text-decoration: underline; text-align: center}
    </style>
  </head>
  <body>
    <h1 style="color: #ff00ff; font-size: 40px; font-weight: bold">Exercícios com textos Avançados</h1>
    <br>
    <h2 style="color: #ffff00; font-size: 30px; font-style: italic">Temas Inovadores</h2>
    <hr>
    <p>Veraneio Vascaína</p>
    <p>vem dobrando a esquina</p>
    <hr>
    <hr>
    <p>Esta vida me maltrata</p>
    <p>Estou virando psicopata</p>
    <hr>
  </body>
</html>
```

Atividade Prática 5



Exercícios com textos

Temas Inovadores

Verancio Vascaína

vem dobrando a esquina

Esta vida me maltrata

Estou virando psicopata

CSS

Valores válidos para o Uso de Cores

background-color:

Sintaxe: background-color: *color* | transparent | initial | inherit;

Exemplo: body {background-color: coral;}

Formas de aplicação:

- código hexadecimal: #ffc6d9
- código rgb: rgb(255,235,0)
- código rgba: rgba(255,235,0, 0.7)
- código hsl: hsl(210,100%,40%)
- código hsla: hsla(155,80%,35%,0.4)
- palavra-chave: red, blue, green...etc
- transparente: transparente

CSS

Propriedades CSS para trabalhar fundos de tela

background-color → Define a cor do fundo;

Exemplo: `<h1 style="background-color:DodgerBlue;">Olá</h1>`

background-image → Define uma imagem de fundo;

Exemplo: `background-image: url("paper.gif");`

CSS

background-repeat → Define a maneira como a imagem de fundo é posicionada;

Exemplo:

```
body {  
    background-image: url("paper.gif");  
    background-repeat: repeat-y;  
}
```

CSS

background-attachment → Define se a imagem de fundo "rola" ou não com a tela;

Exemplo:

```
body {  
    background-image: url("img_tree.gif");  
    background-repeat: no-repeat;  
    background-attachment: fixed;  
}
```

CSS

background-position → Define como e onde a imagem de fundo é posicionada;

Exemplo:

```
body {  
    background-image: url('w3css.gif');  
    background-repeat: no-repeat;  
    background-attachment: fixed;  
    background-position: center;  
}
```

CSS

background-clip → Define a área do box onde a imagem de fundo é aplicada;

Exemplo:

```
div {  
    border: 10px dotted black;  
    padding: 15px;  
    background: lightblue;  
    background-clip: padding-box;  
}
```

CSS

background-origin → Define a posição de origem da imagem no box;

Exemplo:

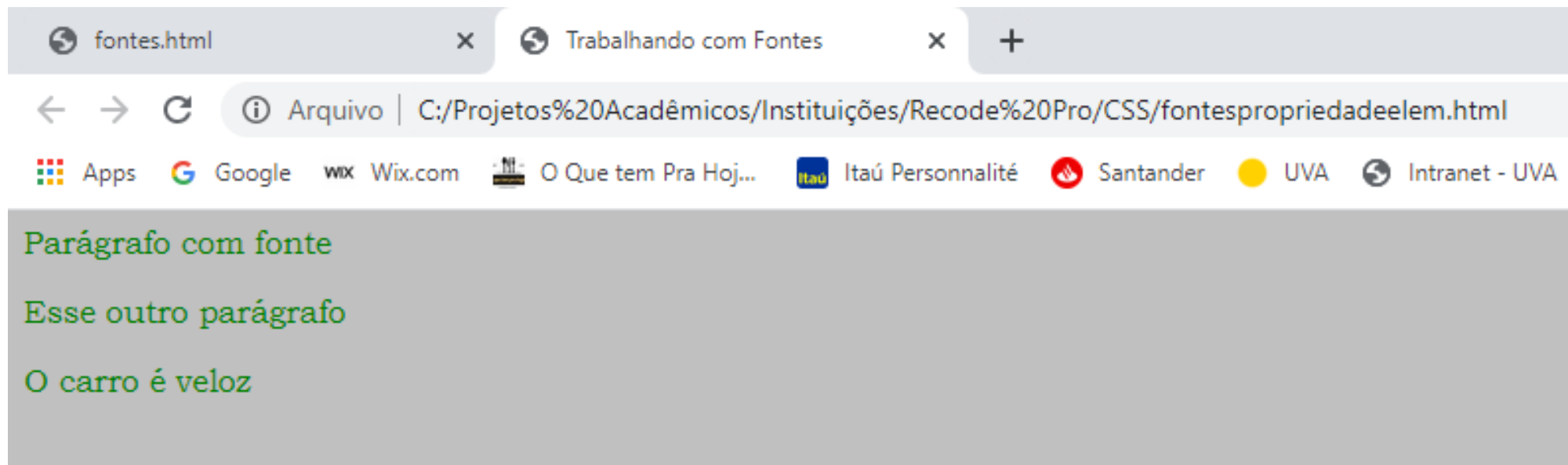
```
#example1 {  
    border: 10px dashed black;  
    padding: 25px;  
    background: url(paper.gif);  
    background-repeat: no-repeat;  
    background-origin: content-box;  
}
```

CSS

Exemplo do CSS com uso de texto e background

```
<html>
  <head>
    <title>Trabalhando com Fontes</title>
    <style>
      p {color: green; font: 30px}
      body {background-color: silver}
    </style>
  </head>
  <body>
    <p>Parágrafo com fonte</p>
    <p>Esse outro parágrafo</p>
    <p>O carro é veloz</p>
  </body>
</html>
```


CSS

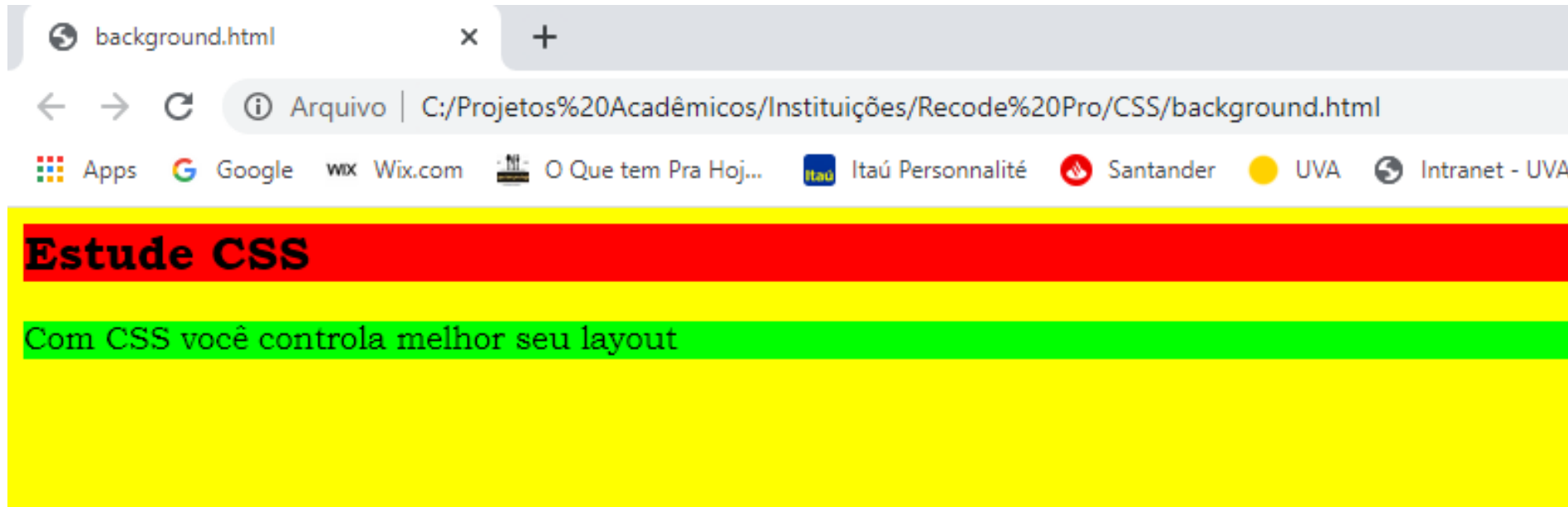


CSS

Exemplo do uso CSS com background

```
<html>
  <head>
    <style>
      body { background-color: yellow; } /* amarelo */
      h2 { background-color: #f00; } /* vermelho */
      p { background-color: #0f0; } /* verde */
    </style>
  </head>
  <body>
    <h2>Estude CSS</h2>
    <p>Com CSS você controla melhor seu layout</p>
  </body>
</html>
```

CSS



CSS

Propriedades CSS para trabalhar com bordas

border-width → Define a espessura das quatro bordas.

Sintaxe: border-width: medium | thin | thick | *length* | initial | inherit;

Exemplo: div {border-width: medium;}

border-style → Define o estilo das quatro bordas.

Sintaxe: border-style:

none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset | initial | inherit;

Exemplo: div {border-style: solid;}

CSS

border-color → Define a cor das quatro bordas.

Sintaxe: border-color: *color* | transparent | initial | inherit;

Exemplo: div {border-color: coral;}

border-top-width → Define a espessura da borda superior.

Sintaxe: border-top-width:
medium | thin | thick | *length* | initial | inherit;

Exemplo: div {border-top-width: thin;}

CSS

border-top-style → Define o estilo da borda superior.

Sintaxe: border-top-style:

none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset | initial | inherit;

Exemplo: div {border-top-style: dotted;}

border-top-color → Define a cor da borda superior.

Sintaxe: border-top-color: *color* | transparent | initial | inherit;

Exemplo: div {border-top-color: coral;}

CSS

border-right-width → Define a espessura da borda direita.

Sintaxe: border-right-width:

medium | thin | thick | *length* | initial | inherit;

Exemplo: div {border-right-width: thin;}

border-right-style → Define a estilo da borda direita.

Sintaxe: border-right-style:

none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset | initial | inherit;

Exemplo: div {border-right-style: dotted;}

CSS

border-right-color → Define a cor da borda direita.

Sintaxe: border-right-color: *color* | transparent | initial | inherit;

Exemplo: div {border-right-color: coral;}

border-bottom-width → Define a espessura da borda inferior.

Sintaxe: border-bottom-width:
medium | thin | thick | *length* | initial | inherit;

Exemplo: div {border-bottom-width: thick;}

CSS

border-bottom-style → Define o estilo da borda inferior.

Sintaxe: border-bottom-style:

none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset | initial | inherit;

Exemplo: div {border-bottom-style: dotted;}

border-bottom-color → Define a cor da borda inferior.

Sintaxe: border-bottom-color: *color* | transparent | initial | inherit;

Exemplo: div {border-bottom-color: coral;}

CSS

border-left-width → Define a espessura da borda esquerda.

Sintaxe: `div {border-left-width: thin;}`

Exemplo:

border-left-style → Define o estilo da borda esquerda.

Sintaxe: `border-left-style:`

`none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset | initial | inherit;`

Exemplo: `div {border-left-style: dotted;}`

CSS

border-left-color → define a cor da borda esquerda.

Sintaxe: border-left-color: *color* | transparent | initial | inherit;

Exemplo: div {border-left-color: coral;}

border-top → Maneira abreviada para todas as propriedades da borda superior.

Sintaxe: border-top: *border-width border-style border-color* | initial | inherit;

Exemplo:

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

CSS

border-right → Maneira abreviada para todas as propriedades da borda direita.

Sintaxe: `border-right: border-width border-style border-color | initial | inherit;`

Exemplo:

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

CSS

border-bottom → Maneira abreviada para todas as propriedades da borda inferior.

Sintaxe: `border-bottom: border-width border-style border-color | initial | inherit;`

Exemplo:

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

CSS

border-left → Maneira abreviada para todas as propriedades da borda esquerda.

Sintaxe: `border-left: border-width border-style border-color | initial | inherit;`

Exemplo:

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

CSS

border → Maneira abreviada para todas as quatro bordas.

Sintaxe: border: *border-width border-style border-color* | initial | inherit;

Exemplo:

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

CSS

border-radius → Define o raio dos cantos do elemento.

Sintaxe: `border-radius: 1-4 length | % / 1-4 length | % | initial | inherit;`

Exemplo:

```
#example1 {  
    border: 2px solid red;  
    border-radius: 25px;  
}
```

```
#example2 {  
    border: 2px solid red;  
    border-radius: 50px 20px;  
}
```

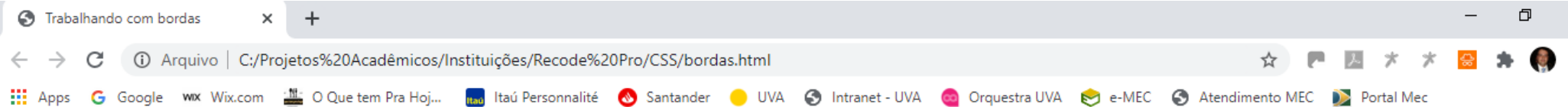

CSS

Exemplo do CSS com uso de bordas:

```
<html>
  <head>
    <title>Trabalhando com bordas</title>
    <style>
      h3 { border-width: medium;
            border-style: solid;
            border-color: #00f;}

      p { border-width: 6px;
          border-style: dashed;
          border-color: #f00;}
    </style>
  </head>
  <body>
    <h3>Borda média, contínua e azul</h3>
    <p>Borda 6px, tracejada e vermelha</p>
  </body>
</html>
```

CSS



Borda média, contínua e azul

Borda 6px, tracejada e vermelha

CSS

Seletor do tipo "id"

O **id**, ou **identificador**, é uma forma interessante para resolver o problema apresentado lá no início do módulo. O **id** é uma propriedade que identifica o elemento, **podendo ser acionada diretamente pelo CSS**.

Para que o CSS saiba que você está querendo selecionar o elemento pelo seu identificador (id), basta usar o caractere especial cerquilha (**#**), seguido pelo nome do identificador escolhido.

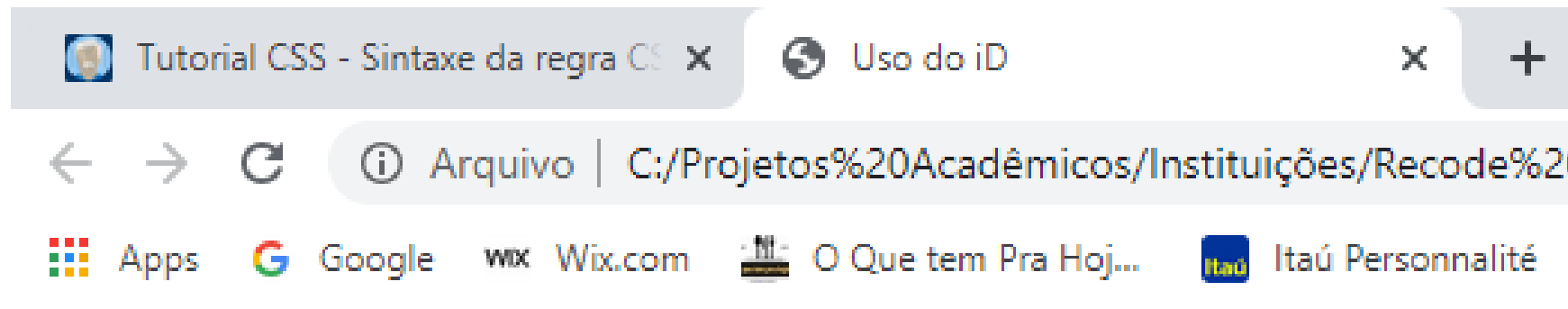
CSS

Exemplo do código:

```
<!DOCTYPE HTML>
<html>
  <style>
    p{color: red}
    #textoazul{color: blue}
    #textoverde{color: green}
  </style>

  <head>
    <title>Uso do iD</title>
  </head>
  <body>
    <p id="textoazul">O texto ficará na cor azul</p>
    <p>O texto ficará na cor vermelha</p>
  </body>
</html>
```

CSS



O texto ficará na cor azul

O texto ficará na cor vermelha

CSS

Seletor do tipo "classe"

Esse seletor, como o próprio nome sugere, permite criar categorias dos elementos, agrupando-os por **classe de estilo**.

A estrutura desse seletor é composta por um ponto (.) seguido de um nome que você escolhe para definir o nome da classe.

No momento, é usada a propriedade **"class"** para definir qual à classe ele pertence.

CSS

Exemplo do código:

```
<!DOCTYPE HTML>
```

```
<html>
```

```
  <style>
```

```
    .azul{color: blue}
```

```
    .verde{color: green}
```

```
    .vermelho{color: red}
```

```
    .fundoamarelo{background-color: yellow}
```

```
  </style>
```

```
  <head>
```

```
    <title>Uso da Classe</title>
```

```
  </head>
```

```
  <body>
```

```
    <p class="azul">O texto ficará na cor azul</p>
```

```
    <p class="verde">O texto ficará na cor verde</p>
```

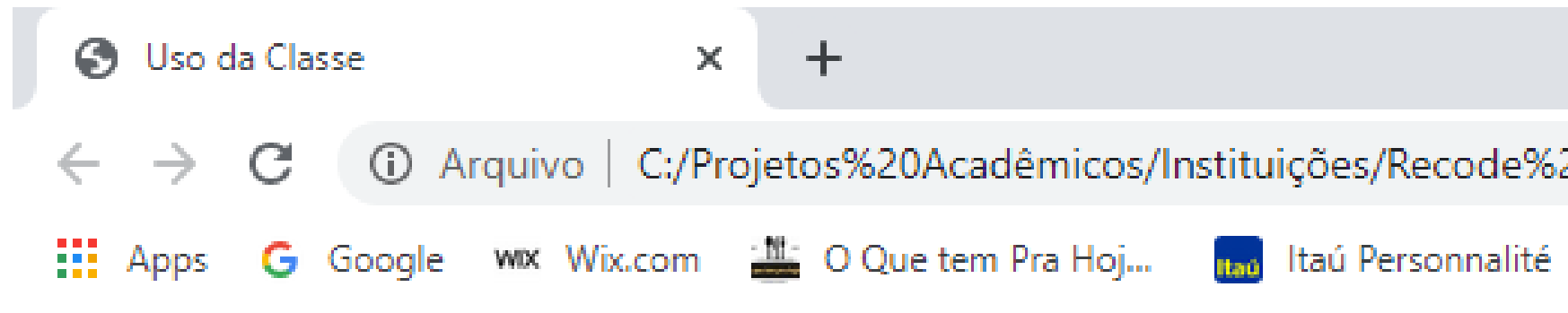
```
    <p class="vermelho fundoamarelo">O texto ficará na cor vermelho com fundo  
    amarelo</p>
```

```
  </body>
```

```
</html>
```

↑
Duas classes sendo usadas (separadas por espaço)

CSS



O texto ficará na cor azul


O texto ficará na cor verde

O texto ficará na cor vermelho com fundo amarelo

CSS

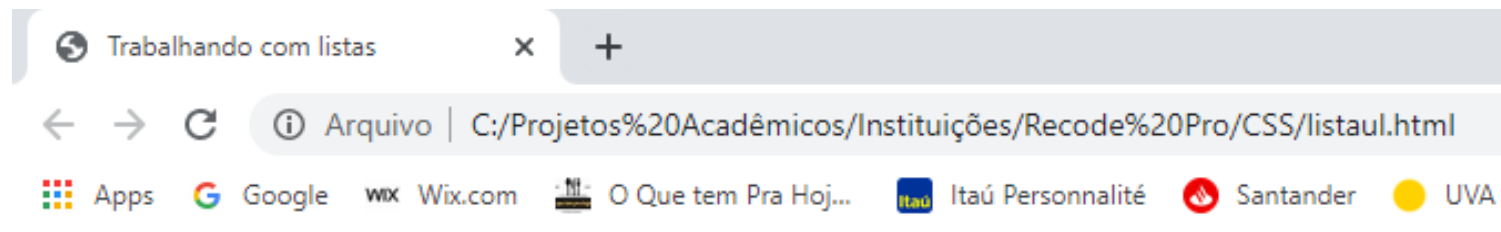
Exemplo do CSS com uso de lista:

```
<html>
  <head>
    <title>Trabalhando com listas</title>
    <style>
      ul.circle {list-style-type: circle;}
    </style>
  </head>
  <body>
    <p><b>Tipos de bebidas</b></p>
    <ul class="circle">
      <li>Item um</li>
      <li>Item dois</li>
      <li>Item três</li>
    </ul>
  </body>
</html>
```



Reflete em tudo que está
no <body>

CSS



Tipos de bebidas

- Item um
- Item dois
- Item três

CSS

Exemplo do CSS com uso de duas listas:

```
<html>
```

```
  <head>
```

```
    <title>Trabalhando com listas</title>
```

```
    <style>
```

```
      ul.circle {list-style-type: circle;}
```

```
      ul.square {list-style-type: square;}
```

```
    </style>
```

```
  </head>
```

```
  <body>
```

```
    <p><b>Tipos de bebidas</b></p>
```

```
    <ul class="circle">
```

```
      <li>Água</li>
```

```
      <li>Refrigerante</li>
```

```
      <li>Cerveja</li>
```

```
    </ul>
```

```
    <br>
```

```
    <p><b>Tipos de comidas</b></p>
```

```
    <ul class="square">
```

```
      <li>Pão</li>
```

```
      <li>Feijão</li>
```

```
      <li>carne</li>
```

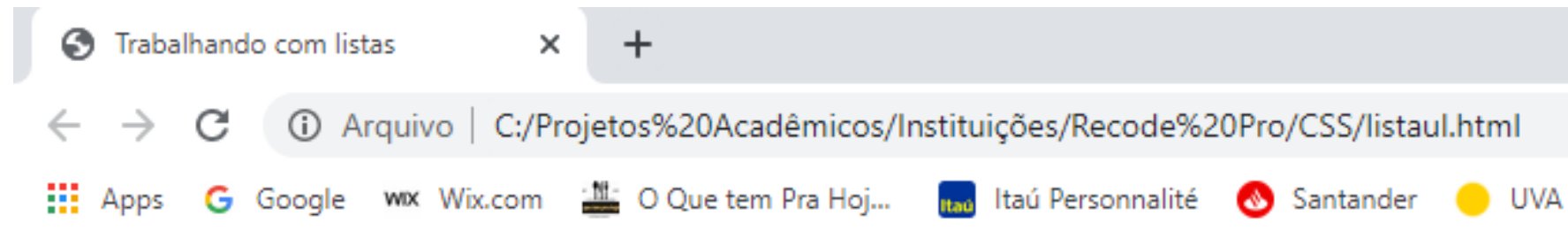
```
    </ul>
```

```
  </body>
```

```
</html>
```

Reflete em tudo que está
no <body>

CSS



Tipos de bebidas

- Água
- Refrigerante
- Cerveja

Tipos de comidas

- Pão
- Feijão
- carne

CSS

Exemplo “listaol.html”:

```
<html>
  <head>
    <title>Trabalhando com listas Ordenadas</title>
    <style>
      ol.teste1 {list-style-type: lower-latin}
      ol.teste2 {list-style-type: decimal-leading-zero;
                 list-style-position: inside}
    </style>
  </head>
  <body>
    <ol class="teste1">
      <li>Caf </li>
      <li>Leite</li>
      <li> gua</li>
    </ol>

    <ol class="teste2">
      <li>Coffee</li>
      <li>Tea</li>
      <li>Milk</li>
    </ol>
  </body>
</html>
```

CSS

Exercício 6:

1. Criar um arquivo html de exerformatos.html;
2. Colocar um título como uso tag <title>;
3. Definir um elemento para padrão de fonte arial e tamanho 20px;
4. Definir um elemento com fundo de tela cinza;
5. Definir uma classe de cor azul para o texto;
6. Definir uma classe de cor verde para o texto;
7. Definir uma classe de fundo branco;
8. Criar um parágrafo que use a classe azul;
9. Criar um parágrafo que use a classe verde;
10. Criar um parágrafo que use a cor do texto azul com fundo branco;

CSS

Seletor do tipo "classe" com arquivo externo

Também é possível criar um arquivo **.css** e criar uma ou mais classes ao invés de colocar a formatação dentro do documento HTML.

Para acessar as classes basta no arquivo HTML fazer referência ao link do arquivo **.css**.

A classe que foi criada no arquivo **.css** tem o nome de **.bloco**

Para criar a classe no arquivo **.css** é preciso que haja o “**ponto**” “.” antes do nome da classe. A mesma deve estar entre a **abertura** e o **fechamento das chaves** “**{}**”.

CSS

Classe criada no arquivo .css (.bloco)

Para criar a classe no arquivo .css é preciso que haja o “ponto” “.” antes do nome da classe. A mesma deve estar entre a abertura e o fechamento das chaves “{}”.

Este é o arquivo **estilo.css** criado para integrar com o outro arquivo html

```
.bloco { color: blue;  
        background-color: yellow;  
        text-align: center;  
}
```



características do bloco

CSS

No arquivo HTML para que a classe seja utilizada em um determinado ponto, é preciso utilizar a sessão `<section>`
`</section>` com a especificação do atributo `class` e o `nome da classe`.

A tag `<link>` define a relação entre o documento atual e um recurso externo. É mais frequentemente usada para vincular a folhas de estilo externas. É um elemento vazio, ele contém apenas atributos.

O atributo `rel` especifica a relação entre o documento atual e o link do outro documento.

CSS

Exemplo:

```
<!DOCTYPE HTML>
```

```
<html>
```

```
  <head>
```

```
    <title>Uso CSS externo</title>
```

```
  </head>
```

```
  <link rel="stylesheet" type="text/css" href="estilo.css">
```

```
  <section class="bloco">
```

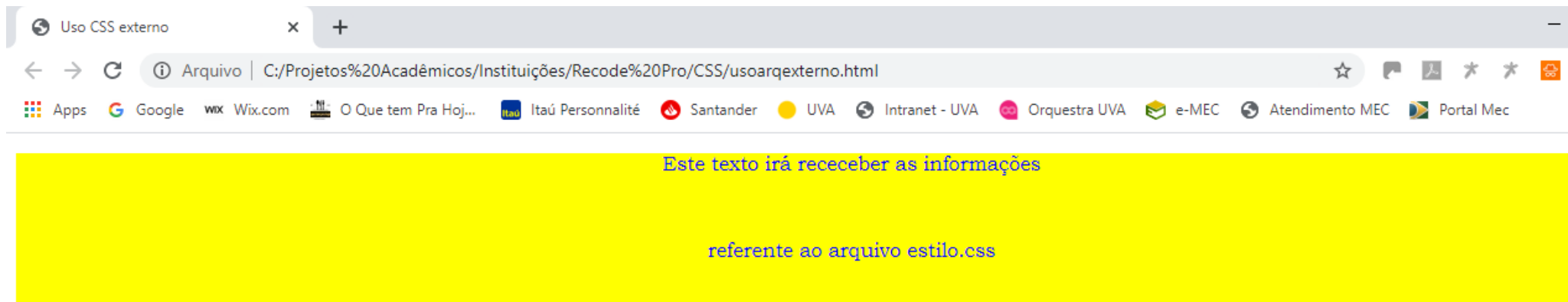
```
    <p>Este texto irá receber as informações</p><br>
```

```
    <p>referente ao arquivo estilo.css</p><br>
```

```
  </section>
```

```
</html>
```

CSS



CSS

Em qual momento devemos utilizar os seletores vistos?

1. Definir que todos os elementos da página terão o mesmo tamanho, é possível utilizar o seletor do tipo do **elemento** como algo mais genérico.
2. Se você pretende destacar apenas um único elemento específico, que não se repete na página, utilize o **id** no elemento e no seletor CSS.
3. Usar as **classes** em elementos que precisam se repetir algumas vezes em uma página, ou quando os elementos precisam acumular estilos de classes diferentes.

CSS

Exercício 7:

1. Criar um arquivo de nome `formatos.css`
 1. Criar uma formatação de fundo de tela, cor da fonte, tipo de fonte e centralização do texto;
2. Criar um arquivo html de `exercclass.html`;
3. Colocar um título como uso tag `<title>`;
4. Fazer o link para que os parágrafos tenham a formatação do arquivo externo `formato.css`;
5. Criar três parágrafos com texto livre;

CSS

Trabalhando com as larguras e alturas



CSS

Definindo altura e largura

```
1 <html>
2 <head>
3   <style>
4
5   body {
6     background-color: silver;
7   }
8
9   textarea{
10     width: 200px;
11     height: 200px;
12   }
13
14 </style>
15 <title>Curso de CSS</title>
16 </head>
17 <body>
18   <h2>Livro de visitas</h2>
19   <form>
20     <textarea>Escreva aqui todos os pontos positivos de sua visita,
      elogios e agradecimentos :-)</textarea>
21     <textarea>Escreva aqui todos os pontos negativos de sua visita,
      críticas e reclamações :-)</textarea>
22   </form>
23 </body>
24 </html>
```

CSS

Temos a página com dois formulários com o mesmo tamanho.



CSS

Definindo altura e largura com porcentagem

```
<!DOCTYPE html>
<html>
<head>
  <style>

  body {
    background-color: silver;
  }

  textarea{
    width: 50%;
    height: 100px;
  }

  .meio {
    width: 50%;
    background-color: orange;
  }
}
```

CSS

```
</style>
<title>Curso de CSS</title>
</head>
<body>
  <h2>Livro de visitas</h2>
  <form>
    <textarea>Escreva aqui todos os pontos positivos de sua visita, elogios e
    agradecimentos :-)</textarea>
    <textarea>Escreva aqui todos os pontos negativos de sua visita, críticas e
    reclamações :-)</textarea>
  </form>

  <form class="meio">
    <h3>Atração preferida</h3>
    <textarea>Escreva aqui todos os pontos negativos de sua visita, críticas e
    reclamações :-)</textarea>
  </form>
</body>
</html>
```

CSS



CSS

Trabalhando com as bordas

Estrutura de espaçamento no CSS é a propriedade ***border***, que define as bordas do elemento.



```
1 p {  
2     border: 5px solid blue;  
3 }
```

CSS

```

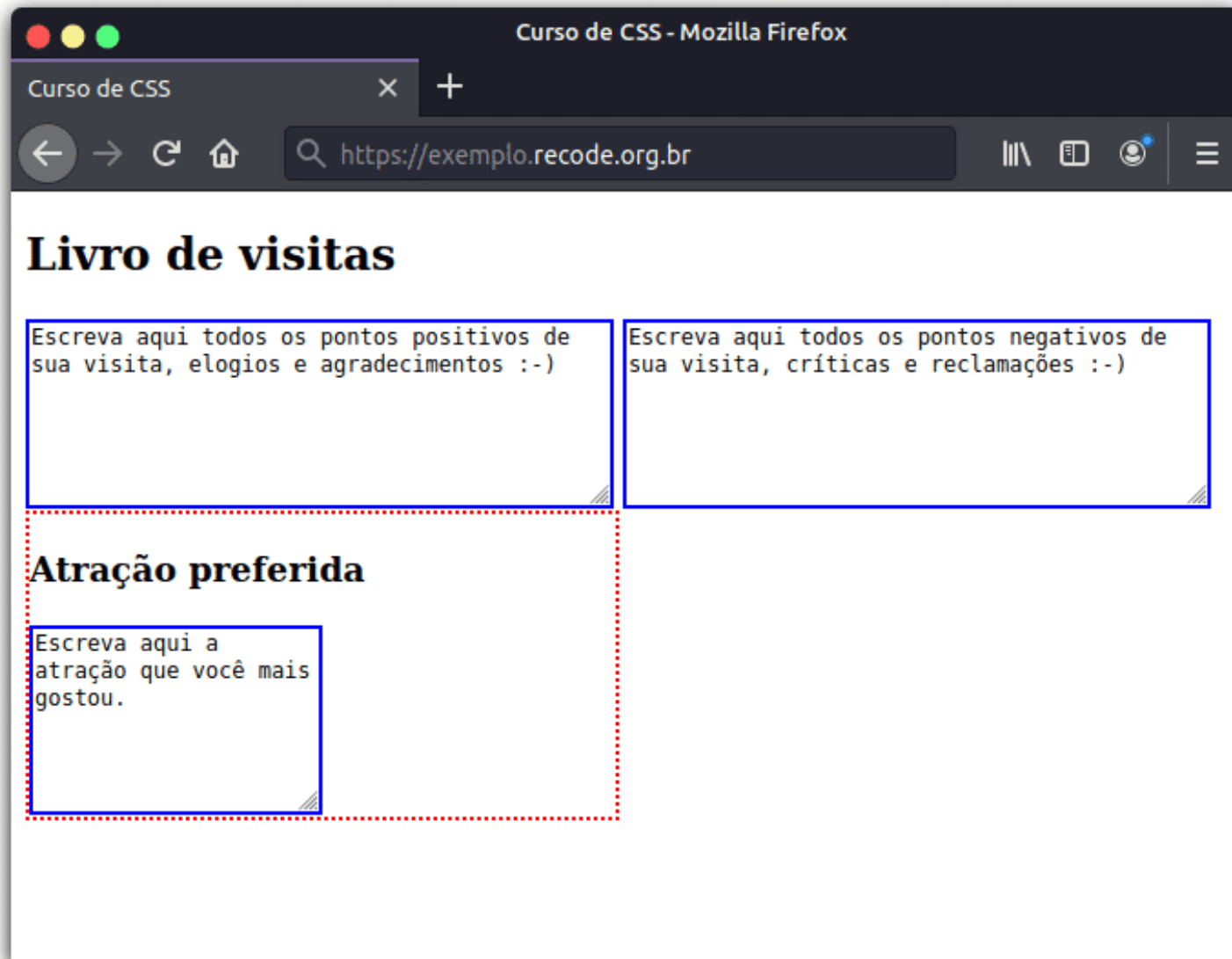
<!DOCTYPE html>
<html>
<head>
  <style>
    textarea {
      width: 50%;
      height: 100px;
      border: 2px solid blue;
    }
    .meio {
      width: 50%;
      border: 4px dotted red;
    }
  </style>
</head>
<body>

<h2>Livro de visitas</h2>
<form>
  <textarea>Escreva aqui todos os pontos positivos de sua visita, elogios e agradecimentos :-)</textarea>
  <textarea>Escreva aqui todos os pontos negativos de sua visita, críticas e reclamações :-/</textarea>
</form>

<form class="meio">
  <h3>Atração preferida</h3>
  <textarea>Escreva aqui a atração que você mais gostou.
</textarea>
</form>

</body>
</html>
```

CSS



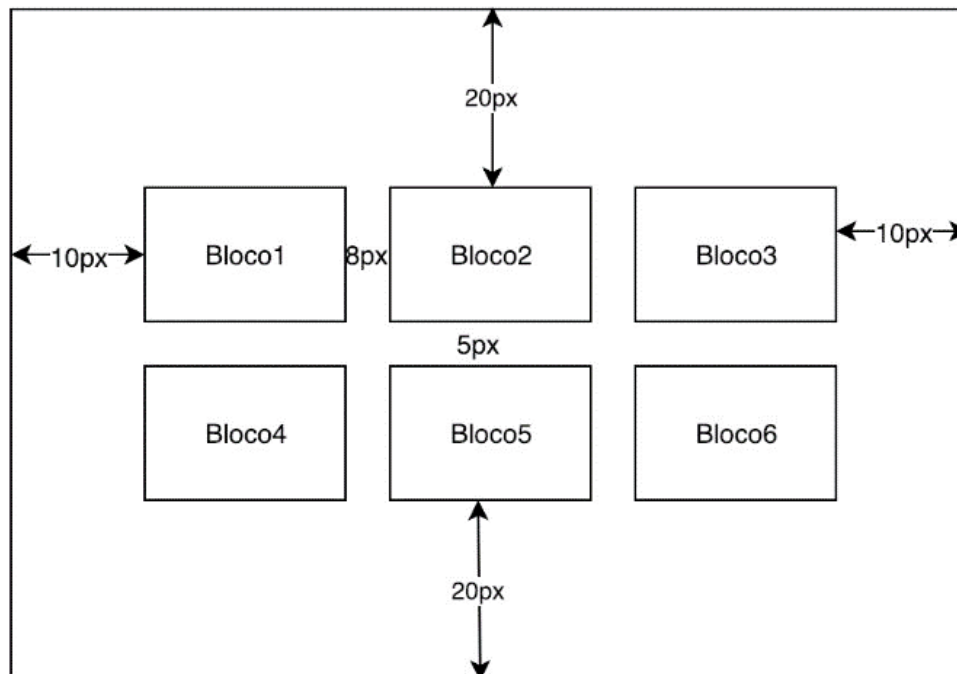
CSS

Trabalhando com Margin e padding

Colocando espaçamentos em seus elementos.

Existem dois tipos de espaçamentos:

- O que separa um bloco do outro, conhecido como margem; e
- O responsável por dar espaço entre o conteúdo de um bloco e as suas bordas, o espaçamento interno.



CSS

Espaçamento Interno com Padding

No espaçamento interno conseguimos incluir uma distância entre o conteúdo e as bordas de um bloco. Para definir esses espaçamentos, utilizamos a propriedade **padding**.

Ela, assim como as bordas, recebe valores numéricos em porcentagem e pixels, além de podermos incluir mais de uma medida (até 4, uma de cada lado) e trabalhar com cada lado separadamente, que são:

- top
- bottom
- right
- left

CSS

```
<!DOCTYPE html>
<html>
<head>
  <style>
    textarea {
      width: 50%;
      height: 100px;
      border: 2px solid blue;
    }
    .meio {
      width: 50%;
      border: 4px dotted red;
      padding: 10px;
    }
  </style>
</head>
<body>

<h2>Livro de visitas</h2>
<form>
  <textarea>Escreva aqui todos os pontos positivos de sua visita, elogios e agradecimentos :-)</textarea>
  <textarea>Escreva aqui todos os pontos negativos de sua visita, críticas e reclamações :-/</textarea>
</form>

<form class="meio">
  <h3>Atração preferida</h3>
  <textarea>Escreva aqui a atração que você mais gostou.</textarea>
</form>

</body>
</html>
```

Curso de CSS

https://exemplo.recodepro.org.br

Livro de visitas

Escreva aqui todos os pontos positivos de sua visita, elogios e agradecimentos :-)

Escreva aqui todos os pontos negativos de sua visita, críticas e reclamações :-)

Atração preferida

Escreva aqui a atração que você mais gostou.

CSS

Margens

Ela define a distância da borda até o início do objeto em questão. Não tem nenhum segredo, as regras aprendidas em bordas e padding são as mesmas para essa propriedade.

Para definir as margens, usamos a propriedade **margin**. Você vai notar que, mesmo sem definir **margin** e **padding** no seu elemento, o HTML já tem uma medida padrão definida. Então, não se preocupe, você pode “zerar”, ou seja, definir as suas próprias medidas.

CSS

A margem segue o **mesmo padrão de bordas e padding**, podendo receber valores numéricos em porcentagem e *pixels*, além de permitir incluir mais de uma medida (até 4, uma para cada lado) e trabalhar com cada lado separadamente. Mas, lembrando:

- top
- bottom
- right
- left

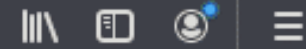
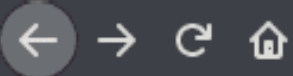
CSS

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      margin: 20%;
    }
    textarea {
      width: 100%;
      height: 100px;
      border: 2px solid blue;
    }
    .meio {
      width: 50%;
      border: 4px dotted red;
      padding: 10px;
    }
  </style>
</head>
<body>

<h2>Livro de visitas</h2>
<form>
  <textarea>Escreva aqui todos os pontos positivos de sua visita, elogios e agradecimentos :-)</textarea>
  <textarea>Escreva aqui todos os pontos negativos de sua visita, críticas e reclamações :-/</textarea>
</form>

<form class="meio">
  <h3>Atração preferida</h3>
  <textarea>Escreva aqui a atração que você mais gostou.</textarea>
</form>

</body>
</html>
```



Livro de visitas

Escreva aqui todos os pontos positivos de sua visita,
elogios e agradecimentos :-)

Escreva aqui todos os pontos negativos de sua visita,
críticas e reclamações :-)

Atração preferida

Escreva aqui a atração que
você mais gostou.

CSS

Display

A propriedade display permite que você defina a maneira como determinado elemento HTML deve ser exibido, ou não.

Por padrão, os elementos se organizem como blocos. Toda vez que adicionamos um elemento HTML (como h1, p, li, etc), eles se organizam um abaixo do outro.

Essa estrutura de blocos é como estes elementos são exibidos por padrão. Com o CSS, podemos adicionar e modificar todos os estilos existentes, inclusive este.

CSS

Display

Nos permite organizar os elementos com as seguintes opções:

block → A declaração **display: block** define o que já conhecemos, que é a sua opção padrão, ou seja, os elementos do tipo bloco são organizados um abaixo do outro.

inline → A declaração **display: inline** exibe blocos na mesma linha, fazendo com que ele se adeque ao tamanho do conteúdo que contém.

inline-block → A declaração **display: inline-block** mescla características dos valores inline e block.

none → A declaração **display: none** faz com que o elemento não seja renderizado.

CSS

```
<!DOCTYPE html>
<html>
<head>
  <style>
    .efeito_mesma_linha {
      display: inline;
    }
    .efeito_bloco {
      display: block;
    }
    .nao_visivel {
      display: none;
    }
  </style>
</head>

<body>
<h1>Estes</h1>
<p>elementos não possuem</p>
<p>nenhum tipo de</p>
<h2>estilo.</h2>
<hr>

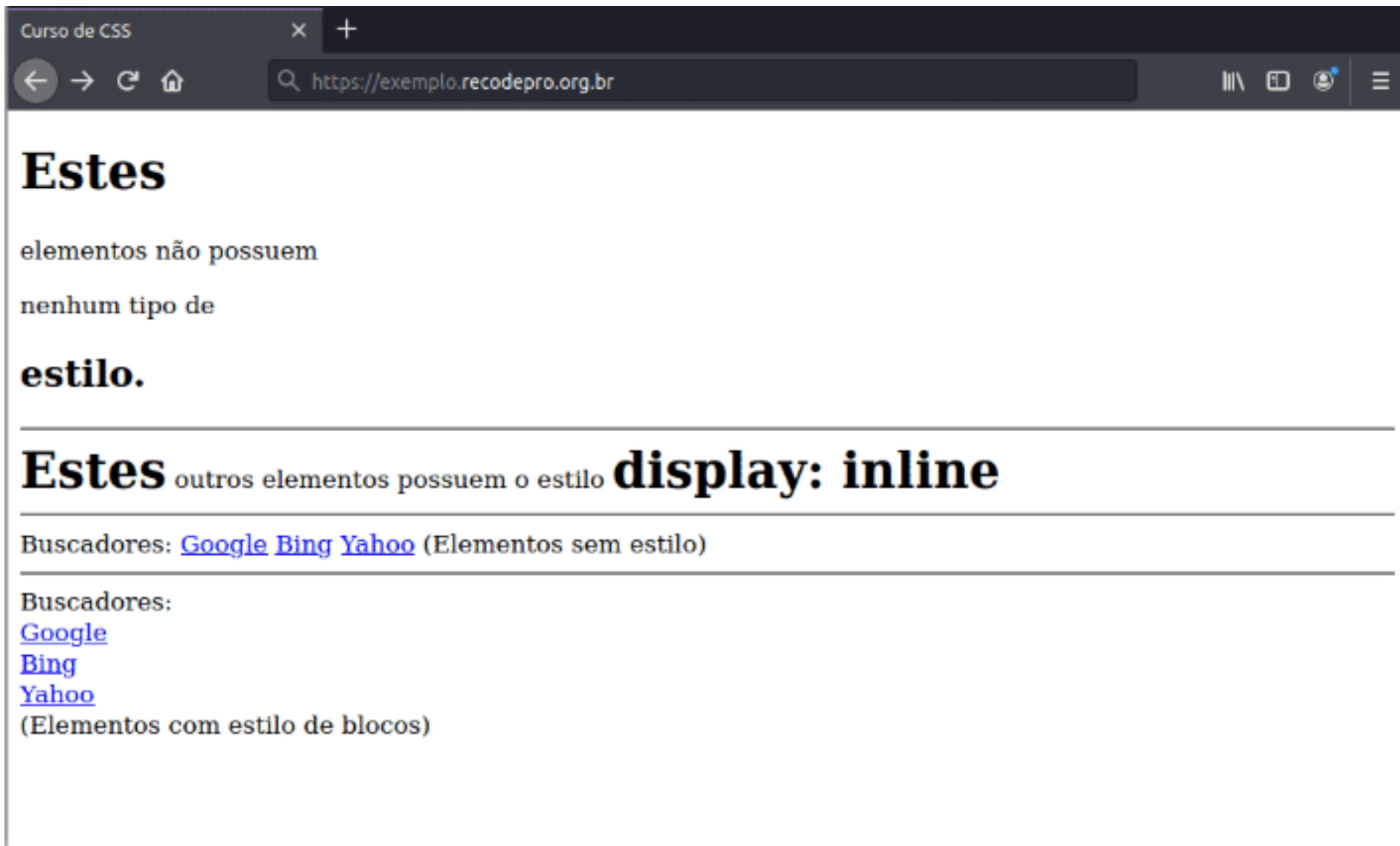
<h1 class="efeito_mesma_linha">Estes</h1>
<p class="efeito_mesma_linha">outros elementos</p>
<p class="efeito_mesma_linha">possuem o estilo</p>
<h1 class="efeito_mesma_linha">display: inline</h1>
<h1 class="nao_visivel">JÁ ESTE TEXTO GRANDE NÃO SERÁ EXIBIDO!!!</h1>
<hr>
```

CSS

```
<label>Buscadores:</label>
<a href="www.google.com">Google</a>
<a href="www.bing.com">Bing</a>
<a href="www.yahoo.com">Yahoo</a>
<label>(Elementos sem estilo)</label>
<hr>
```

```
<label class="efeito_bloco">Buscadores:</label>
<a class="efeito_bloco" href="www.google.com">Google</a>
<a class="efeito_bloco" href="www.bing.com">Bing</a>
<a class="efeito_bloco" href="www.yahoo.com">Yahoo</a>
<label class="efeito_bloco">(Elementos com estilo de blocos)</label>
</body>
</html>
```

CSS



CSS

Posição

Nos permite definir o posicionamento dos elementos.

Margin-Top (Margem Superior) → Refere-se a margem superior, o espaçamento entre o início da página e o início do bloco.

Margin-Bottom (Margem Inferior) → Refere-se a margem inferior, o espaçamento entre o fim do ultimo bloco e o fim da página.

Margin-Right (Margem Direita) → Refere-se ao espaçamento da margem direita, a distância entre o limite direito do bloco e a lateral direita da página.

Margin-Left (Margem Esquerda) → Refere-se ao espaçamento da margem esquerda, a distância entre o limite esquerdo do bloco e a lateral esquerda da página.

CSS

Vamos agora conhecer os tipos de posicionamento:

Relative → A declaração **position: relative** pode ser entendida através do seu nome, ou seja, é uma posição que possui relação com algo, e nesse caso, com o elemento pai. Esta é a opção padrão dos elementos. No módulo anterior aprendemos sobre margens (**margin**) que define a distância dos objetos das suas margens (superior, esquerda, inferior e direita) e que elas podem ser definidas separadamente (margin-top, margin-right, margin-bottom e margin-left).

CSS

Absolut → A declaração **position: absolut** é muito interessante, ela faz com que o elemento ignore a sua localização hierárquica (seus pais) e passa a fazer referência diretamente à página do navegador, como se estivesse flutuando sobre todos os demais elementos do seu documento HTML. Este estilo é muito utilizado e permite informações dinâmicas, um exemplo disso são as mensagens flutuantes que aparecem no site.

CSS

Fixed → A declaração **position: fixed** também é muito legal e funciona quase da mesma forma que o **absolut**: nesta opção o elemento fica fixo na página, onde quisermos que ele fique. Você pode rolar a página para cima e para baixo, mas ele fica sempre no mesmo local que foi definido, em relação à página. Essa posição é muitas vezes utilizada para menus.

CSS

Float

A propriedade **float** faz com que os elementos fiquem flutuando na ordem em que aparecem dentro da sua hierarquia, ou seja, dentro do elemento pai. Os elementos podem flutuar da esquerda para a direita ou da direita para a esquerda.

opções de valores para float:

Left → A declaração **float: left** faz com que o nosso elemento “flutue à esquerda”, logo ele sai do fluxo normal, onde os elementos blocos se posicionam um abaixo do outro e os coloca ao longo do lado direito. Esta propriedade tem um efeito semelhante (mas não igual) à **display: inline** (que encaixa os elementos na mesma linha, enquanto houver espaço).

CSS

Right → A declaração **float: right** faz com que o nosso elemento “flutue à direita”. Este efeito é bem interessante, pois funciona como se o fluxo fosse invertido, iniciando na margem direita em direção à margem esquerda, da mesma forma como os chineses e japoneses escrevem.

CSS

```
<!DOCTYPE html>
<html>
<head>
  <style>
    .grup01 {
      background: #cccccc;
      border: 1px solid blue;
      width: 100%;
      height: 200px;
    }

    .grupo2 {
      background: yellow;
      border: 1px solid blue;
      width: 100%;
      height: 200px;
    }

    .grup01 * {
      float: left;
    }

    .grupo2 * {
      float: right;
    }
  </style>
</head>
```

CSS

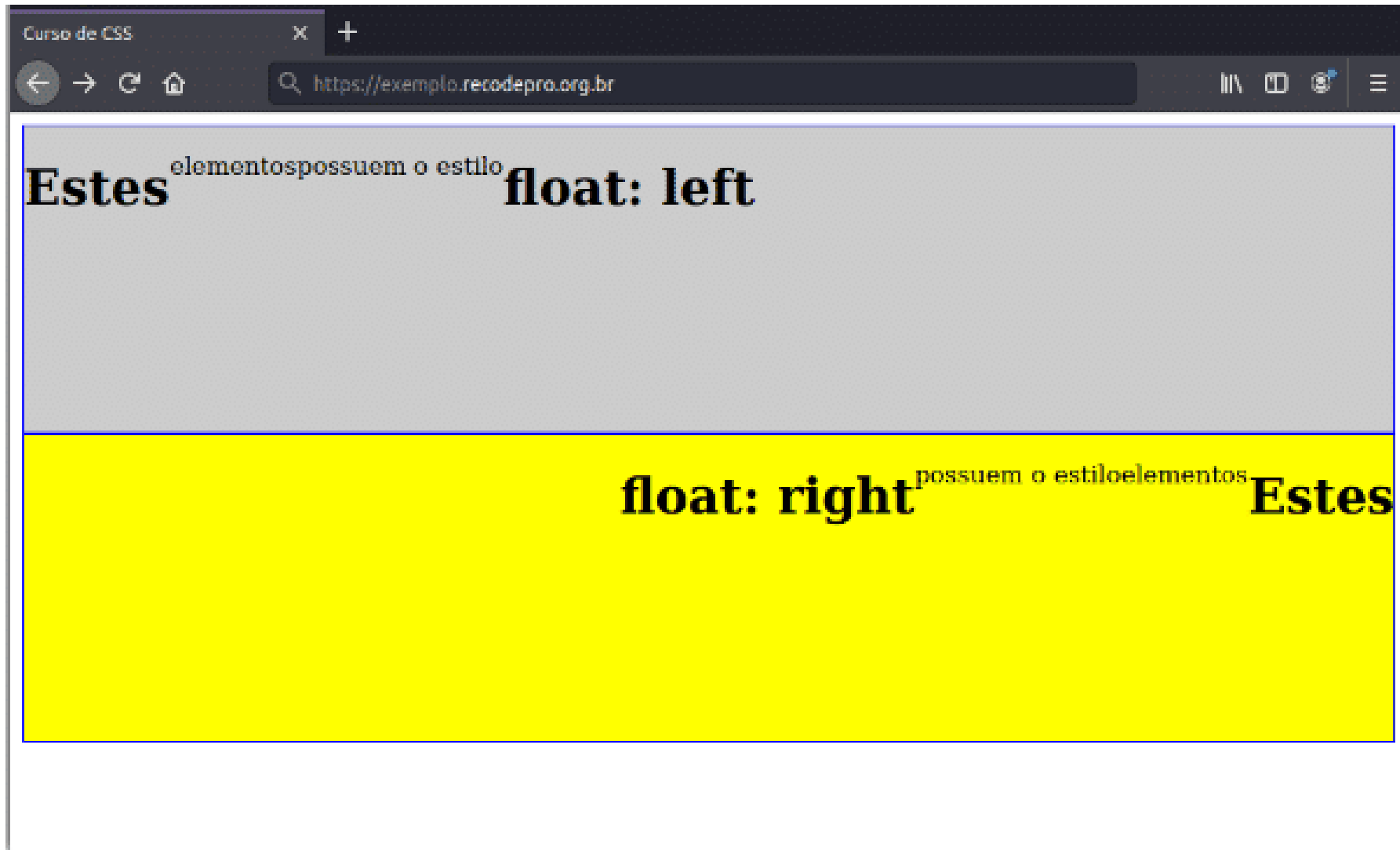
```
<body>

<div class="grupo1">
  <h1>Estes</h1>
  <p>elementos</p>
  <p>possuem o estilo</p>
  <h1>float: left</h1>
</div>

<div class="grupo2">
  <h1>Estes</h1>
  <p>elementos</p>
  <p>possuem o estilo</p>
  <h1>float: right</h1>
</div>

</body>
</html>
```

CSS



CSS

Div

A tag `<div>` define uma divisão ou seção em um documento HTML.

É usada como um contêiner para elementos HTML - que são estilizados com CSS ou manipulados com JavaScript.

É facilmente estilizada usando o atributo class ou id.

Qualquer tipo de conteúdo pode ser colocado dentro da `<div>` tag!

CSS

Exemplo:

```
<html>
<head>
  <style>
    .myDiv {
      border: 5px outset red;
      background-color: lightblue;
      text-align: center;
    }
  </style>
</head>
<body>
  <div class="myDiv">
    <h2>This is a heading in a div element</h2>
    <p>This is some text in a div element.</p>
  </div>
</body>
</html>
```

CSS – Material Complementar

Tabelas

Para especificar as bordas da tabela em CSS, use a propriedade **border**.

O exemplo abaixo especifica uma borda preta para os elementos `<table>`, `<th>` e `<td>`

```
table, th, td {border: 1px solid black;}
```

Primeiro nome	Último nome
Pedro	Griffin
Lois	Griffin

CSS

Recolher bordas da mesa

A propriedade border-collapse define se as bordas da tabela devem ser reduzidas em uma única borda.

```
table {border-collapse: collapse;}  
table, th, td {border: 1px solid black;}
```

Primeiro nome	Último nome
Pedro	Griffin
Lois	Griffin

CSS

Largura e altura da mesa

A largura e a altura de uma tabela são definidas pelas propriedades **width** e **height**.

O exemplo abaixo define a largura da tabela em 100% e a altura dos elementos <th> em 50px.

```
table {width: 100%;}  
th {height: 50px;}
```

Primeiro nome	Último nome	Poupança
Pedro	Griffin	\$ 100
Lois	Griffin	\$ 150
Joe	Swanson	\$ 300

CSS

Alinhamento horizontal

A propriedade **text-align** define o alinhamento horizontal (como esquerda, direita ou centro) do conteúdo em `<th>` ou `<td>`.

Por padrão, o conteúdo dos elementos `<th>` é alinhado ao centro e o conteúdo dos elementos `<td>` é alinhado à esquerda.

O exemplo a seguir alinha à esquerda o texto em elementos `<th>`.

th {text-align: left;}

Primeiro nome	Último nome	Poupança
Pedro	Griffin	\$ 100
Lois	Griffin	\$ 150
Joe	Swanson	\$ 300

CSS

Alinhamento vertical

A propriedade **vertical-align** define o alinhamento vertical (como superior, inferior ou meio) do conteúdo em <th> ou <td>.

Por padrão, o alinhamento vertical do conteúdo em uma tabela é médio (para os elementos <th> e <td>).

O exemplo a seguir define o alinhamento vertical do texto para baixo para elementos <td>.

td {height: 50px; vertical-align: bottom;}

Primeiro nome	Último nome	Poupança
Pedro	Griffin	\$ 100
Lois	Griffin	\$ 150
Joe	Swanson	\$ 300

CSS

Tabelas Listradas

Para tabelas listradas de zebra, use o **nth-child()** seletor e adicione um **background-color** a todas as linhas da tabela par (ou ímpar).

Exemplo:

```
tr:nth-child(even) {background-color: #f2f2f2;}
```

Primeiro nome	Último nome	Poupança
Pedro	Griffin	\$ 100
Lois	Griffin	\$ 150
Joe	Swanson	\$ 300

CSS

Cor da Tabela

Especifica através da propriedade `background-color` a cor de fundo e a cor do texto dos elementos `<th>`.

Exemplo:

```
th {background-color: #4CAF50; color: white;}
```

Primeiro nome	Último nome	Poupança
Pedro	Griffin	\$ 100
Lois	Griffin	\$ 150
Joe	Swanson	\$ 300

CSS

Tabela Responsiva

Uma tabela responsiva exibirá uma barra de rolagem horizontal se a tela for muito pequena para exibir o conteúdo completo.

Adicione um elemento de contêiner (como `<div>`) ao `overflow-x:auto` redor do elemento `<table>` para torná-lo responsivo:

Exemplo:

```
<div style="overflow-x:auto;">
  <table>
    ... table content ...
  </table>
</div>
```

CSS

Exemplo de uso:

```
<style>
    .vermelho{
        background-color: red;
    }
    .amarelo{
        background-color: yellow;
    }
    .azul{
        background-color: blue;
    }
</style>
<table border="1">
    <colgroup>
        <col class="vermelho">
        <col class="amarelo">
        <col class="azul">
    </colgroup>
    <tr>
        <th>Nome</th>
        <th>Idade</th>
        <th>Profissao</th>
    </tr>
```

CSS

```
<tr>  
  <td>Ted</td>  
  <td>22</td>  
  <td>Estudante</td>  
</tr>  
<tr>  
  <td>Junior</td>  
  <td>26</td>  
  <td>Escritor</td>  
</tr>  
</table>
```


CSS

tabelacol.html



Arquivo | C:/Projetos%20Acadêmicos/Instituições/Recode%20Pro/CSS/tabelacol.html



Apps



Google

WIX

Wix.com



O Que tem Pra Hoj...



Itaú Personalité



Santander



UVA

Nome	Idade	Profissao
Ted	22	Estudante
Junior	26	Escritor