

CSS



Para Iniciantes

QUEBRANDO AS REGRAS
QUEBRANDO AS REGRAS

Sumário

INTRODUÇÃO	1
SOFTWARE EDITOR DE CÓDIGOS	2
TIPO DE BORDAS	3
PROPIEDADE BORDER-RADIUS	6
• EXEMPLO-1	6
• EXEMPLO-2	8
• EXEMPLO-3	10
• EXEMPLO-4	12
• EXEMPLO-5	14
MÓDULO CSS3 PARA CORES	16
• DECLARAÇÃO COM RGBA	16
• DECLARAÇÃO COM GRADIENTE	18
• DECLARAÇÃO COM GRADIENTE LINEAR	20
• DECLARAÇÃO COM GRADIENTE RADIAL	26
ATRIBUTO BOX-SHADOW	28
• COMPATIBILIDADE ENTRE OS NAVEGADORES	29
• EXEMPLO-1	30
• EXEMPLO-2	32
• EXEMPLO-3	34
ATRIBUTO TEXT-SHADOW	36
• EXEMPLO-1	36
• EXEMPLO-2	38
• EXEMPLO-3	40
CSS3 ANIMATIONS	42
• EXEMPLO-1	43
• EXEMPLO-2	45
• EXEMPLO-3	47

Sumário

GALERIA DE IMAGENS CSS3	49
BOTÕES CSS3	52
CENTRALIZANDO ELEMENTOS NA PÁGINA	54
ATRIBUTO BOX-SIZING	58
ANIMAÇÃO CSS3 – PROJETO FINAL	63
AGRADECIMENTOS	64

Introdução:

O que é CSS e qual sua importância?

CSS é acrônimo de **Cascading Style Sheets** que em português seria algo como “folhas de estilo em cascata”. É uma política que define como os elementos que compõem uma página, um documento ou um aplicativo Web. Quando falamos de acessibilidade, performance e manutenção, tem como objetivo fazer a versão do site, da interatividade e da apresentação de um site ou aplicativo web.

O Módulo da **CSS3** denominado **CSS3 Color Module - Level 3** destina-se a especificar as diferentes maneiras de se atribuir valores para as propriedades CSS que admitem cores tais como background-color e color. A linguagem utiliza propriedades e valores para cores de textos, backgrounds, bordas e outras partes dos elementos em um documento. Esta especificação descreve os valores de cores e suas propriedades bem como opacidade de cores.

O **CSS3** chegou causando um alvoroço muito grande no mundo do web design, pois ele vai muito além de alterar cores, fontes e espaçamentos. **CSS3** é a terceira mais nova versão das famosas **Cascading Style Sheets (ou simplesmente CSS)**, onde se define estilos para seu projeto web. Com efeitos de transição, imagem, imagem de fundo/Background e outros, que dão um estilo novo e elegante a seus projetos web. Ou em todos os aspectos de design do layout da página.

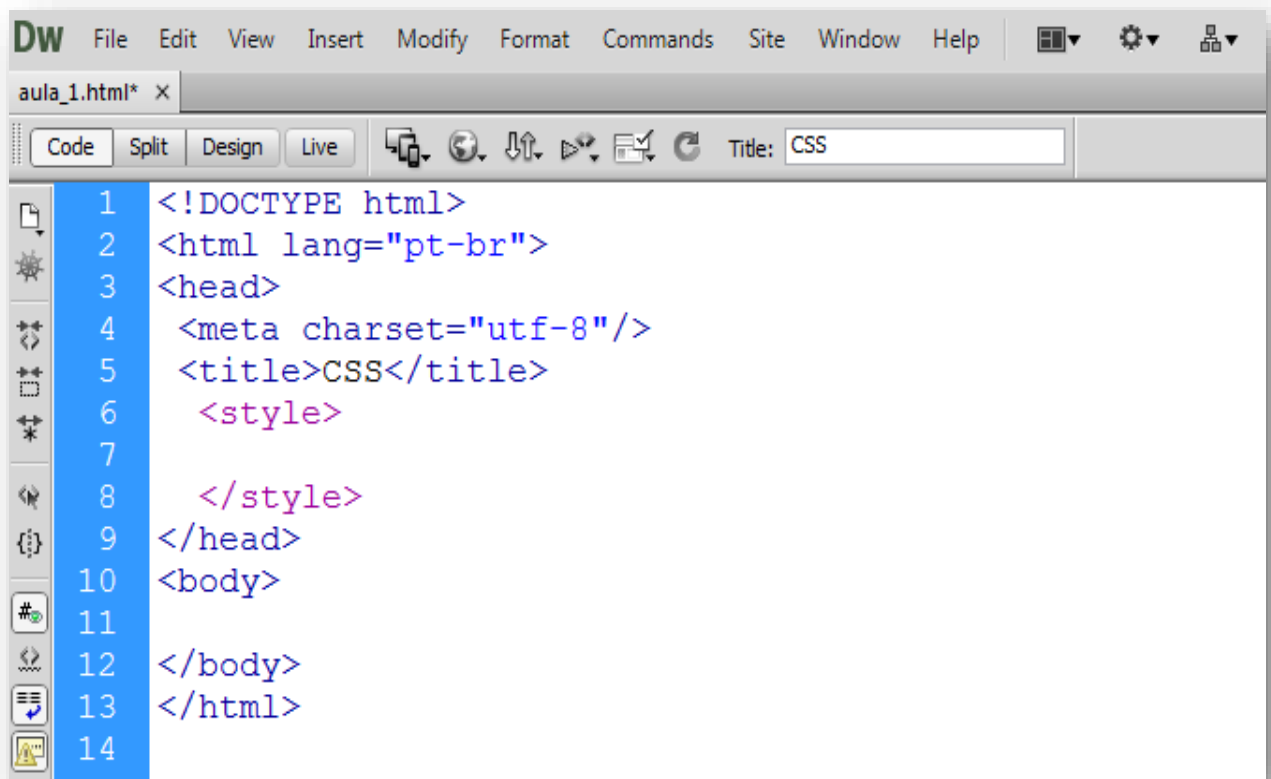
A principal função do **CSS3** é abolir as imagens de plano/Background de fundo, bordas arredondadas, apresentar transições e efeitos para criar animações de vários tipos, como um simples relógio de ponteiros. Isso se deve aos novos **Browsers/Navegadores**, que chegaram, com suporte à essa linguagem, como o **Google Chrome, Opera, Internet Explorer 9, Safari e Mozilla Firefox**. Uma das grandes **vantagens do CSS3** é a possibilidade de personalização praticamente inacabável, com diversas possibilidades de estilização em uma página HTML. Dessa forma, a produtividade é significativamente maior, e os erros são reduzidos.



Software editor de códigos

Nesta etapa do curso vamos fazer uso do software **Dreamweaver** da **Adobe**. Você pode utilizar o software de sua preferência, pois existem vários editores de códigos, como o Sublime Text, Not Pad++ que são gratuitos, até os pagos, como o PhpStorm –JetBrains, dentre outros.

Agora vamos ver na prática o que é possível fazer com a poderosa **versão CSS3**. Como já se sabe o que é uma folha de estilo, vamos digitar a estrutura padrão no documento HTML, para assim inserir o arquivo **CSS** no documento, pois iremos utilizar o **CSS inline**. Siga a descrição abaixo:

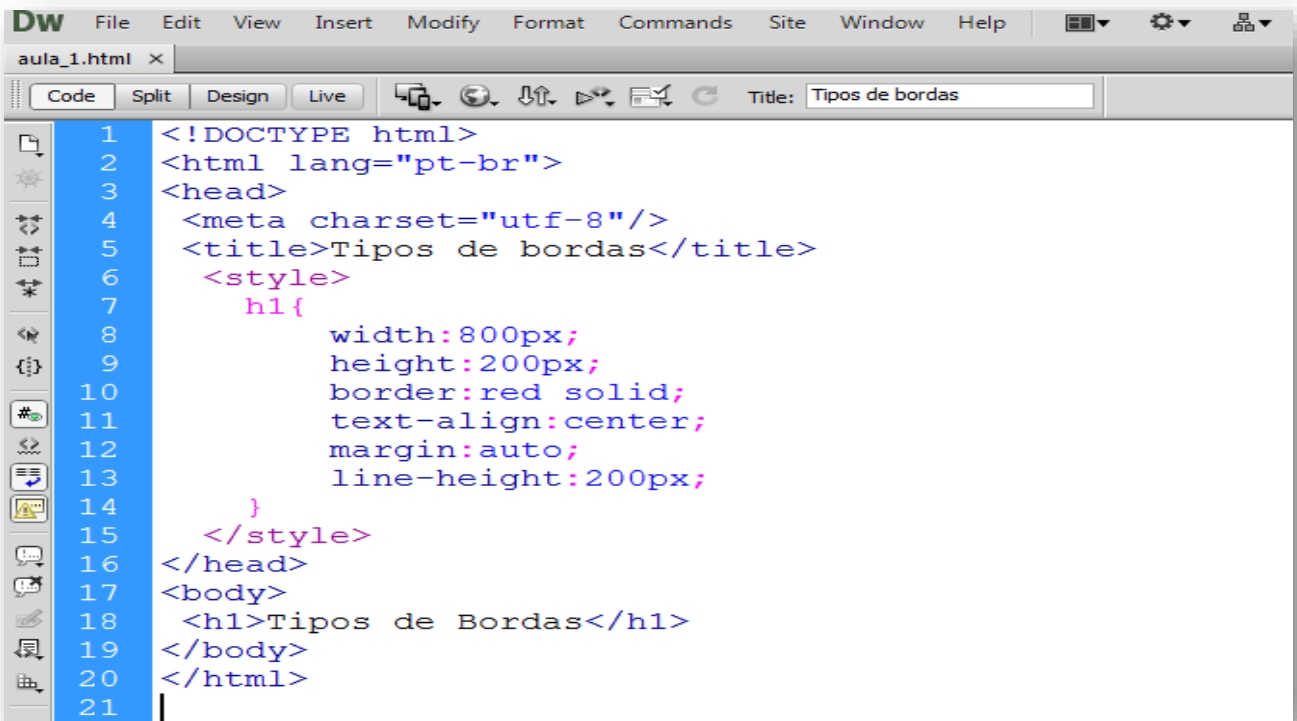


```
Dw File Edit View Insert Modify Format Commands Site Window Help
aula_1.html* x
Code Split Design Live Title: CSS
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4 <meta charset="utf-8"/>
5 <title>CSS</title>
6 <style>
7
8 </style>
9 </head>
10 <body>
11
12 </body>
13 </html>
14
```

Veja como facilita a visualização dos códigos digitados em um editor de código software. Realmente é um mão na roda. Lembrando que para aperfeiçoar suas habilidades e familiarizar cada vez mais com a linguagem, o uso do Bloco de Notas é essencial.

Tipo de bordas

Muito bom, agora vamos começar conhecendo os tipos de bordas. Vamos fazer um retângulo no body e depois estiliza-lo. Faça conforme a descrição abaixo:



```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Tipos de bordas</title>
6   <style>
7     h1{
8       width:800px;
9       height:200px;
10      border:red solid;
11      text-align:center;
12      margin:auto;
13      line-height:200px;
14    }
15  </style>
16 </head>
17 <body>
18   <h1>Tipos de Bordas</h1>
19 </body>
20 </html>
21
```

Salvando e abrindo no navegador vai ficar assim:



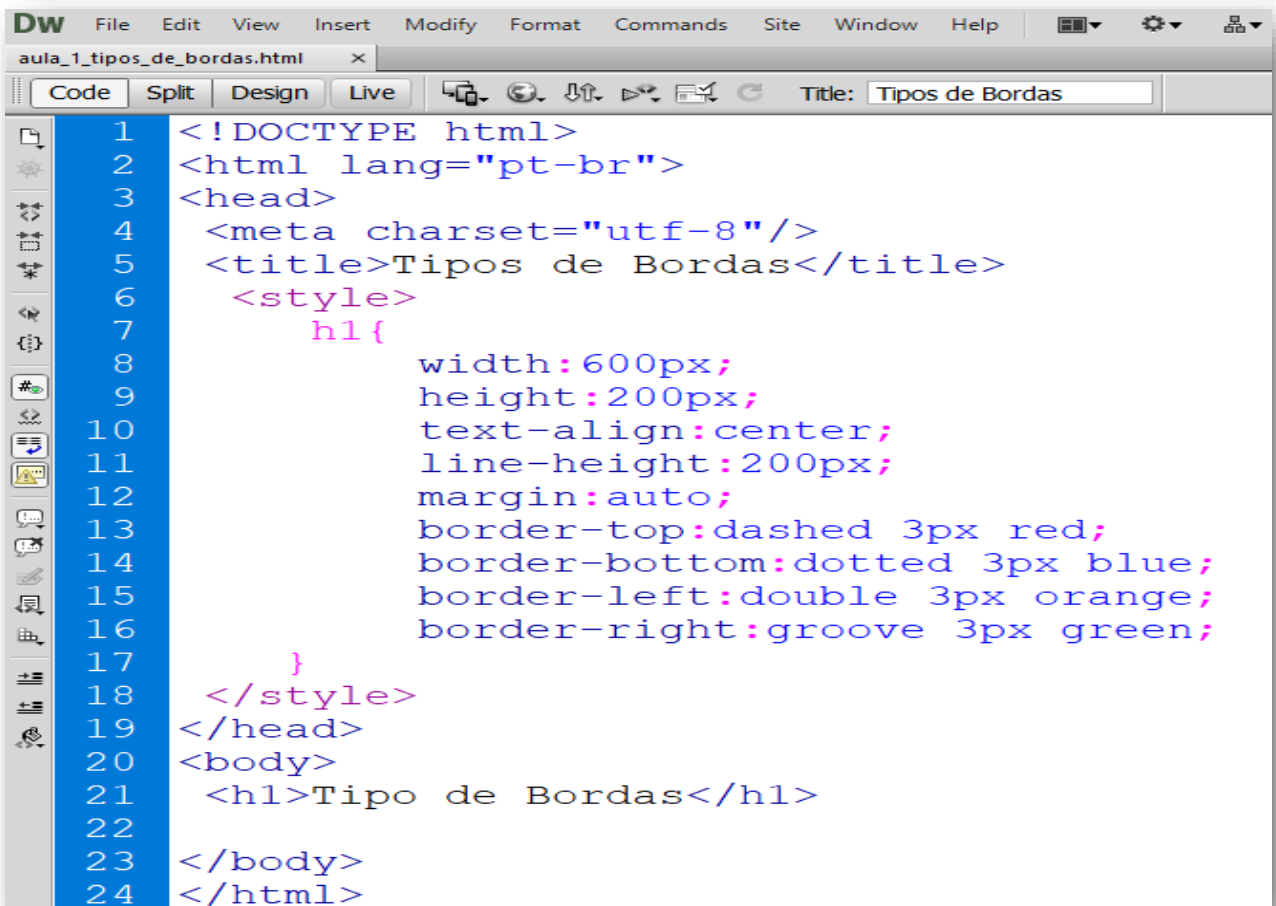
Vamos entender o que acabamos de fazer:

Digitamos no documento HTML **Tipos de Bordas** e inserimos o atributo **h1** pra título principal.

Depois estilizamos o atributo **h1** no arquivo **CSS** :

- 800 pixels de largura.
- 200 pixels de altura.
- Centralizado na página, juntamente com seu conteúdo.
- Com borda sólida, na cor vermelha e com espessura de 1 pixel.

As propriedades das bordas permitem que você controle o estilo, a cor e a espessura das bordas de um elemento **HTML**. As propriedades podem ser declaradas para cada uma das quatro bordas individualmente ou em uma declaração abreviada. Vamos ver como isso se comporta na prática: Faça conforme a descrição abaixo:



```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Tipos de Bordas</title>
6   <style>
7     h1{
8       width:600px;
9       height:200px;
10      text-align:center;
11      line-height:200px;
12      margin:auto;
13      border-top:dashed 3px red;
14      border-bottom:dotted 3px blue;
15      border-left:double 3px orange;
16      border-right:groove 3px green;
17    }
18  </style>
19 </head>
20 <body>
21   <h1>Tipo de Bordas</h1>
22
23 </body>
24 </html>
```

Salvando e abrindo no navegador vai ficar assim:

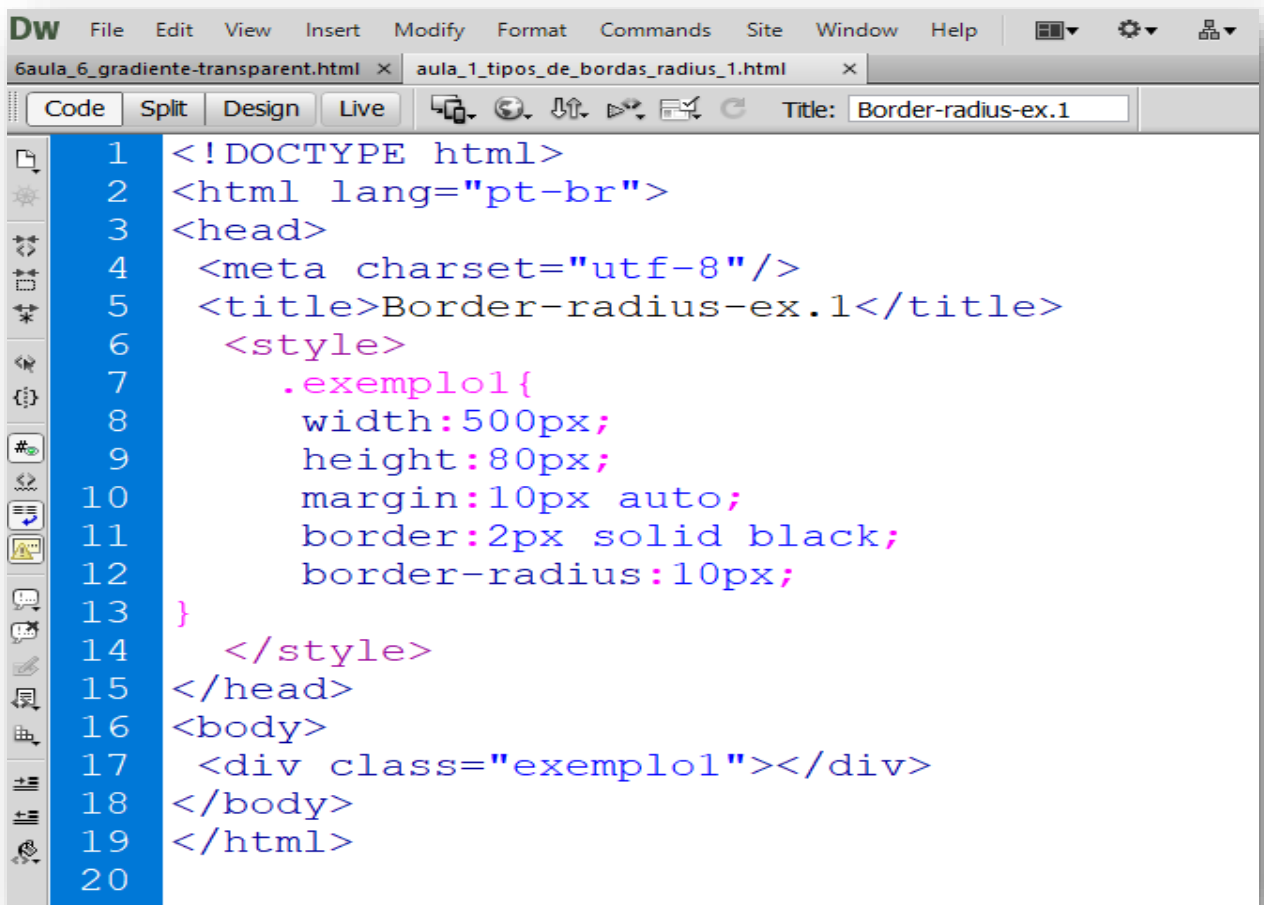


Propriedade Border-Radius

A propriedade **CSS border-radius** permite definir como bordas arredondadas são. A curva de cada esquina é definida usando um ou dois raios, definindo sua forma: círculo ou elipse.

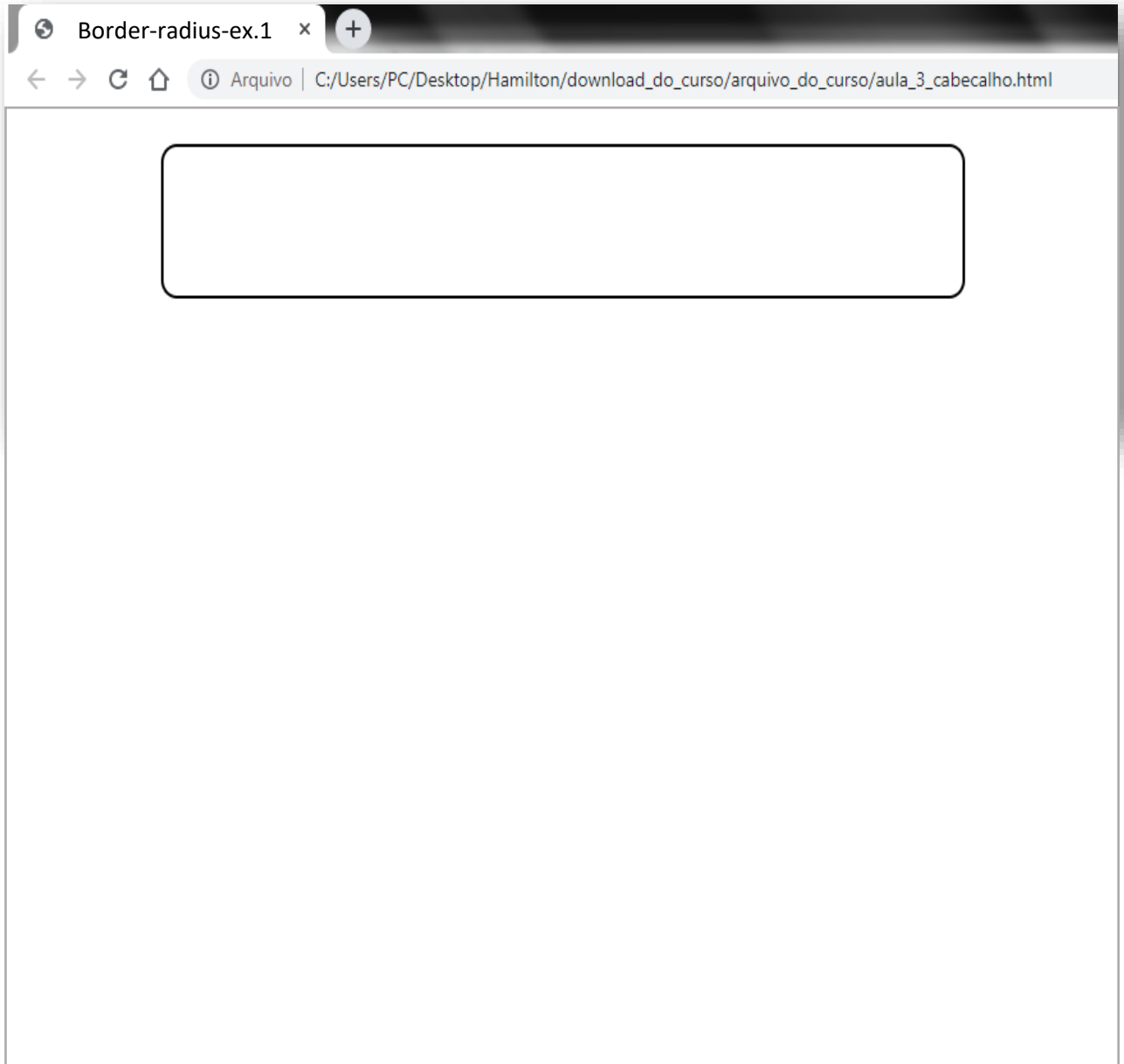
veja alguns exemplos:

- Exemplo 1

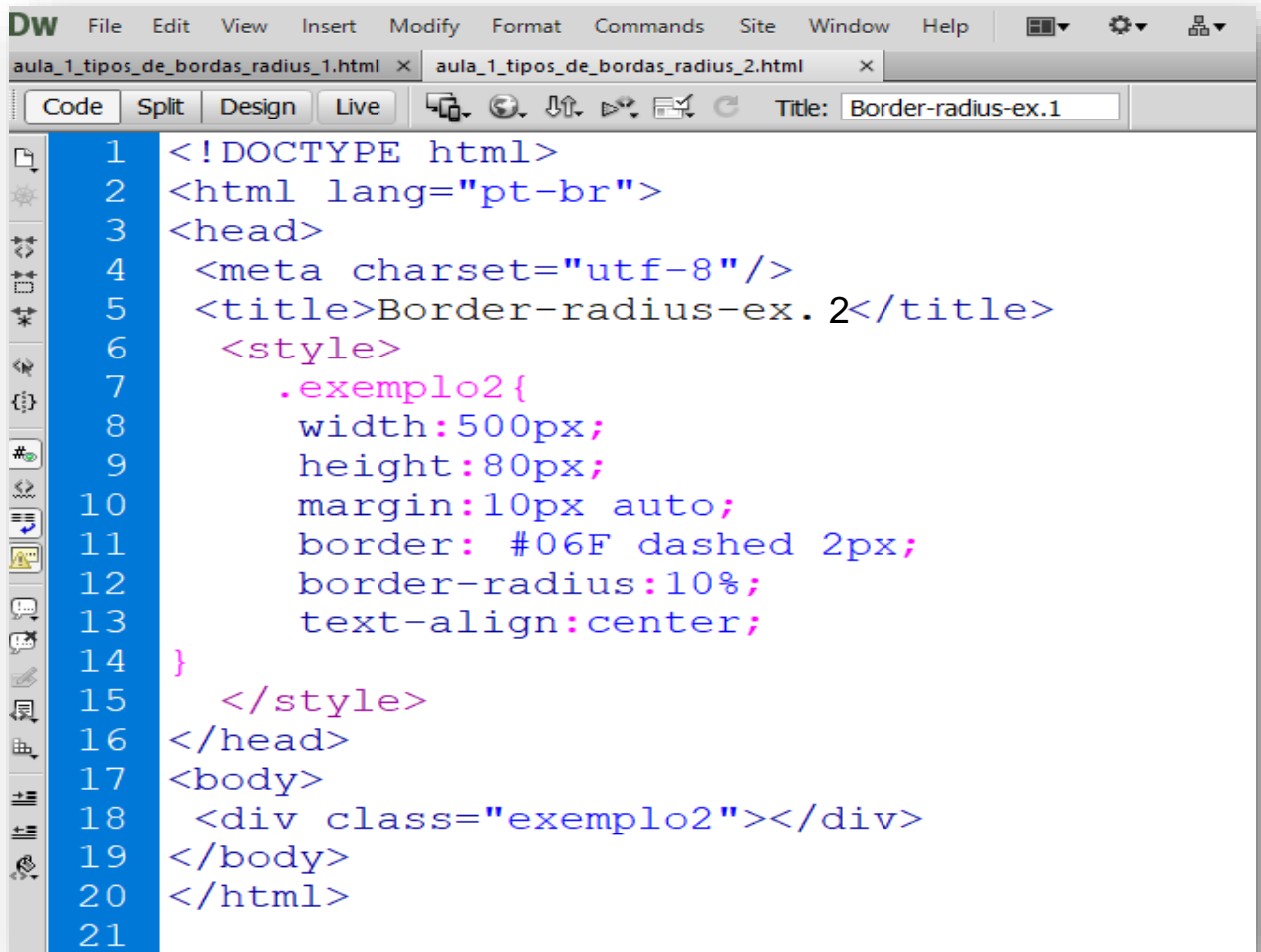


```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Border-radius-ex.1</title>
6   <style>
7     .exemplo1{
8       width:500px;
9       height:80px;
10      margin:10px auto;
11      border:2px solid black;
12      border-radius:10px;
13    }
14  </style>
15 </head>
16 <body>
17   <div class="exemplo1"></div>
18 </body>
19 </html>
20
```

Salvando e abrindo no navegador vai ficar assim:

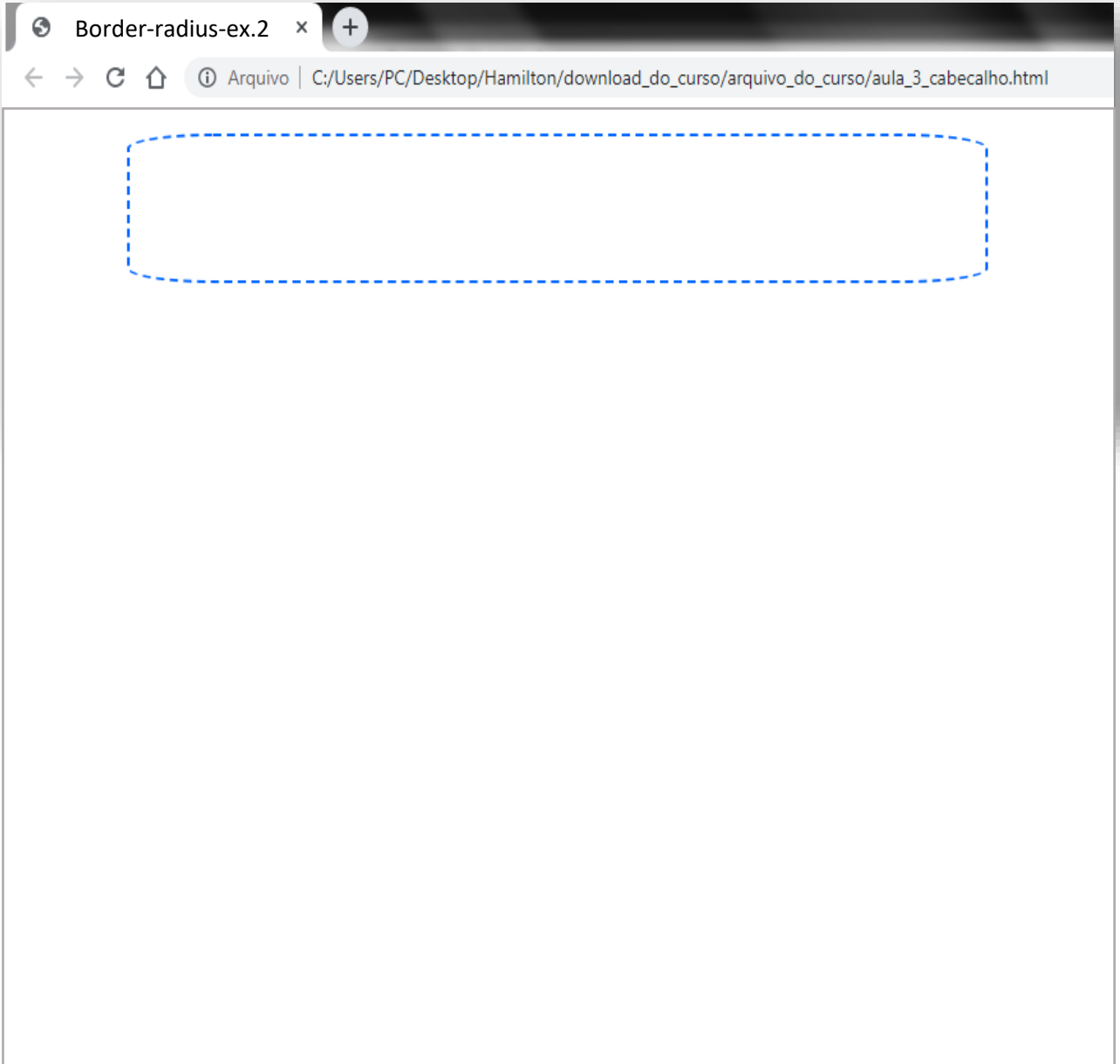


- Exemplo 2

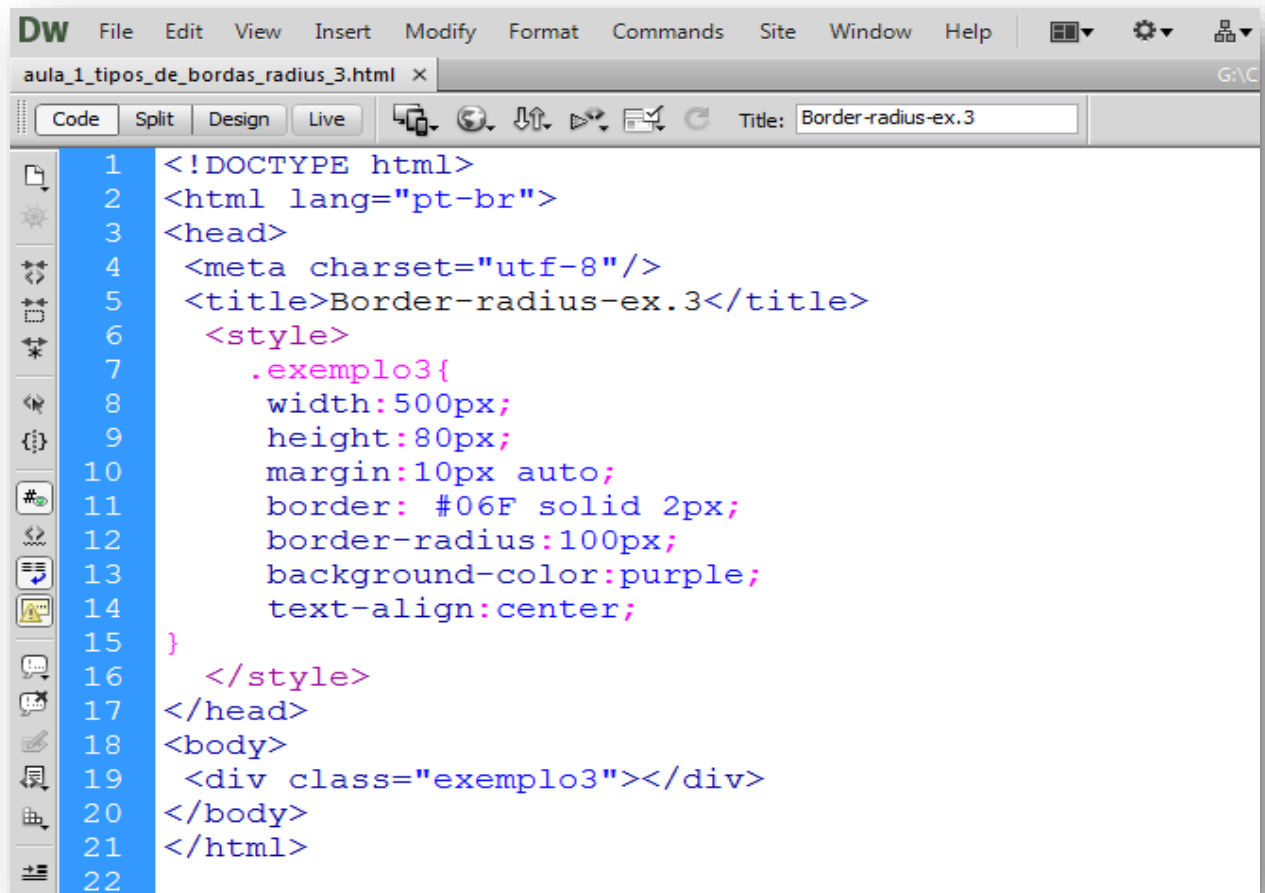


```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Border-radius-ex. 2</title>
6   <style>
7     .exemplo2{
8       width:500px;
9       height:80px;
10      margin:10px auto;
11      border: #06F dashed 2px;
12      border-radius:10%;
13      text-align:center;
14    }
15  </style>
16 </head>
17 <body>
18   <div class="exemplo2"></div>
19 </body>
20 </html>
21
```

Salvando e abrindo no navegador vai ficar assim:

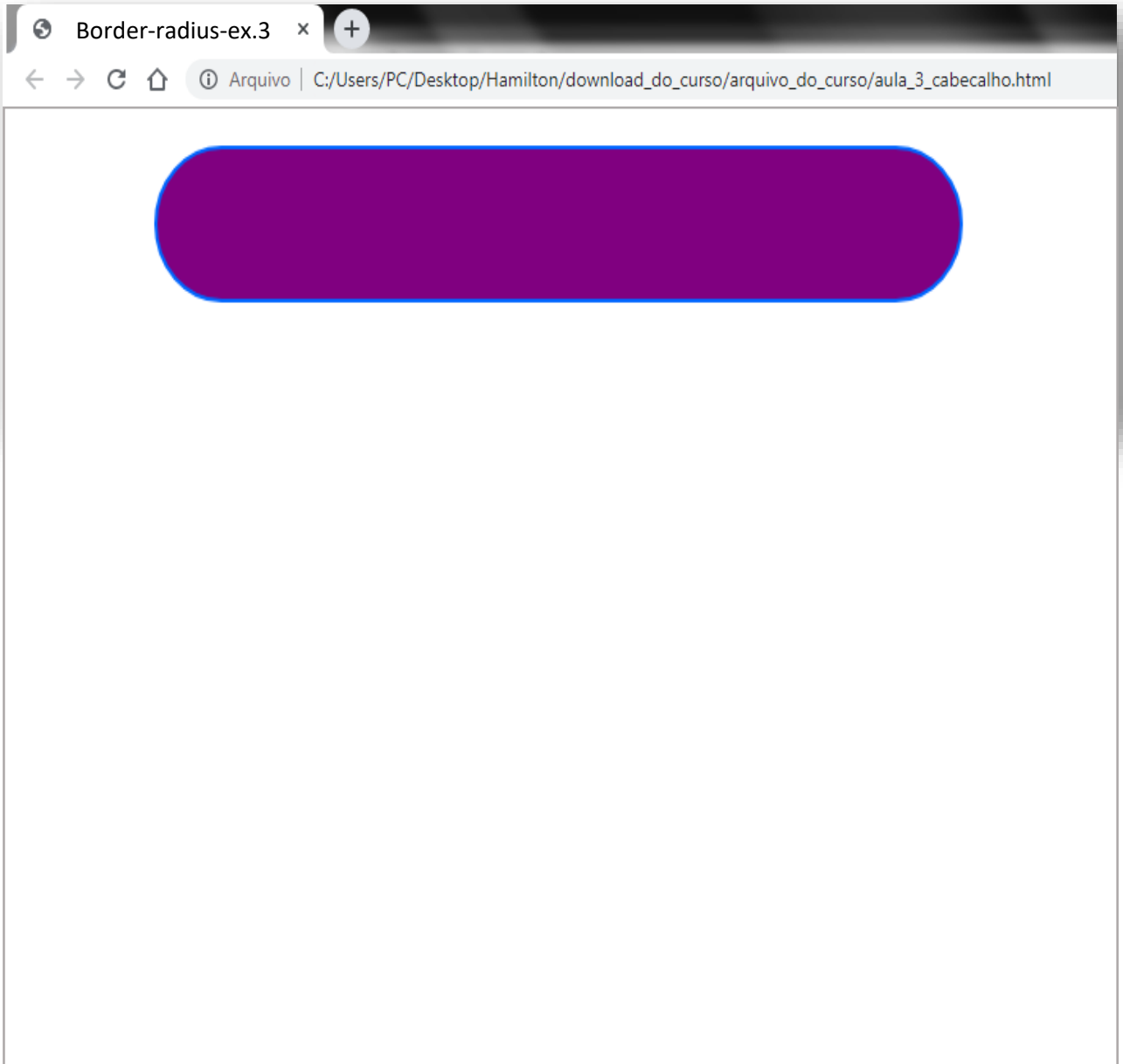


- Exemplo 3

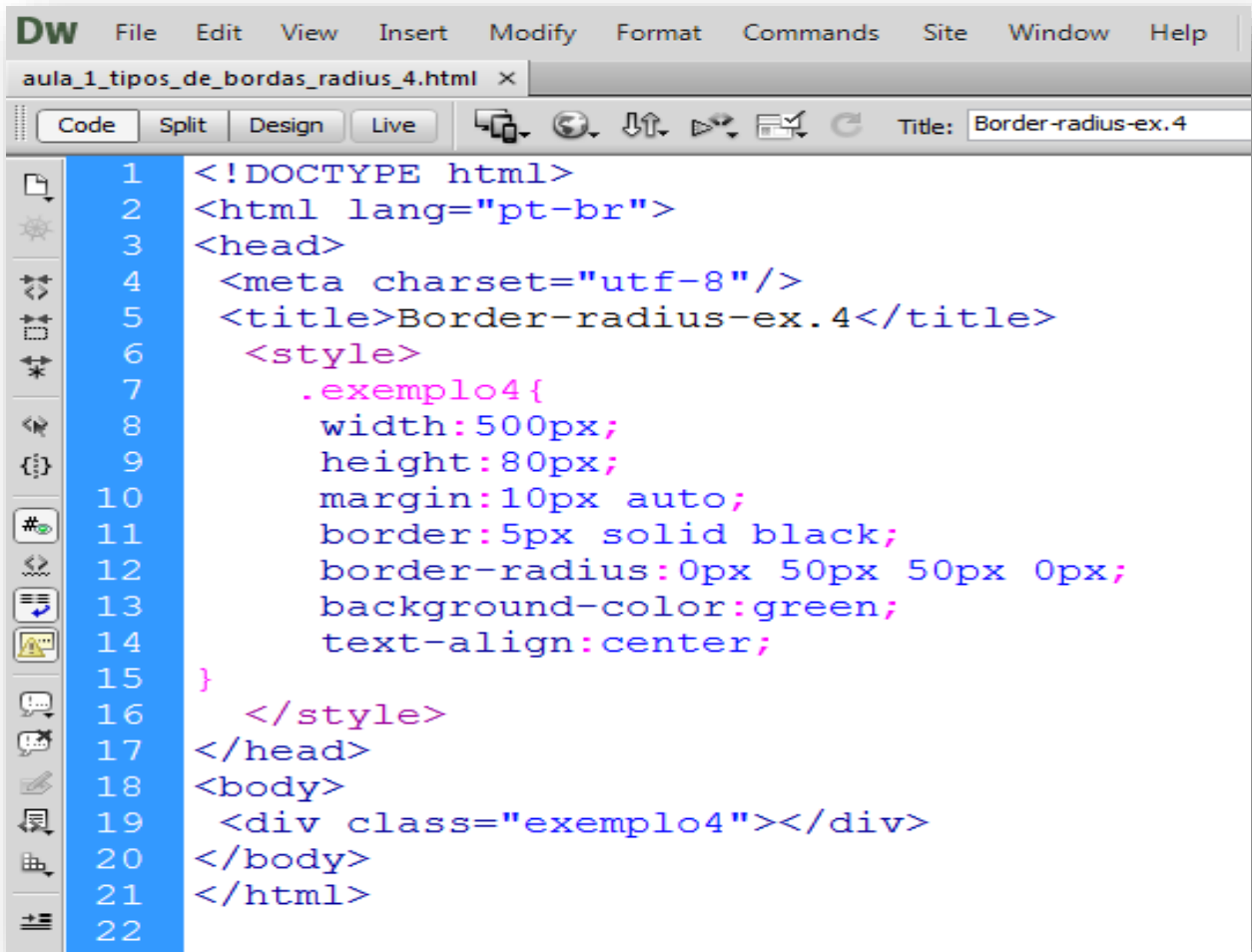


```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Border-radius-ex.3</title>
6   <style>
7     .exemplo3{
8       width:500px;
9       height:80px;
10      margin:10px auto;
11      border: #06F solid 2px;
12      border-radius:100px;
13      background-color:purple;
14      text-align:center;
15    }
16  </style>
17 </head>
18 <body>
19   <div class="exemplo3"></div>
20 </body>
21 </html>
22
```

Salvando e abrindo no navegador vai ficar assim:

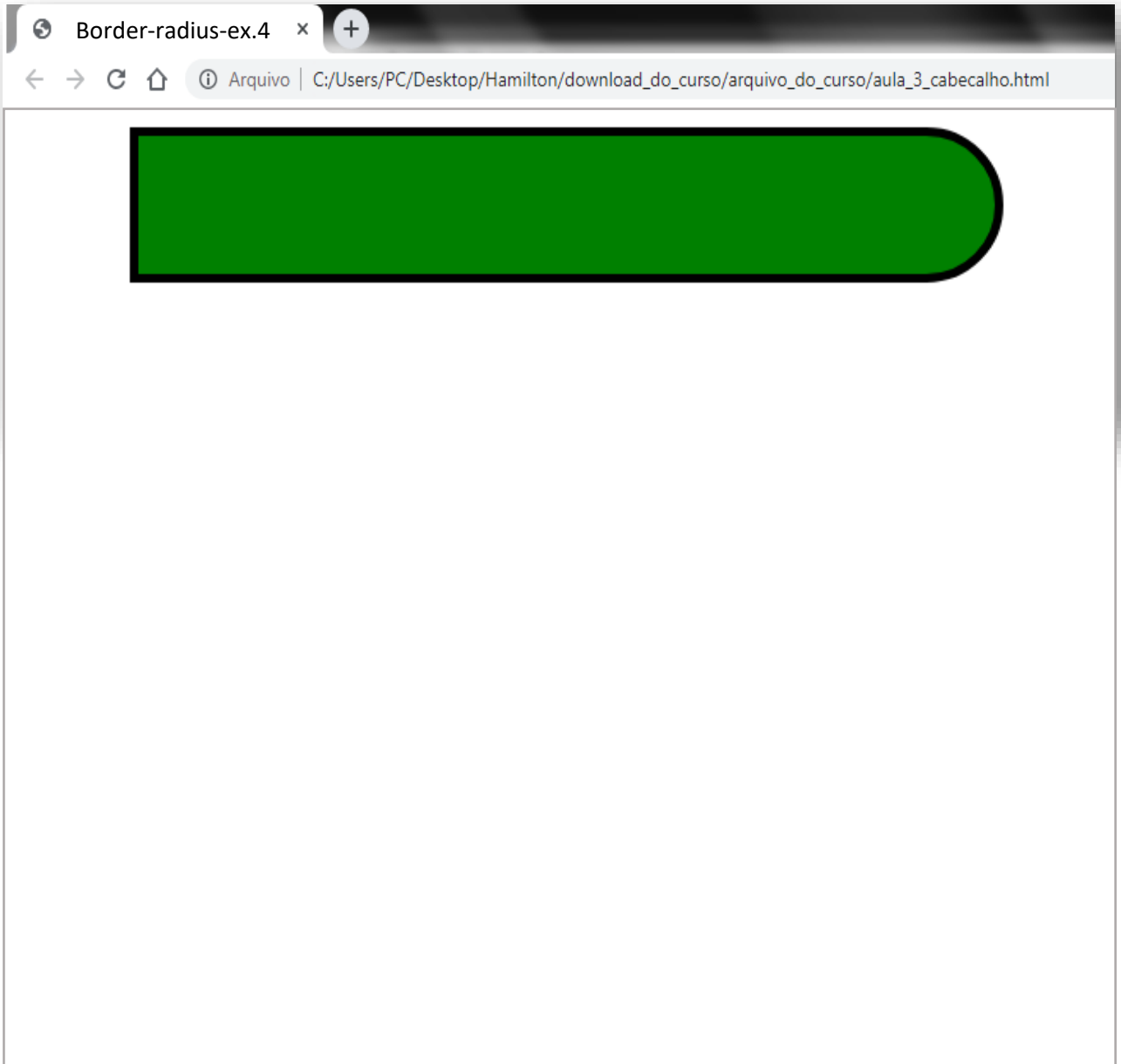


- Exemplo 4

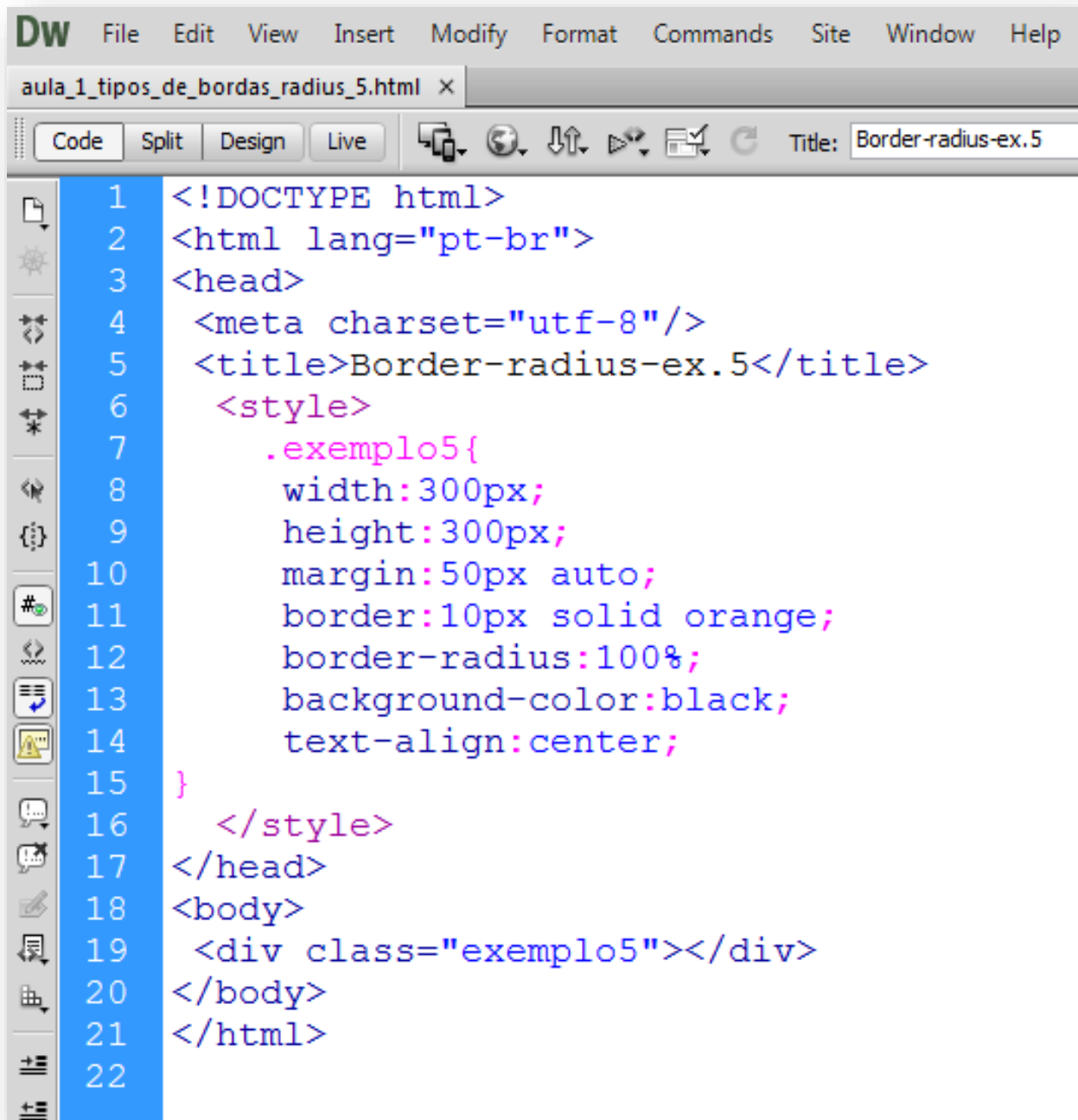


```
DW File Edit View Insert Modify Format Commands Site Window Help
aula_1_tipos_de_bordas_radius_4.html x
Code Split Design Live
Title: Border-radius-ex.4
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Border-radius-ex.4</title>
6   <style>
7     .exemplo4{
8       width:500px;
9       height:80px;
10      margin:10px auto;
11      border:5px solid black;
12      border-radius:0px 50px 50px 0px;
13      background-color:green;
14      text-align:center;
15    }
16  </style>
17 </head>
18 <body>
19   <div class="exemplo4"></div>
20 </body>
21 </html>
22
```

Salvando e abrindo no navegador vai ficar assim:

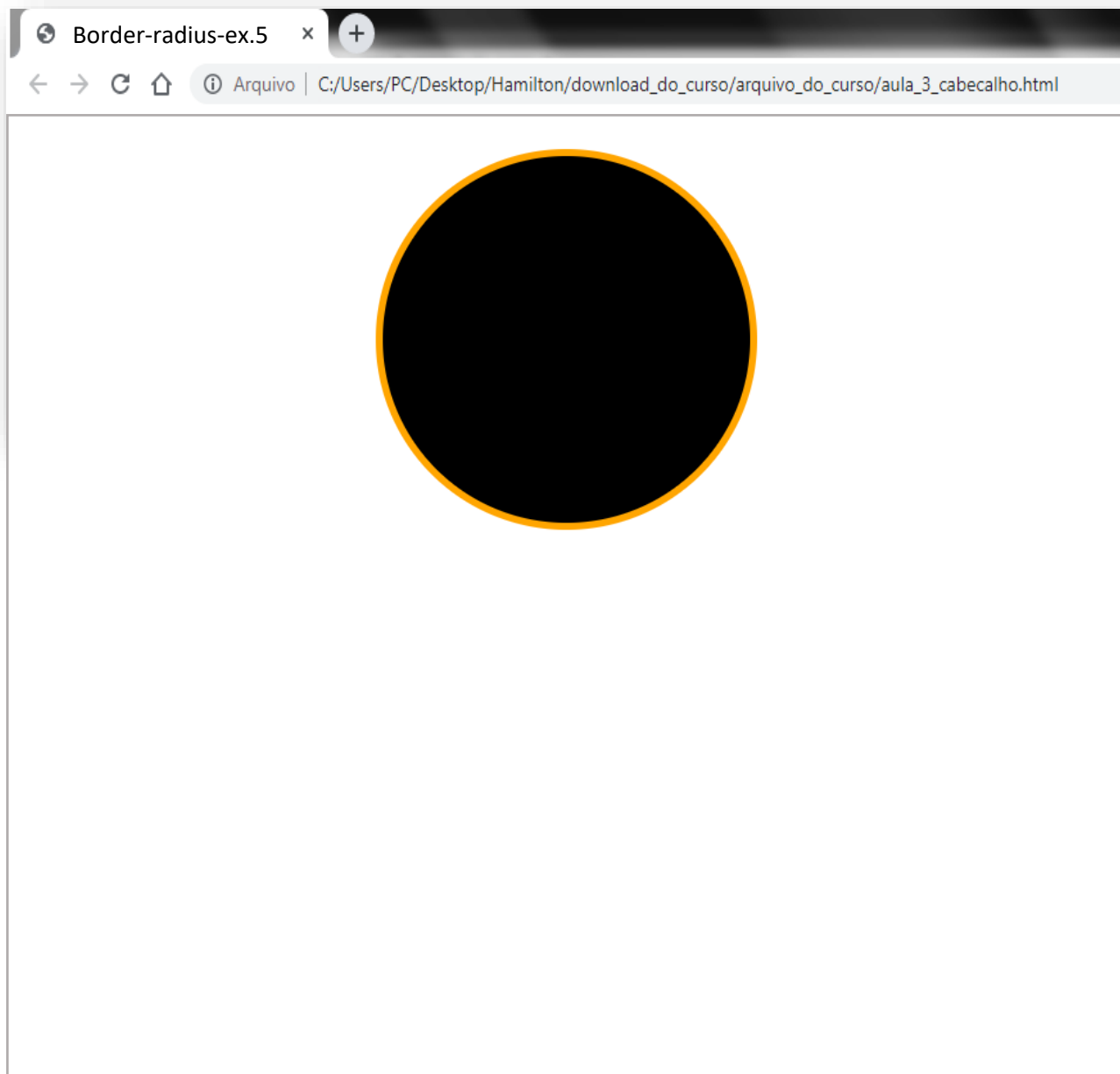


- Exemplo 5



```
DW File Edit View Insert Modify Format Commands Site Window Help
aula_1_tipos_de_bordas_radius_5.html x
Code Split Design Live
Title: Border-radius-ex.5
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Border-radius-ex.5</title>
6   <style>
7     .exemplo5{
8       width:300px;
9       height:300px;
10      margin:50px auto;
11      border:10px solid orange;
12      border-radius:100%;
13      background-color:black;
14      text-align:center;
15    }
16  </style>
17 </head>
18 <body>
19   <div class="exemplo5"></div>
20 </body>
21 </html>
22
```

Salvando e abrindo no navegador vai ficar assim:



Módulo CSS3 para cores

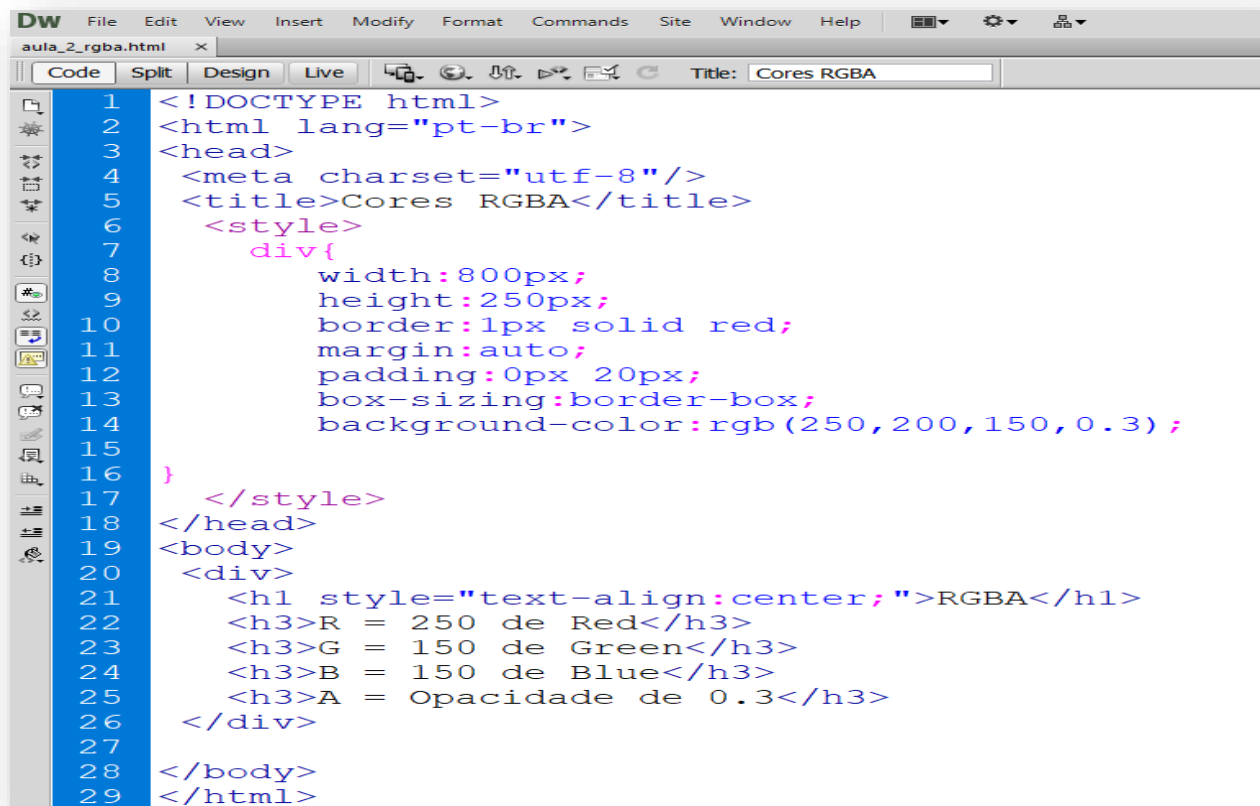
• Declaração com RGBA

A declaração de cores com uso de **RGB (red, green, blue)** está prevista na atual versão da **CSS**. O Módulo **CSS3** para Cores estendeu o uso de **RGB** criando **RGBA (red, green, blue, alpha-opacity)** acrescentando mais um argumento na declaração da cor, que permite definir a opacidade em uma escala decimal de 0 a 1.

Os valores **RGB** podem ser declarados em escala numérica de 0 a 255 ou percentual de 0 a 100%, conforme explicado no curso citado anteriormente.

Os campos **RGB (red, green, blue)** admitem valores numéricos de 0 a 255. Não contempla porcentagens embora sejam válidas. O campo **A (opacidade alfa)** admite decimais de 0 a 1 sendo 0 transparente e 1 opaco. Valores decimais devem ser escritos com uso de ponto. Exemplo: 0.5.

Vamos ver isso na prática: Vamos declarar a cor de fundo do retângulo usando o atributo **RGBA**. Faça conforme a descrição abaixo:



```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Cores RGBA</title>
6   <style>
7     div{
8       width:800px;
9       height:250px;
10      border:1px solid red;
11      margin:auto;
12      padding:0px 20px;
13      box-sizing:border-box;
14      background-color:rgb(250,200,150,0.3);
15    }
16  </style>
17 </head>
18 <body>
19   <div>
20     <h1 style="text-align:center;">RGBA</h1>
21     <h3>R = 250 de Red</h3>
22     <h3>G = 150 de Green</h3>
23     <h3>B = 150 de Blue</h3>
24     <h3>A = Opacidade de 0.3</h3>
25   </div>
26 </body>
27 </html>
```

Salvando e abrindo no navegador vai ficar assim:



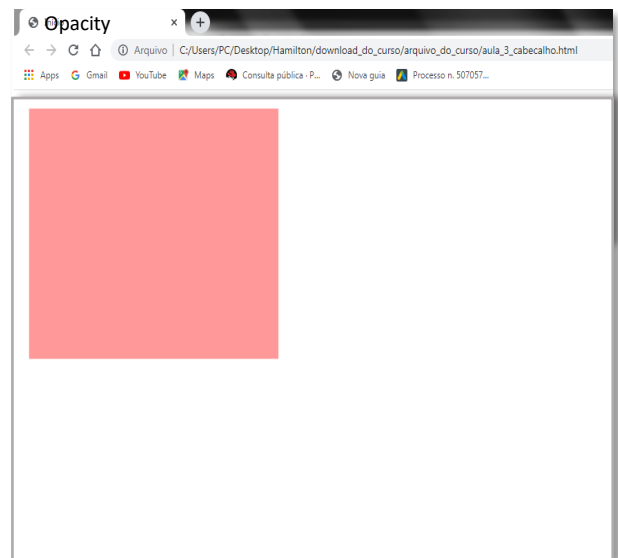
A propriedade CSS **opacity** especifica a transparência de um elemento, isto é, o grau no qual o background atrás do elemento é sobreposto. O valor aplica-se ao elemento como um todo, incluindo seu conteúdo, apesar de o valor não ser herdado por elementos filhos.

Vamos ver na pratica:

```

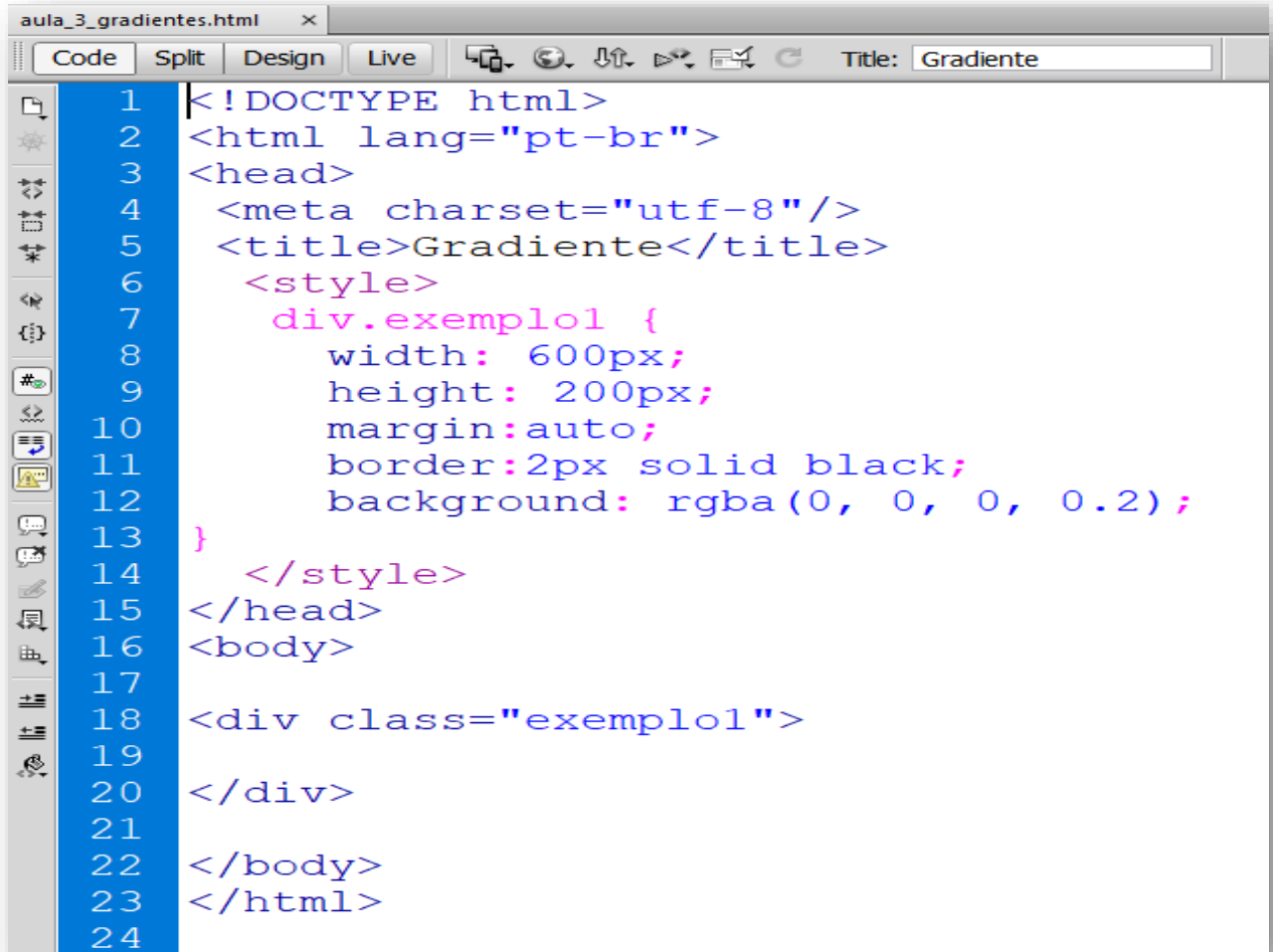
1 <!DOCTYPE HTML>
2 <html lang="pt-br">
3   <head>
4     <meta charset="UTF-8">
5     <title>Opacity</title>
6     <style type="text/css">
7       div{
8         width:300px;
9         height:300px;
10        background-color:#F00;
11        opacity:0.4;
12      }
13    </style>
14  </head>
15  <body>
16    <div></div>
17  </body>
18 </html>
  
```

Salvando e abrindo no navegador vai ficar assim:



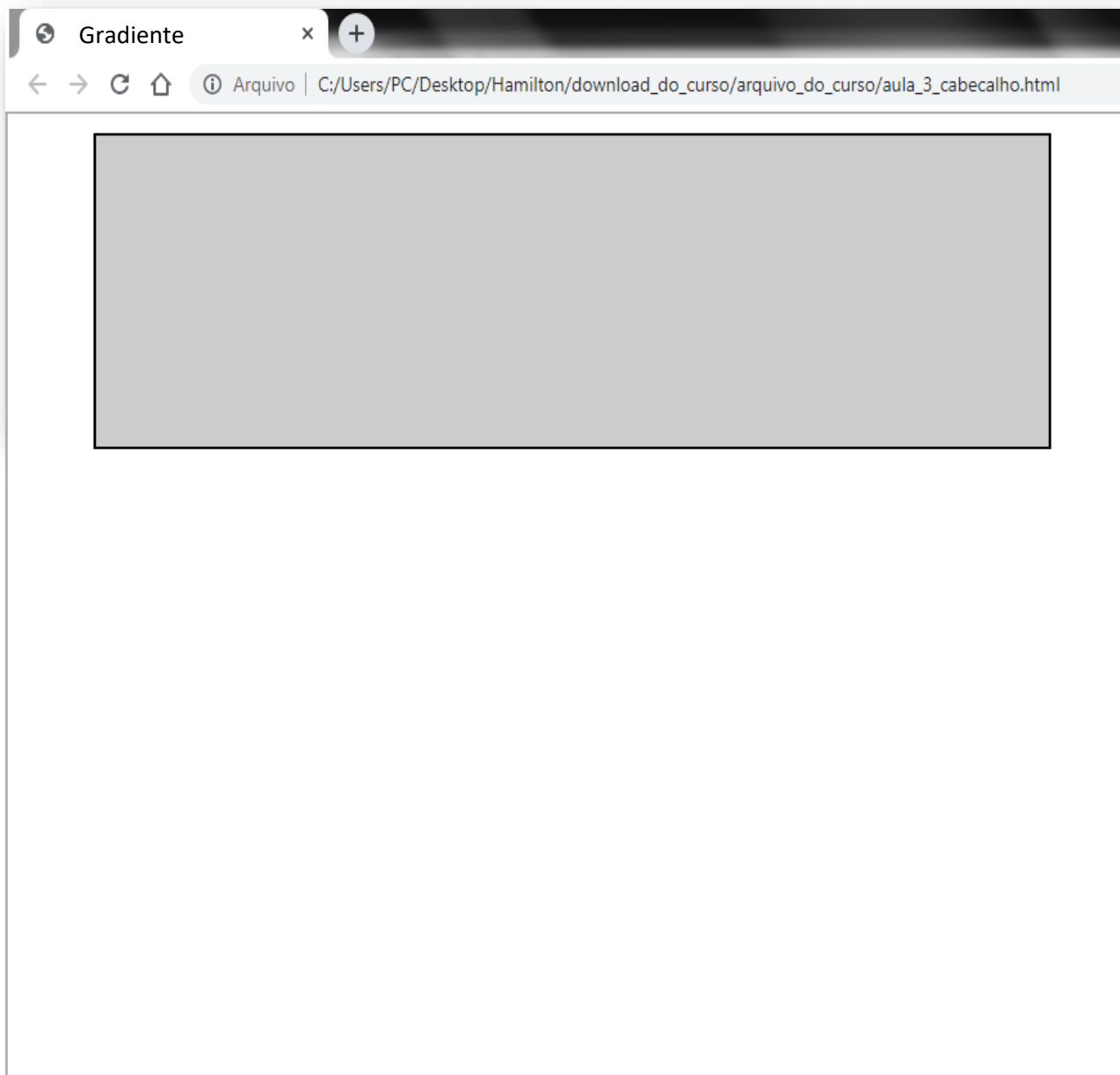
- **Declaração com Gradiente**

Uma aplicação prática bem interessante para uso de **RGBA** ou **HSLA** é a possibilidade de se criar uma máscara transparente a ser aplicada sobre uma cor sólida obtendo como resultado o efeito gradiente. O elemento **div.exemplo1** será estilizado com uma cor de fundo preta com 20% de opacidade. Siga descrição abaixo:



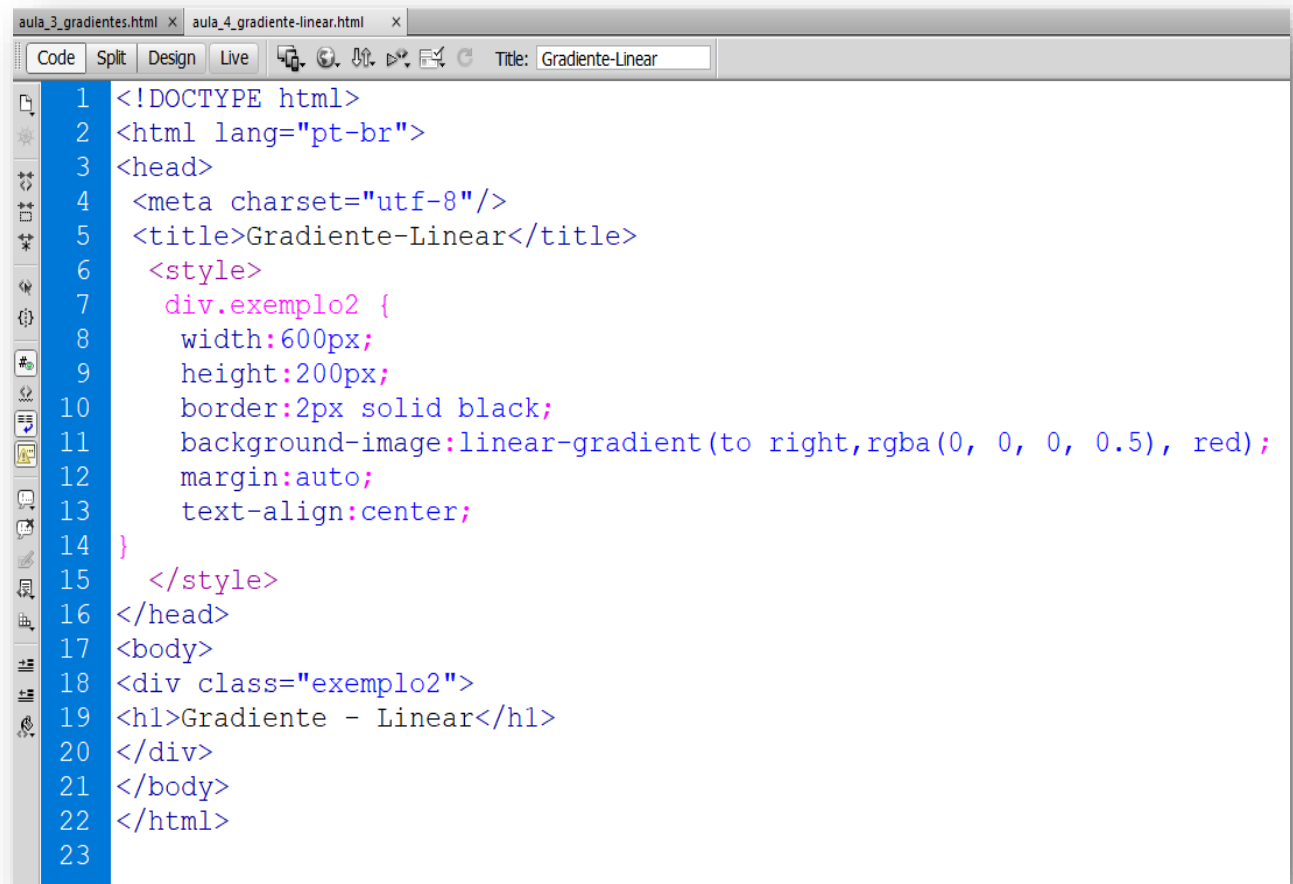
```
aula_3_gradientes.html x
Code Split Design Live
Title: Gradiente
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Gradiente</title>
6   <style>
7     div.exemplo1 {
8       width: 600px;
9       height: 200px;
10      margin:auto;
11      border:2px solid black;
12      background: rgba(0, 0, 0, 0.2);
13    }
14  </style>
15 </head>
16 <body>
17
18 <div class="exemplo1">
19
20 </div>
21
22 </body>
23 </html>
24
```

Salvando e abrindo no navegador vai ficar assim:



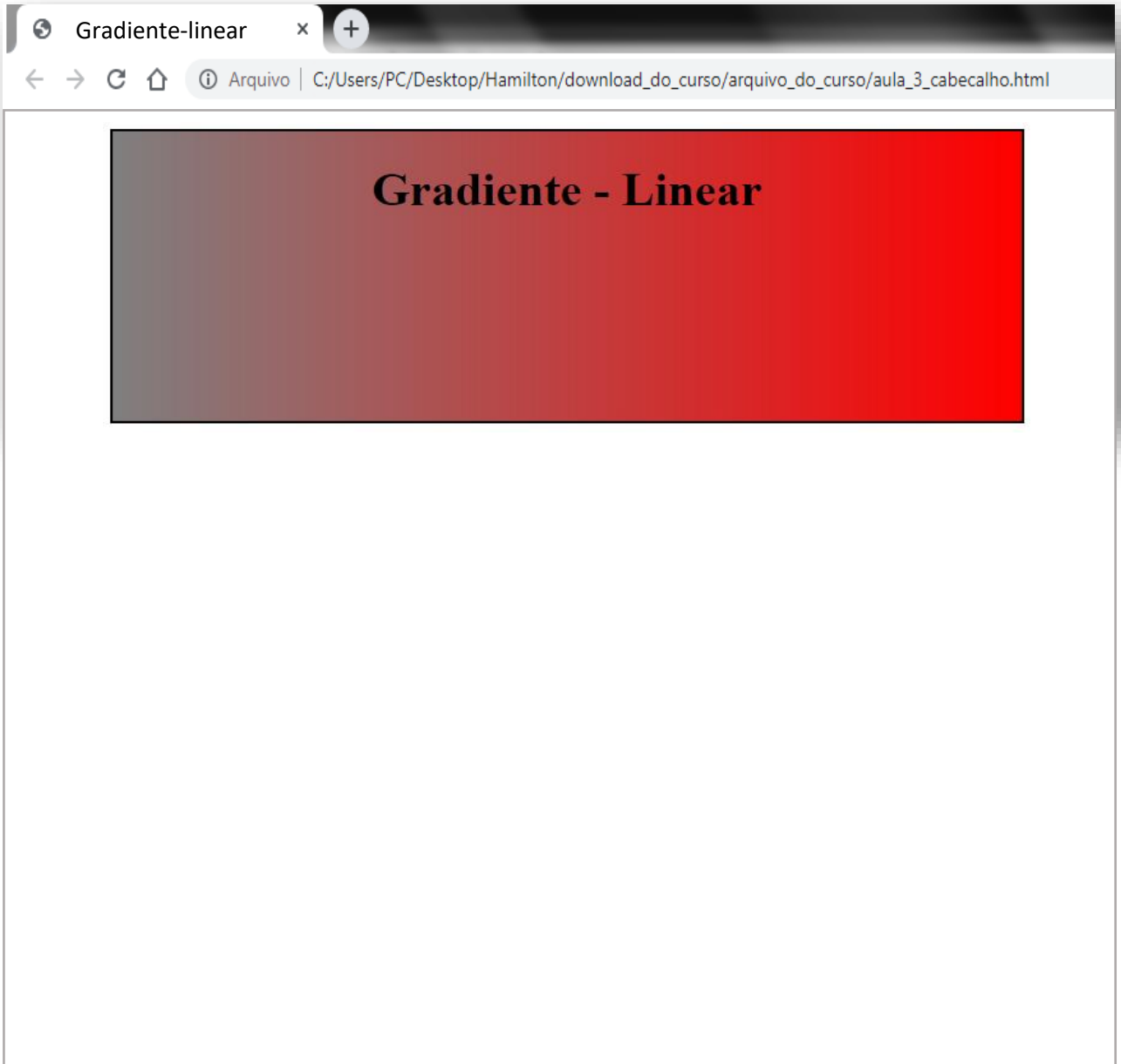
- **Declaração com Gradiente Linear**

Vamos agora criar um gradiente linear constituído de duas cores: a cor preta transparente mostrada no exemplo anterior e a cor vermelha e defini-lo (o gradiente) como imagem de fundo de um div. O elemento **div.exemplo2** será estilizado com uma cor de fundo preta, 50% de opacidade que vai se tornando vermelha totalmente opaca no sentido horizontal da esquerda para a direita, conforme mostrado a seguir:

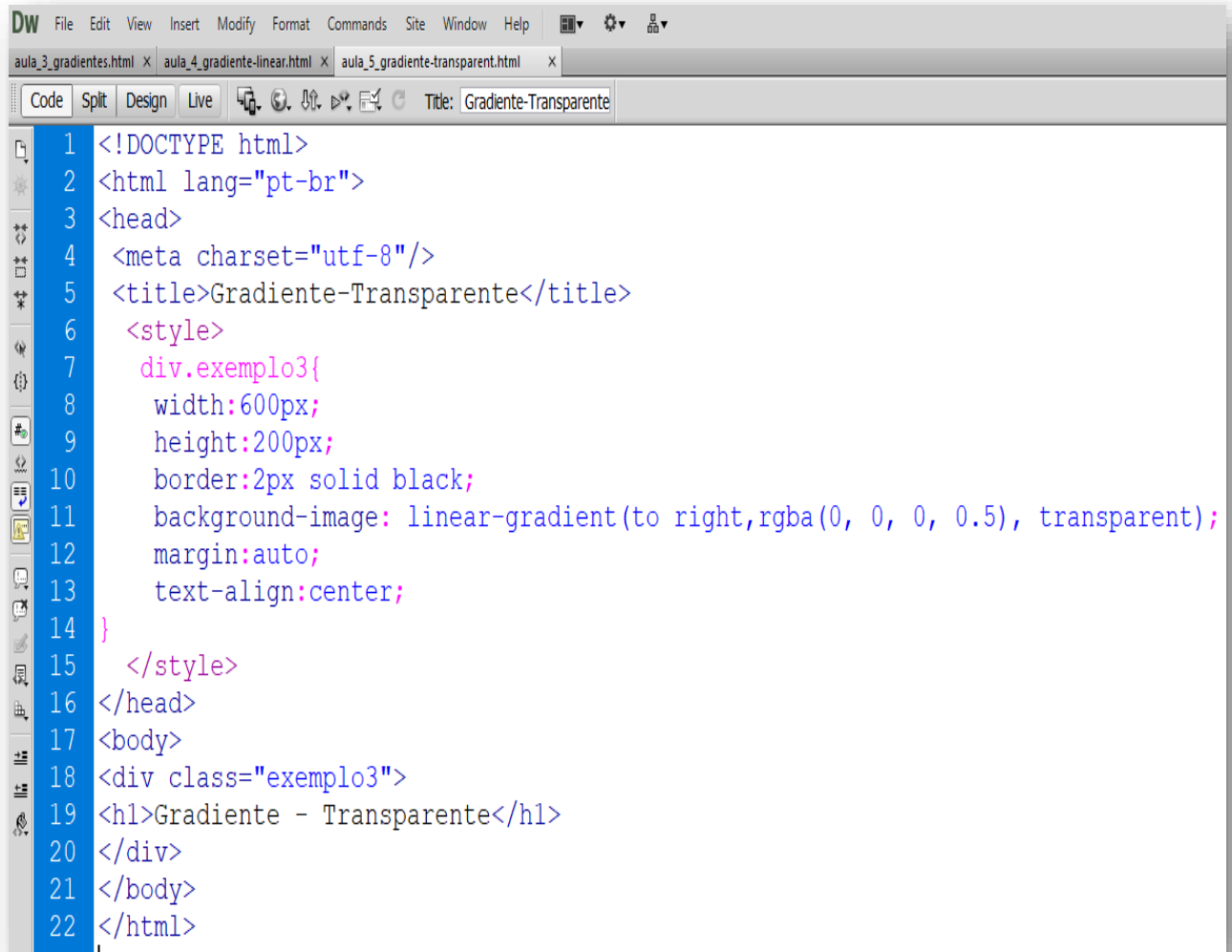


```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4 <meta charset="utf-8"/>
5 <title>Gradiente-Linear</title>
6 <style>
7   div.exemplo2 {
8     width:600px;
9     height:200px;
10    border:2px solid black;
11    background-image:linear-gradient(to right,rgba(0, 0, 0, 0.5), red);
12    margin:auto;
13    text-align:center;
14  }
15 </style>
16 </head>
17 <body>
18 <div class="exemplo2">
19 <h1>Gradiente - Linear</h1>
20 </div>
21 </body>
22 </html>
23
```

Salvando e abrindo no navegador vai ficar assim:

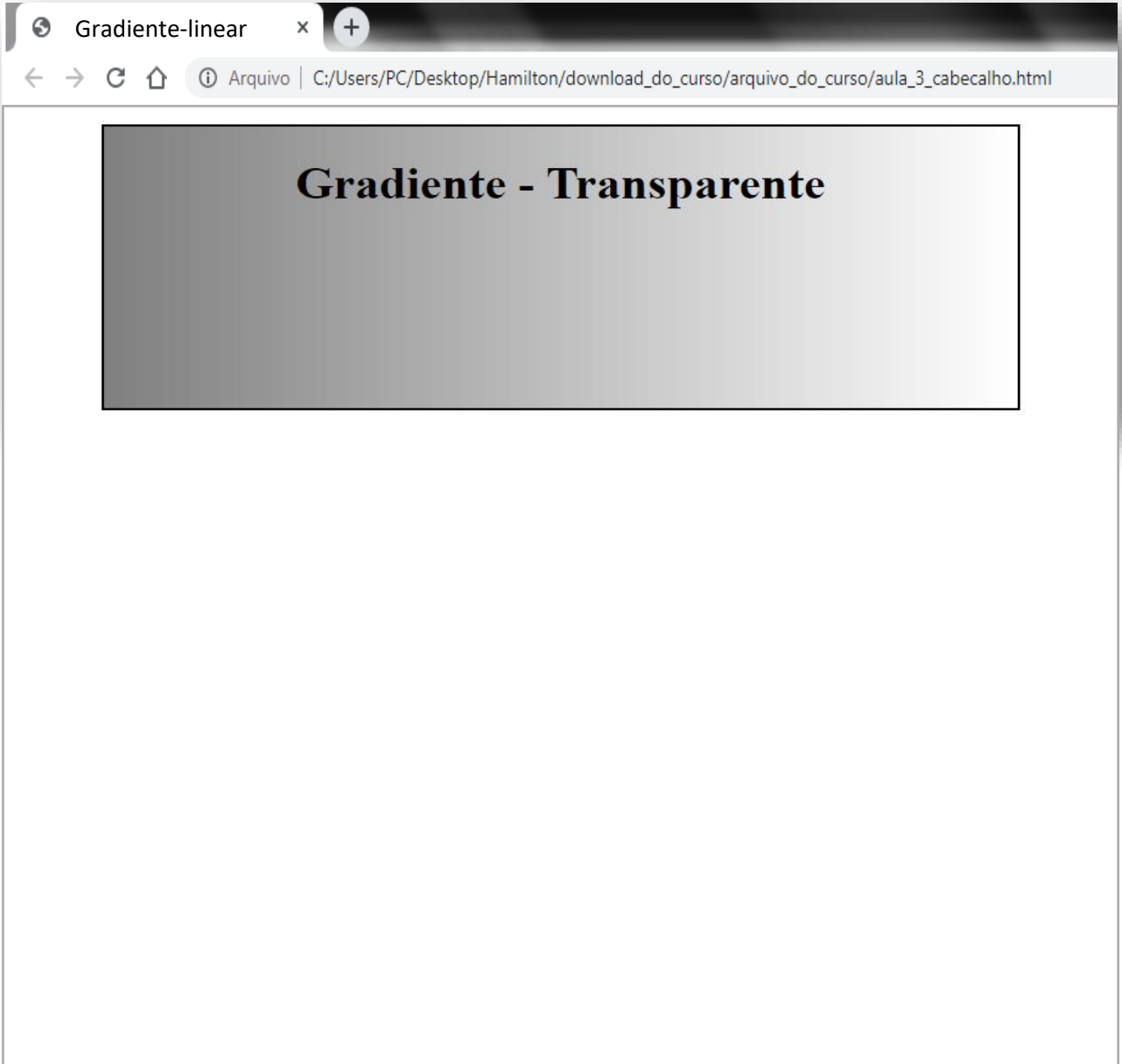


Suponha agora que nosso gradiente linear seja constituído das seguintes duas cores: a cor preta transparente (como usada no exemplo anterior) e a cor transparente. Lembre-se que em **CSS** um dos valores para cores é a palavra chave **transparent** que define uma cor transparente. O elemento **div.exemplo3** será estilizado com uma cor de fundo em gradiente de preta, 50% opaca para transparente, conforme mostrado a seguir:

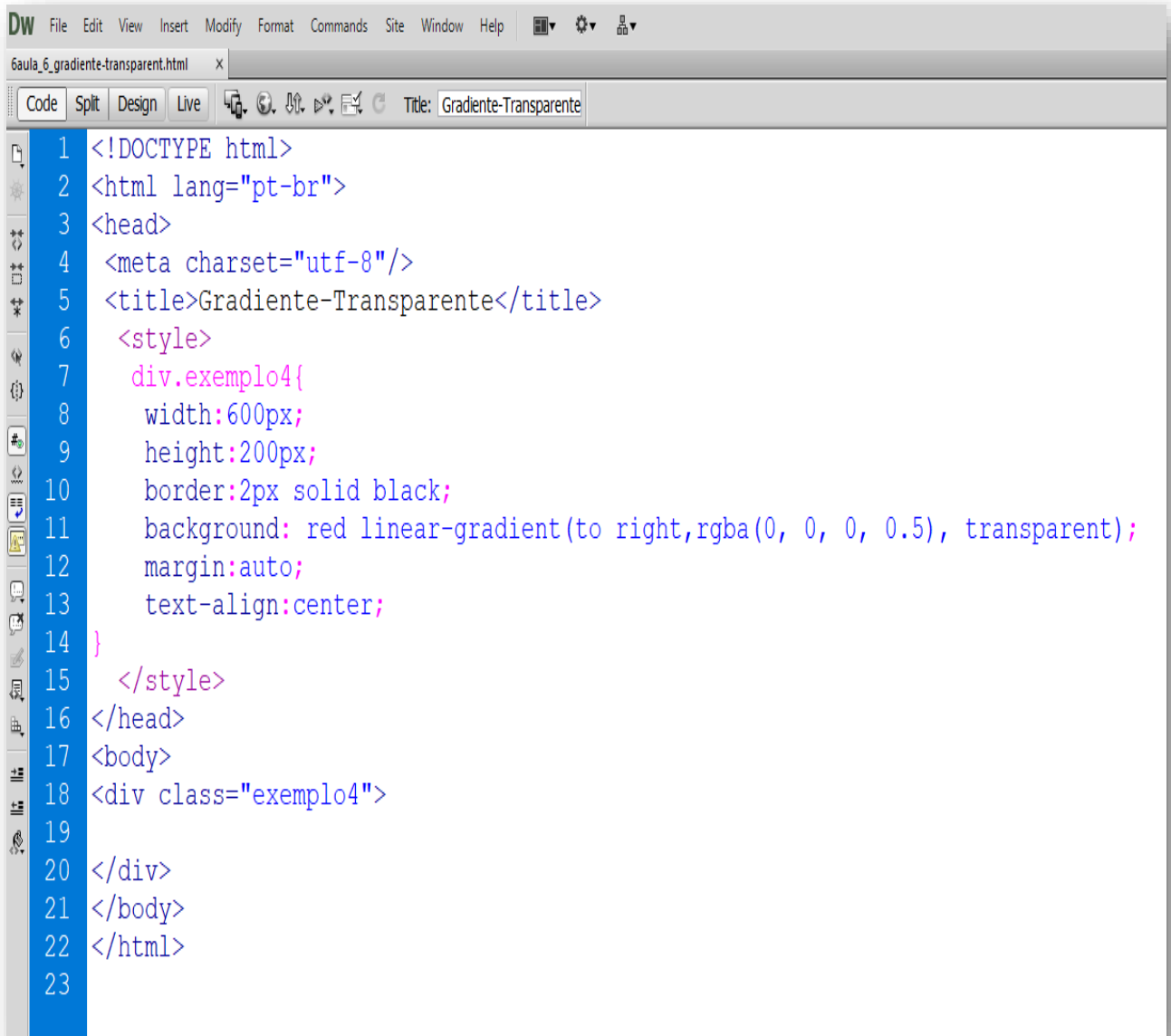


```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Gradiente-Transparente</title>
6   <style>
7     div.exemplo3{
8       width:600px;
9       height:200px;
10      border:2px solid black;
11      background-image: linear-gradient(to right,rgba(0, 0, 0, 0.5), transparent);
12      margin:auto;
13      text-align:center;
14    }
15  </style>
16 </head>
17 <body>
18 <div class="exemplo3">
19 <h1>Gradiente - Transparente</h1>
20 </div>
21 </body>
22 </html>
```

Salvando e abrindo no navegador vai ficar assim:



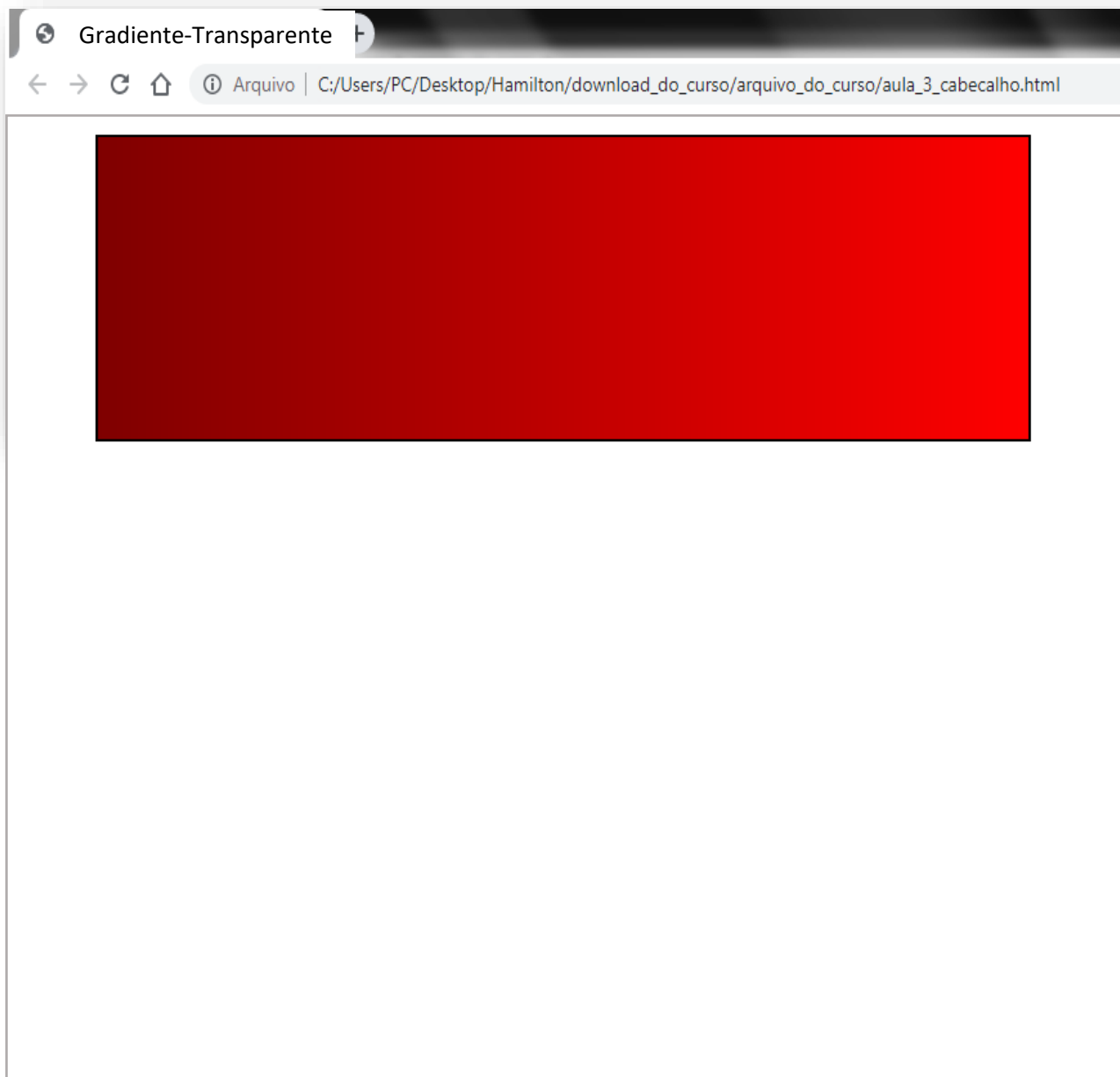
O que fizemos na **div exemplo3** foi criar uma máscara transparente. Suponha que essa máscara seja aplicada sobre um elemento div com uma cor de fundo sólida conforme a regra **CSS** mostrada a seguir.



```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4 <meta charset="utf-8"/>
5 <title>Gradiente-Transparente</title>
6 <style>
7   div.exemplo4{
8     width:600px;
9     height:200px;
10    border:2px solid black;
11    background: red linear-gradient(to right,rgba(0, 0, 0, 0.5), transparent);
12    margin:auto;
13    text-align:center;
14  }
15 </style>
16 </head>
17 <body>
18 <div class="exemplo4">
19
20 </div>
21 </body>
22 </html>
23
  
```

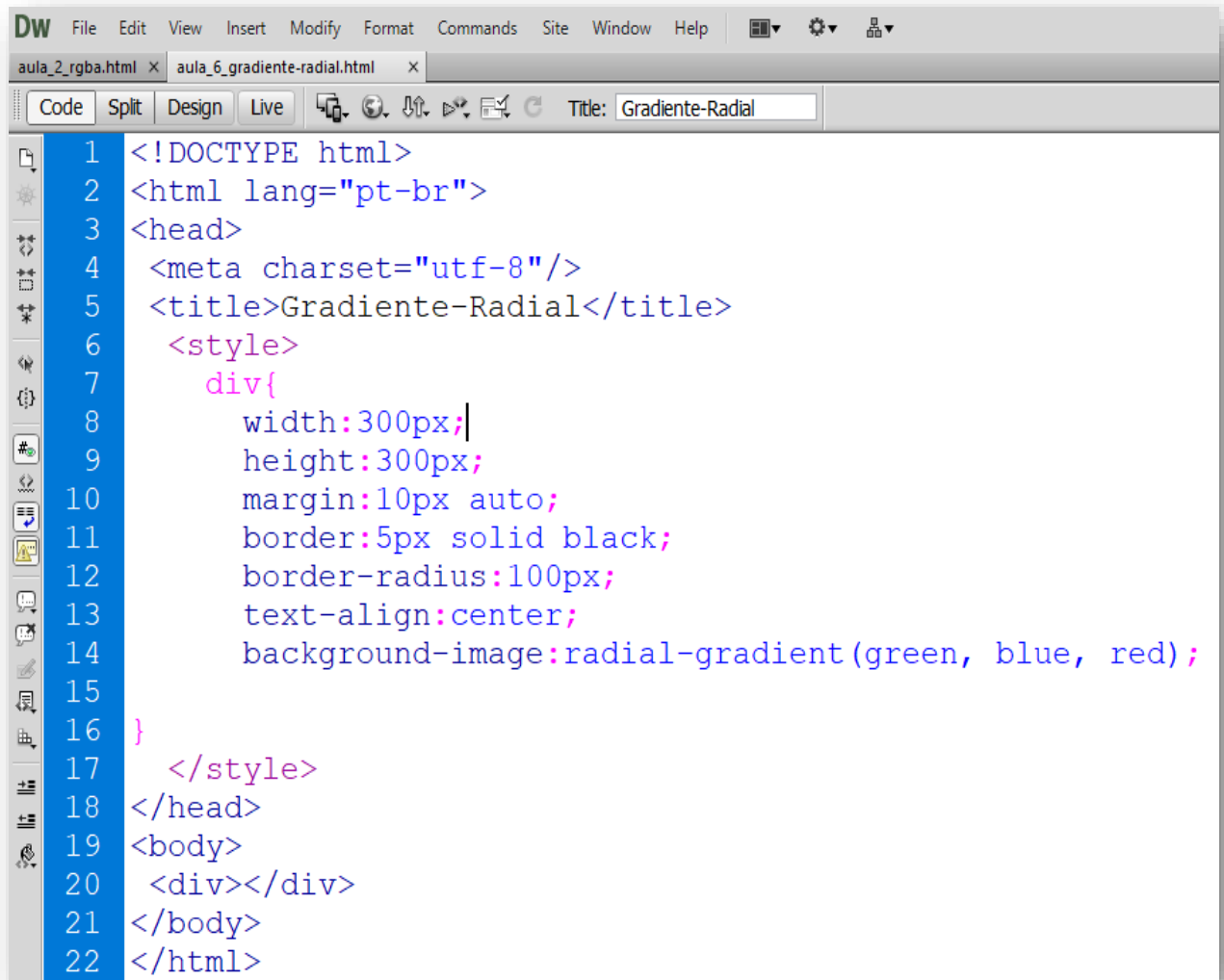
Salvando e abrindo no navegador vai ficar assim:



- **Declaração com Gradiente -Radial**

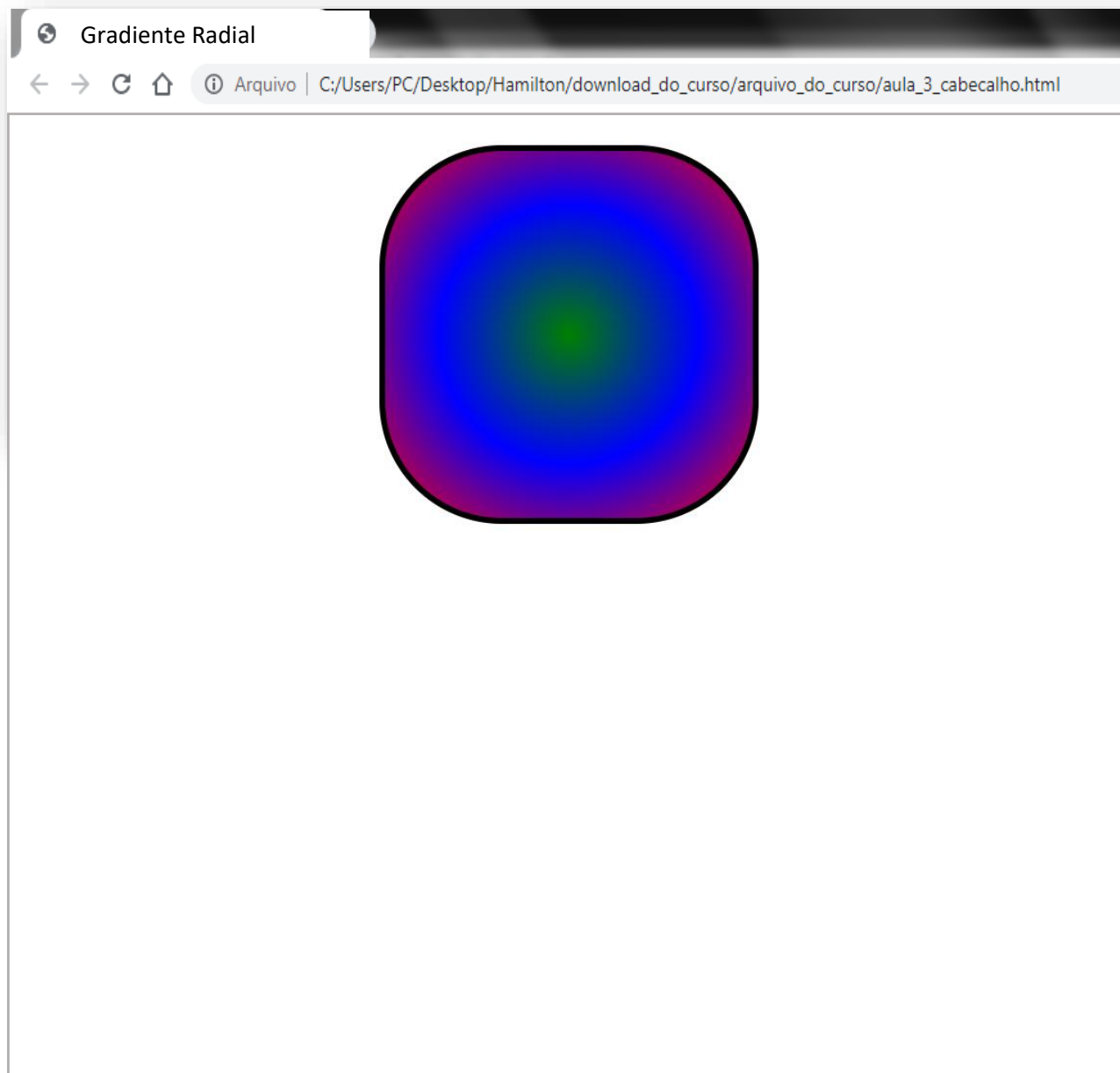
A função **radial-gradient()** também cria um efeito de transição entre duas ou mais cores. A principal diferença em relação ao gradiente linear é que ela é aplicada a partir do ponto central do elemento e irradia de forma circular ou elíptica.

Vamos ver como funciona na pratica:



```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4 <meta charset="utf-8"/>
5 <title>Gradiente-Radial</title>
6 <style>
7   div{
8     width:300px;|
9     height:300px;
10    margin:10px auto;
11    border:5px solid black;
12    border-radius:100px;
13    text-align:center;
14    background-image:radial-gradient(green, blue, red);
15  }
16 }
17 </style>
18 </head>
19 <body>
20 <div></div>
21 </body>
22 </html>
```

Salvando e abrindo no navegador vai ficar assim:



Atributo Box-Shadow

O atributo box-shadow precisa que o desenvolvedor passe alguns parâmetros padrões para que seja configurada algumas características das sombras, como esfumado, divisão de sombra, se a sombra terá alguma cor diferente da normal (preta) ou a própria caixa (box). Sua principal função é aplicar sombras em **box's**, ou qualquer tipo de elemento em sua página **HTML** usando apenas alguns códigos **CSS**.

```
box-shadow: 5px -9px 3px #000;
```

Vamos entender a ordem dos códigos:

- **Deslocamento horizontal da sombra:** A sombra de um elemento costuma estar um pouco deslocada em relação ao elemento que a produz (o box). Quanto mais deslocamento tiver uma sombra, o elemento que a produz parecerá que está mais separado da tela da página.
- **Deslocamento vertical da sombra:** O segundo valor que colocamos no atributo box-shadow é o deslocamento vertical da sombra com respeito à posição do elemento que a produz (box). Este valor é similar ao deslocamento horizontal. Valores positivos indicam que a sombra aparecerá abaixo do elemento e valores negativos farão com que a sombra apareça deslocada um pouco para cima.
- **Esfumado:** O terceiro valor indica quanto queremos que esteja esfumada a borda da sombra, ou seja, significa se a sombra vai ter uma transparência ou não.
- **Cor da sombra:** O último atributo que se indica no atributo box-shadow é a cor da sombra. Geralmente as sombras possuem uma cor preta ou cinza, porém podemos definir a cor que quisermos para a sombra, basta colocar seu código hexadecimal

Compatibilidade entre os navegadores:

Navegadores antigos não possuem muito suporte para CSS3, mas caso você esteja utilizando os navegadores atualizados, irá funcionar perfeitamente.

Por enquanto, podemos utilizar box-shadow nas versões mais modernas do navegador Opera. Por sua parte, navegadores baseados em Mozilla e WebKit têm suporte a esta funcionalidade de CSS3, porém, através de uns atributos CSS com uma ligeira variação em seu nome.

Atributo box-shadow para navegadores baseados em Mozilla, como Firefox: De maneira temporária, Firefox é capaz de interpretar o atributo -moz-box-shadow, por exemplo:

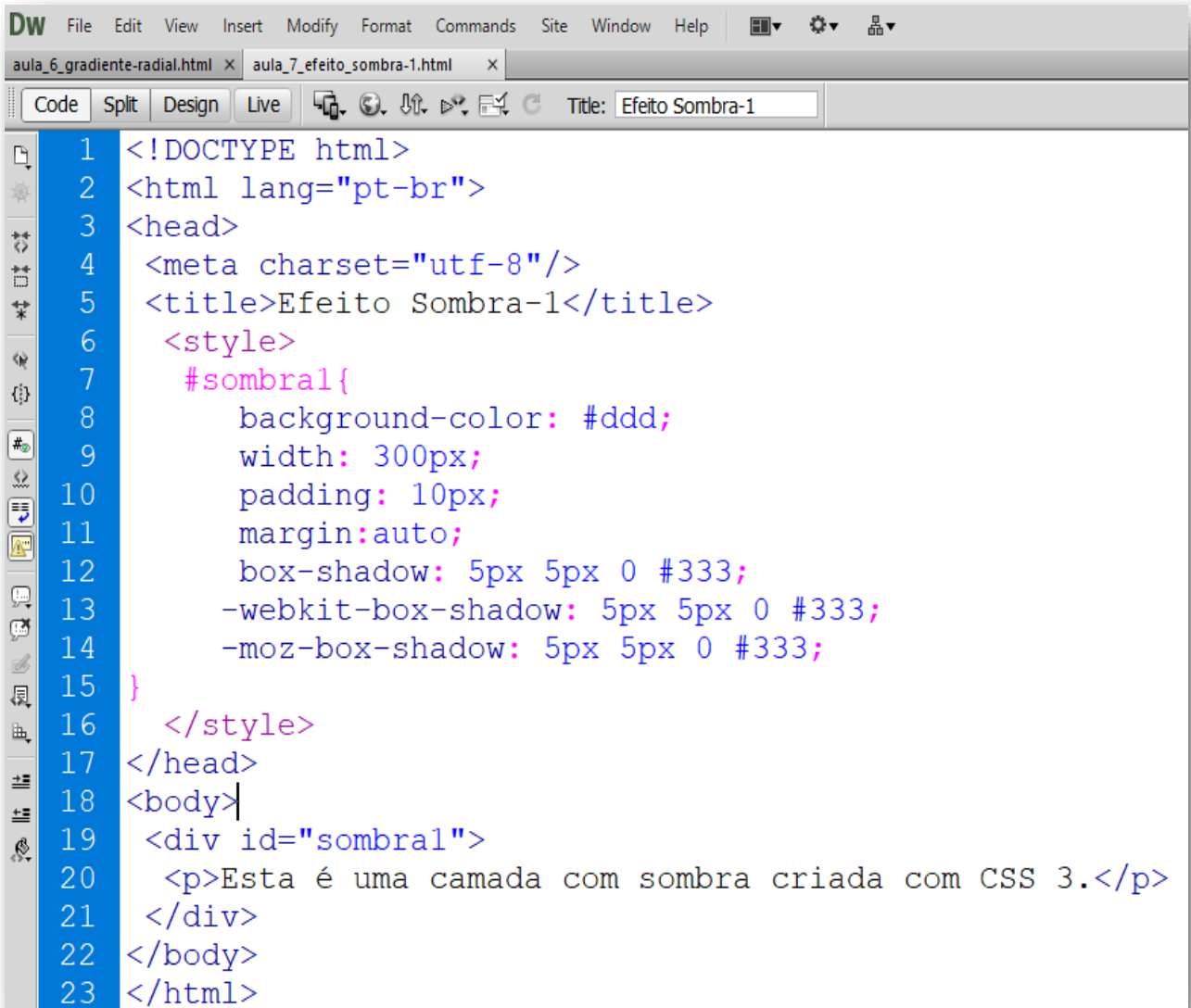
```
-moz-box-shadow: 1px 1px 0px #090;
```

Atributo box-shadow para navegadores baseados em WebKit, como Safari ou **Google Chrome**: Nesses momentos e de maneira temporária, navegadores como Chrome ou Safari entendem o atributo: -webkit-box-shadow, por exemplo:

```
-webkit-box-shadow: 3px 3px 1px #fc8;
```

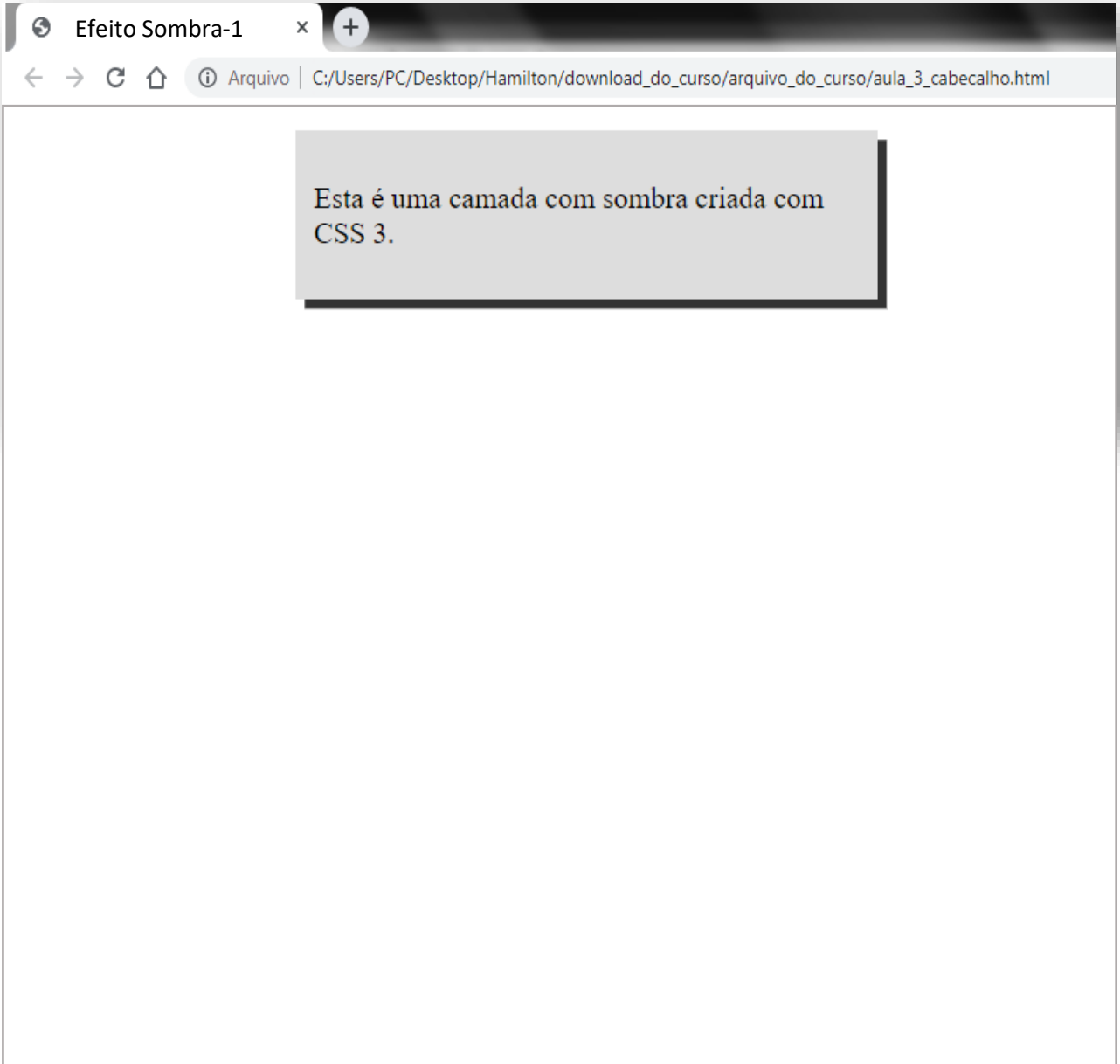

Vamos ver alguns exemplos na prática:

Exemplo-1:

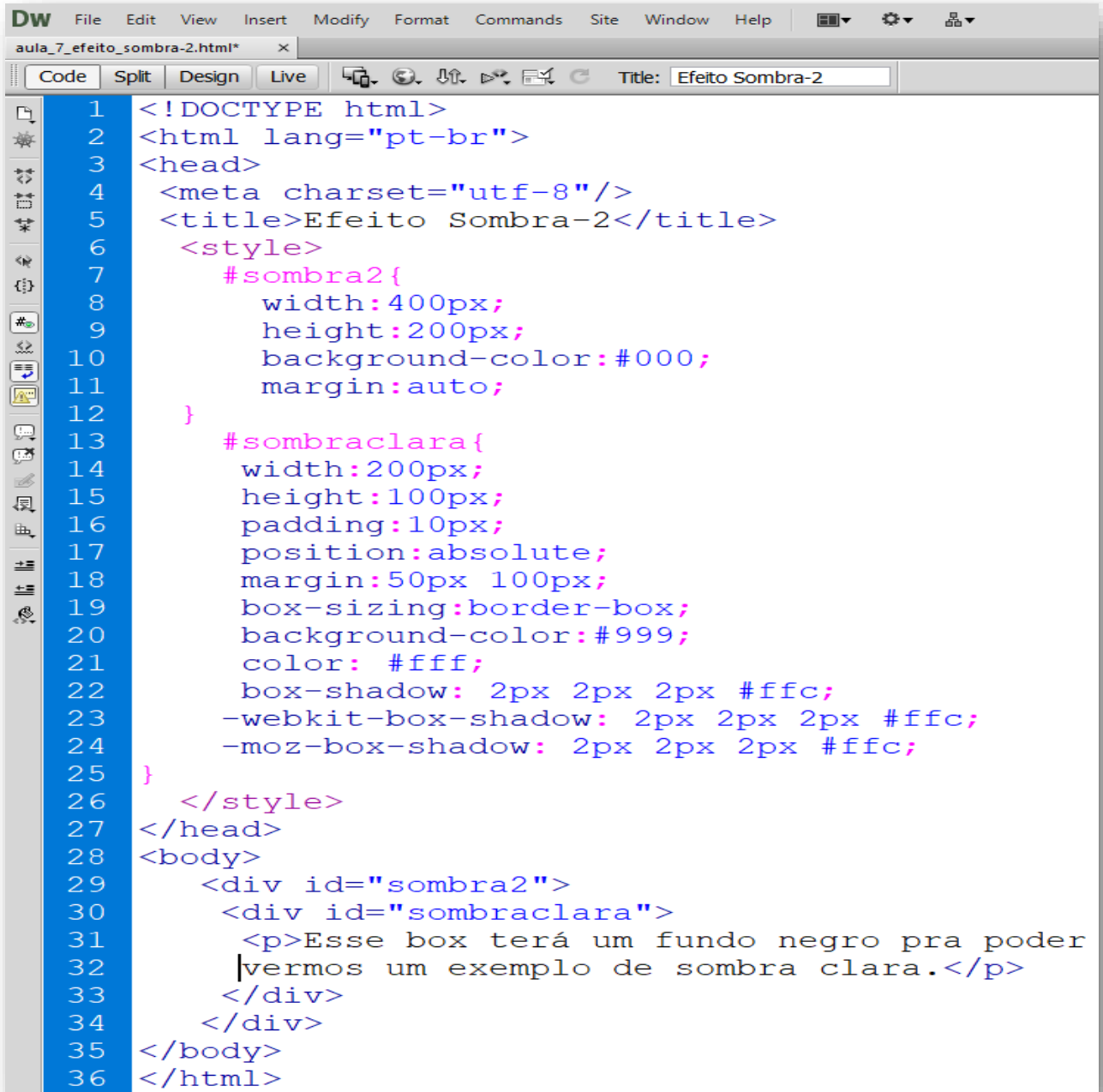


```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Efeito Sombra-1</title>
6   <style>
7     #sombra1{
8       background-color: #ddd;
9       width: 300px;
10      padding: 10px;
11      margin:auto;
12      box-shadow: 5px 5px 0 #333;
13      -webkit-box-shadow: 5px 5px 0 #333;
14      -moz-box-shadow: 5px 5px 0 #333;
15    }
16  </style>
17 </head>
18 <body>
19   <div id="sombra1">
20     <p>Esta é uma camada com sombra criada com CSS 3.</p>
21   </div>
22 </body>
23 </html>
```

Salvando e abrindo no navegador vai ficar assim:



Exemplo-2:

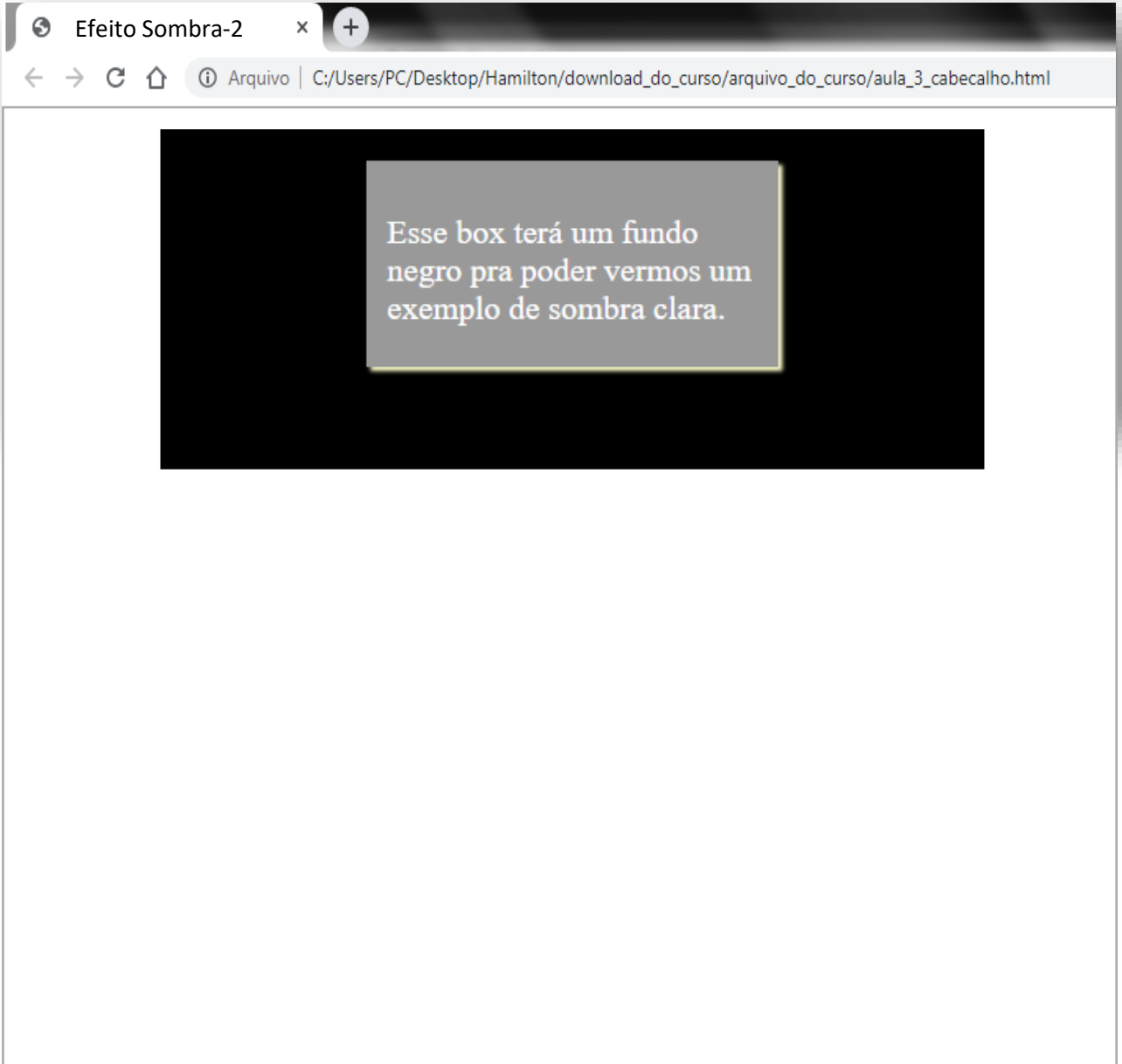


```

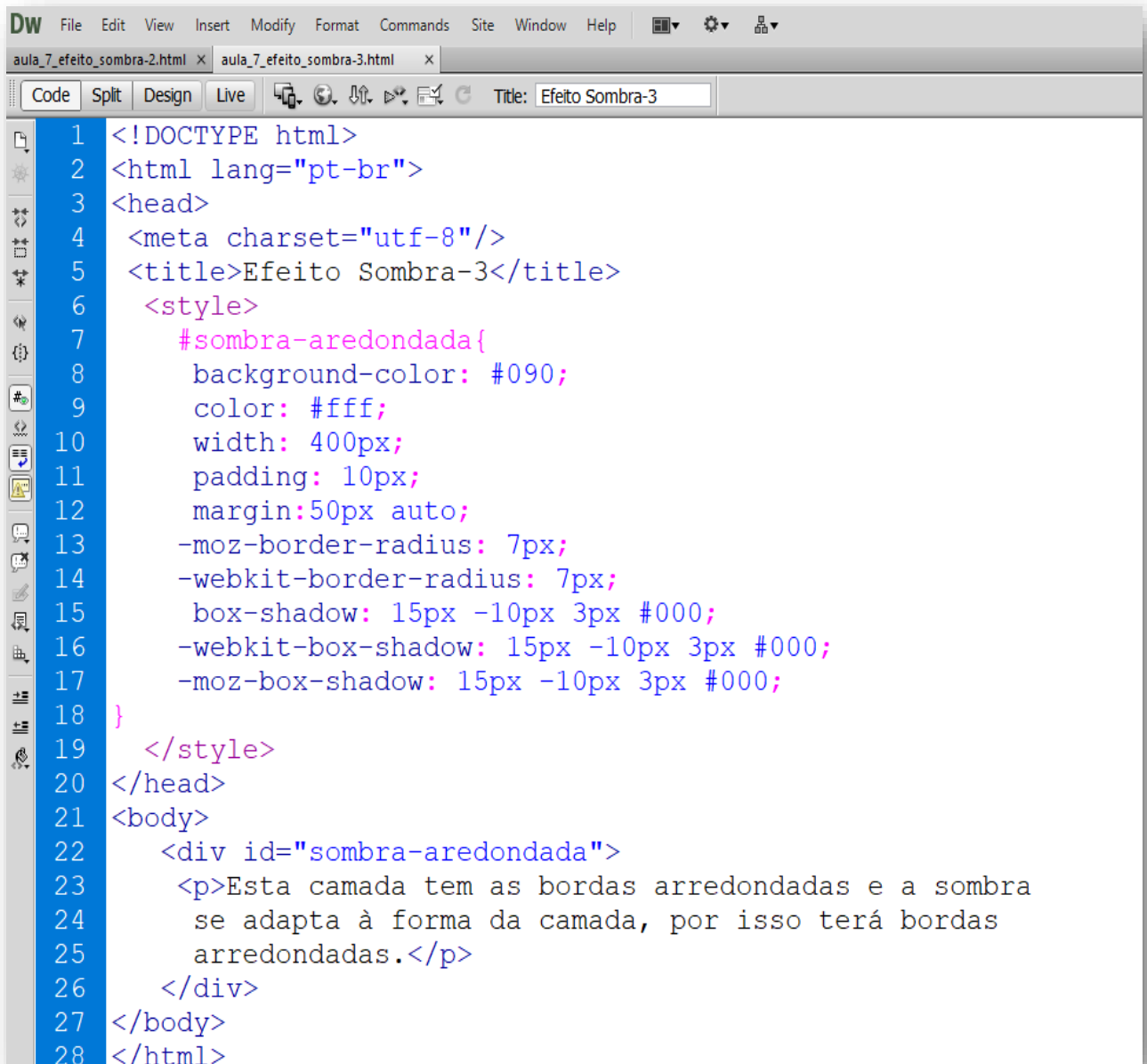
1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4    <meta charset="utf-8"/>
5    <title>Efeito Sombra-2</title>
6    <style>
7      #sombra2{
8        width:400px;
9        height:200px;
10       background-color:#000;
11       margin:auto;
12     }
13     #sombraclara{
14       width:200px;
15       height:100px;
16       padding:10px;
17       position:absolute;
18       margin:50px 100px;
19       box-sizing:border-box;
20       background-color:#999;
21       color: #fff;
22       box-shadow: 2px 2px 2px #ffc;
23       -webkit-box-shadow: 2px 2px 2px #ffc;
24       -moz-box-shadow: 2px 2px 2px #ffc;
25     }
26   </style>
27 </head>
28 <body>
29   <div id="sombra2">
30     <div id="sombraclara">
31       <p>Esse box terá um fundo negro pra poder
32       |vermos um exemplo de sombra clara.</p>
33     </div>
34   </div>
35 </body>
36 </html>

```

Salvando e abrindo no navegador vai ficar assim:



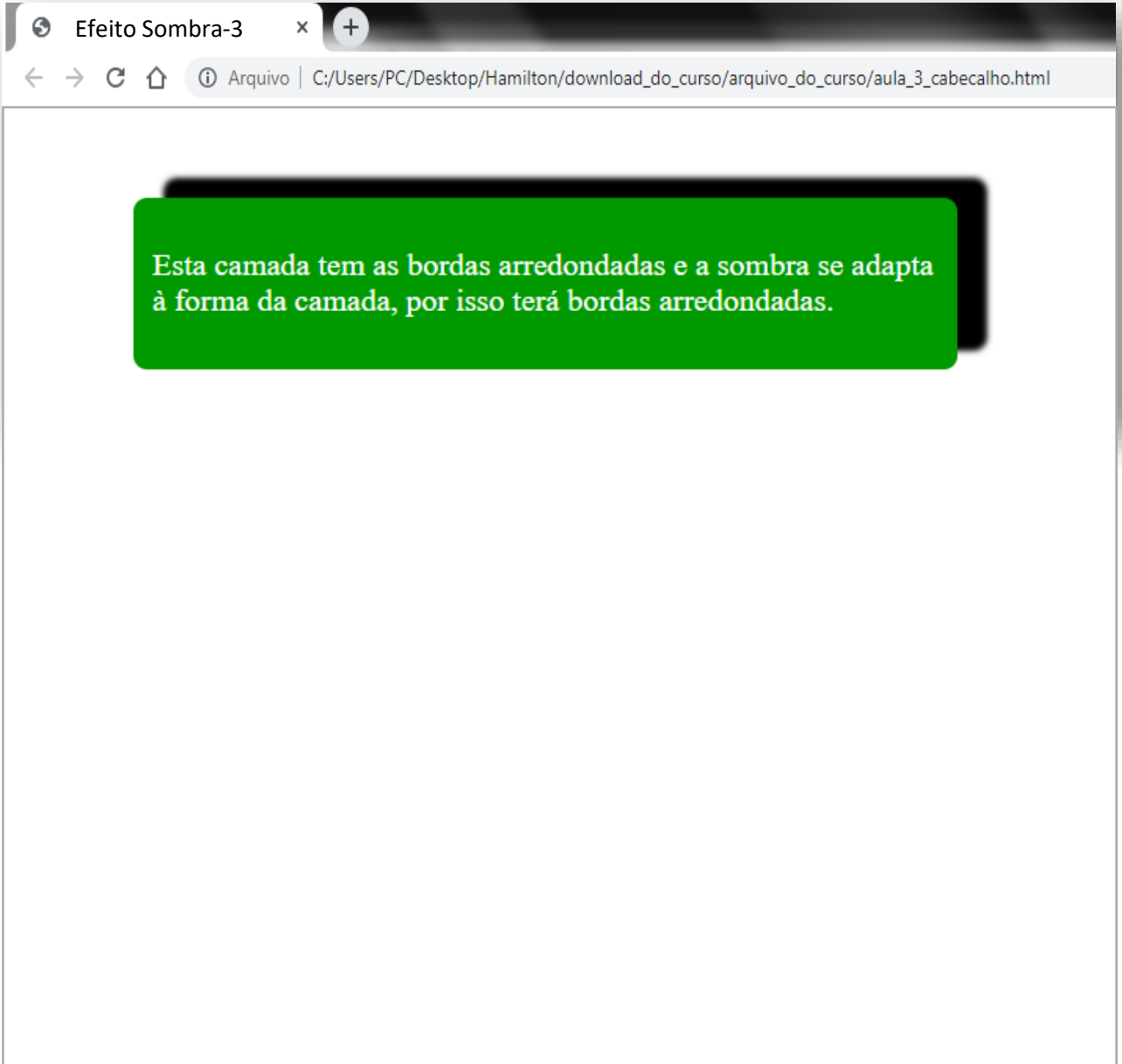
Exemplo-3:



```

1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="utf-8"/>
5   <title>Efeito Sombra-3</title>
6   <style>
7     #sombra-aredondada{
8       background-color: #090;
9       color: #fff;
10      width: 400px;
11      padding: 10px;
12      margin:50px auto;
13      -moz-border-radius: 7px;
14      -webkit-border-radius: 7px;
15      box-shadow: 15px -10px 3px #000;
16      -webkit-box-shadow: 15px -10px 3px #000;
17      -moz-box-shadow: 15px -10px 3px #000;
18    }
19  </style>
20 </head>
21 <body>
22   <div id="sombra-aredondada">
23     <p>Esta camada tem as bordas arredondadas e a sombra
24       se adapta à forma da camada, por isso terá bordas
25       arredondadas.</p>
26   </div>
27 </body>
28 </html>
  
```

Salvando e abrindo no navegador vai ficar assim:

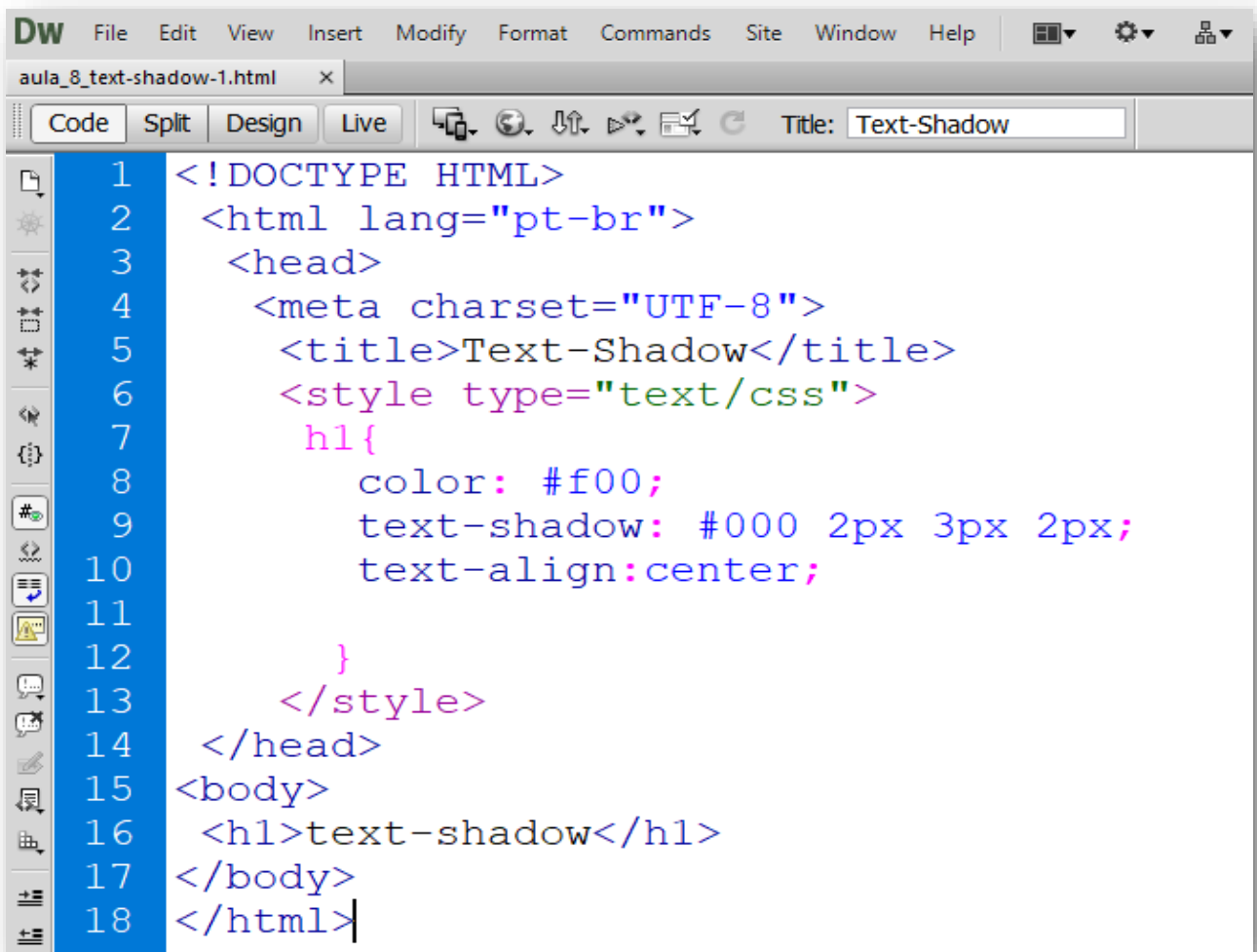


Atributo Text-Shadow

O `text-shadow` é um recurso muito legal que foi adicionado junto com outras inúmeras novidades do **CSS3**, ele é responsável por criar o efeito de sombras nos textos sem precisar de utilizar nenhuma imagem para isso, deixando seu site mais leve e com recursos visuais mais bonitos, podendo proporcionar uma experiência legal aos usuários.

Infelizmente os navegadores mais antigos e o Internet Explorer não possuem suporte para essa funcionalidade. Assim como em toda declaração **CSS**, devemos aplicar o efeito sobre algum elemento **HTML**. Vamos criar um documento **HTML** e adicionar o elemento `<h1>`, nele iremos aplicar o efeito **text-shadow**:

Exemplo:1



```
1 <!DOCTYPE HTML>
2 <html lang="pt-br">
3 <head>
4 <meta charset="UTF-8">
5 <title>Text-Shadow</title>
6 <style type="text/css">
7   h1{
8     color: #f00;
9     text-shadow: #000 2px 3px 2px;
10    text-align:center;
11  }
12 </style>
13 </head>
14 <body>
15 <h1>text-shadow</h1>
16 </body>
17 </html>
```

Salvando e abrindo no navegador vai ficar assim:



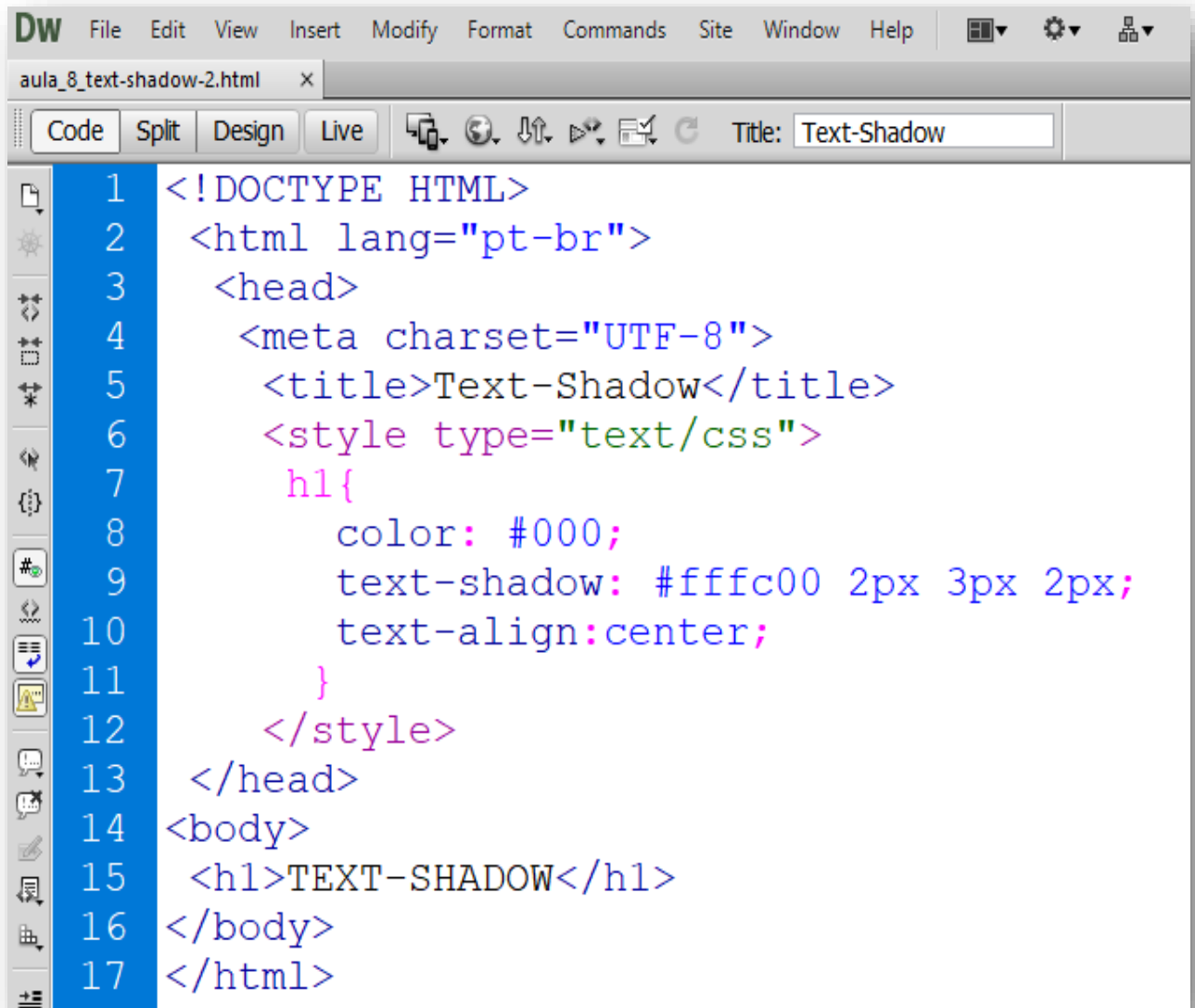
Vamos entender o efeito que acabamos de fazer:

Aplicamos uma cor vermelha ao texto e uma borda na cor preta, utilizamos valores contidos na tabela de cores em **HTML**. Na mesma declaração devemos dizer o posicionamento da sombra sobre o elemento em que ela está sendo aplicada. A ordem de declaração do posicionamento é:

- Distância horizontal
- Distância vertical
- Raio

Como em qualquer outra declaração em **CSS**, podemos utilizar qualquer unidade de medida para definir a distância e o raio da sombra, não apenas pixels, permitindo ao desenvolvedor utilizar a que mais agrada e mais atender às suas necessidades.

Exemplo:2



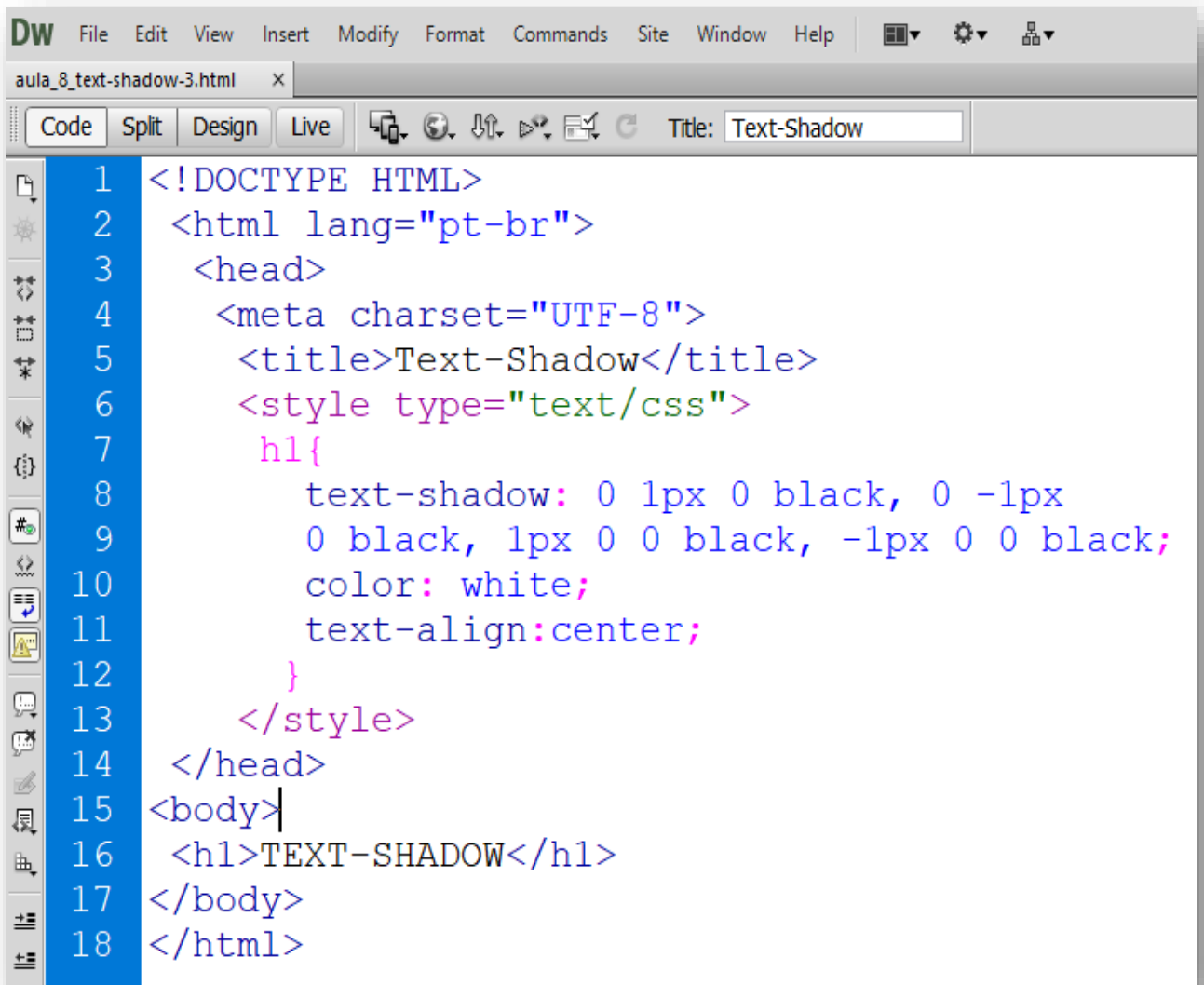
```
1 <!DOCTYPE HTML>
2 <html lang="pt-br">
3 <head>
4 <meta charset="UTF-8">
5 <title>Text-Shadow</title>
6 <style type="text/css">
7   h1{
8     color: #000;
9     text-shadow: #fffc00 2px 3px 2px;
10    text-align:center;
11  }
12 </style>
13 </head>
14 <body>
15 <h1>TEXT-SHADOW</h1>
16 </body>
17 </html>
```

Salvando e abrindo no navegador vai ficar assim:



Nesse segundo exemplo fizemos uma sombra de cor amarela em um texto de cor preta, você pode ficar a vontade em utilizar qualquer uma das cores da tabela.

Exemplo:3



```
1 <!DOCTYPE HTML>
2 <html lang="pt-br">
3 <head>
4 <meta charset="UTF-8">
5 <title>Text-Shadow</title>
6 <style type="text/css">
7   h1{
8     text-shadow: 0 1px 0 black, 0 -1px
9     0 black, 1px 0 0 black, -1px 0 0 black;
10    color: white;
11    text-align:center;
12  }
13 </style>
14 </head>
15 <body>
16 <h1>TEXT-SHADOW</h1>
17 </body>
18 </html>
```

Salvando e abrindo no navegador vai ficar assim:



Neste ultimo exemplo fizemos uma sombra preta em um texto de cor branca e percebeba que deu um efeito de contorno do texto.

CSS3 ANIMATIONS:

A chave para criar animações em **CSS3** é regra **@keyframes**. Nesta regra, a animação é criada. Assim, o usuário precisa especificar um estilo **CSS** dentro da regra **@keyframes** e a animação irá mudar gradualmente a partir do estilo atual para o novo estilo.

vamos começar com pequenas noções básicas de animações **CSS3**. Como todos sabemos, a animação é uma nova propriedade CSS que permite a animar a maioria dos elementos **HTML** (como div, h1 e span) sem **JavaScript** ou **Flash**.

Os usuários podem alterar os estilos como eles quiserem e quantas vezes quiserem. A única coisa que precisa ser feita é o de especificar quando a mudança ocorrerá em por cento, que é o mesmo como 0% e 100%. Em termos simples, 0% é o início da animação, 100% é quando a animação é completa.

Nota: Para utilizar essa propriedade em outros navegadores, é preciso adicionar um prefixo. Firefox exige o prefixo -moz, Chrome e Safari exigem o prefixo -webkit, e Opera requerem o prefixo -o, entre outros.

Agora, quando a animação é criada pelo usuário usando a regra **@keyframes**, as animações criadas devem se linkar para o seletor. Se a animação não for linkada, então ela não vai mostrar resultado nenhum, ou seja, nenhuma animação será feita.

Nesse processo, o usuário precisa definir duas propriedades diferentes. Estas propriedades podem ser qualquer coisa, mas, neste artigo iremos definir as duas propriedades mais importantes isto é, o nome da animação e a duração da animação.

Vamos criar uma animação muito simples. Neste, estamos linkando o elemento "**CSS3 animation**" a um elemento div e, em seguida, fornecer a duração da animação.

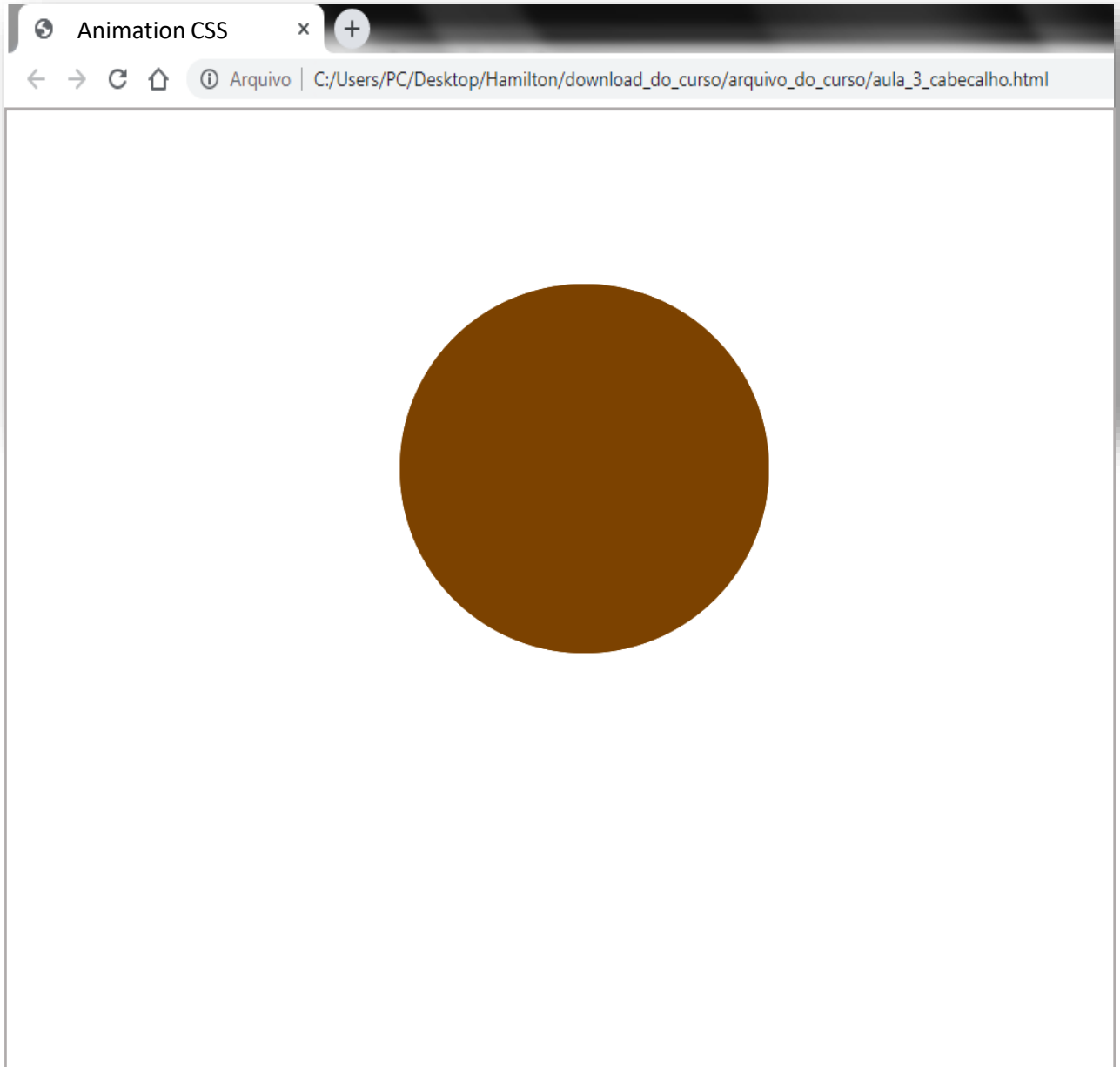
Exemplo -1:

```

1 <!DOCTYPE HTML>
2 <html lang="pt-br">
3 <head>
4 <meta charset="UTF-8">
5 <title>Animation CSS3</title>
6 <style type="text/css">
7   div{
8     width:300px;
9     height:300px;
10    border-radius:100%;
11    margin:100px auto;
12    background:red;
13    animation:aprendendoCSS3animation 10s;
14    -moz-animation:aprendendoCSS3animation 10s; /* Firefox */
15    -webkit-animation:aprendendoCSS3animation 10s; /* Safari e Chrome */
16    -o-animation:aprendendoCSS3animation 10s; /* Opera */
17  }
18 @keyframes aprendendoCSS3animation
19 {
20   from {background:red;}
21   to {background:green;}
22 }
23 @-moz-keyframes aprendendoCSS3animation /* Firefox */
24 {
25   from {background:red;}
26   to {background:green;}
27 }
28 @-webkit-keyframes aprendendoCSS3animation /* Safari e Chrome */
29 {
30   from {background:red;}
31   to {background:green;}
32 }
33 @-o-keyframes aprendendoCSS3animation /* Opera */
34 {
35   from {background:red;}
36   to {background:green;}
37 }
38 </style>
39 </head>
40 <body>
41   <div></div>
42 </body>
43 </html>
44
45

```

Salvando e abrindo no navegador vai ficar assim:

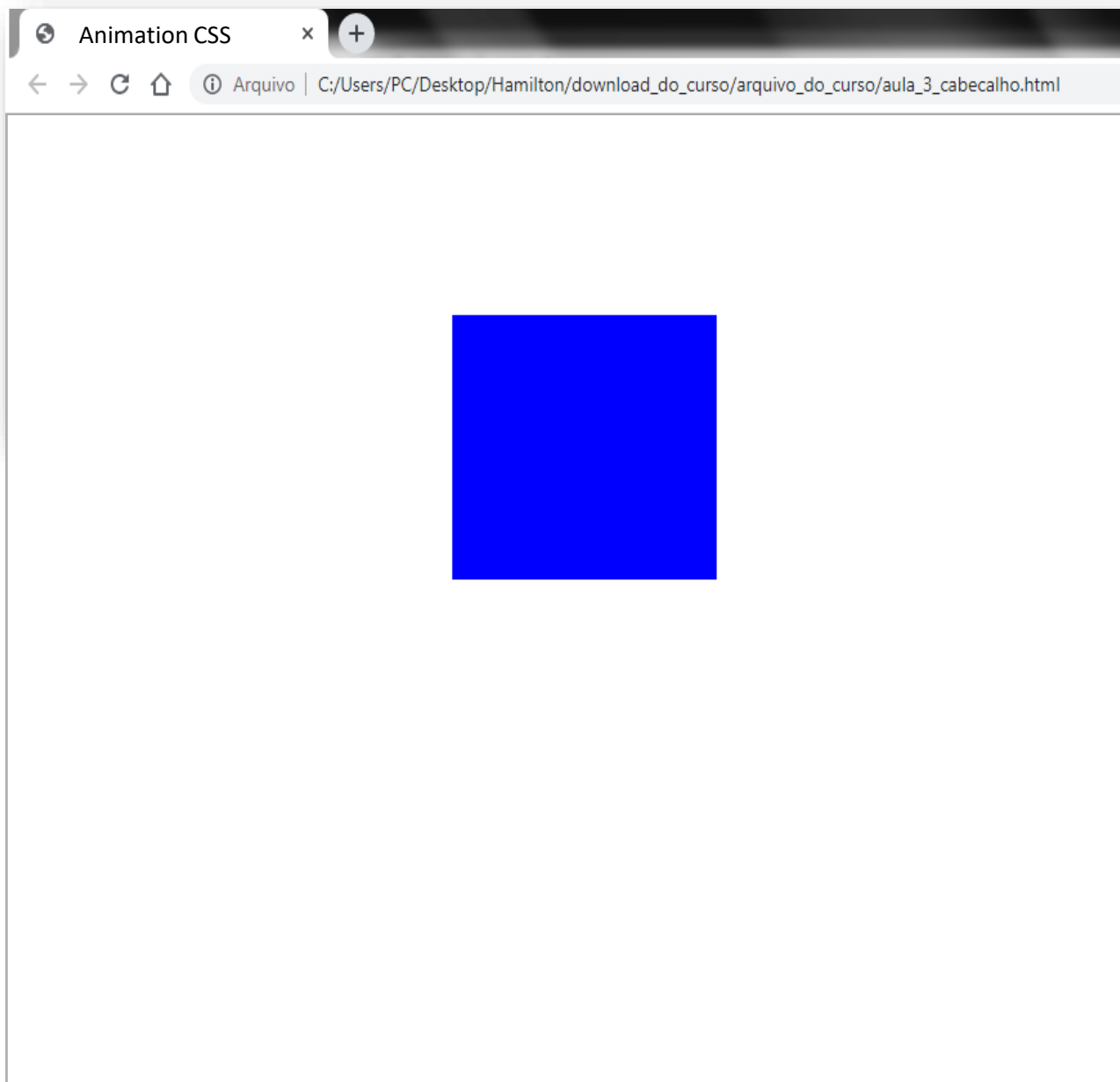


Vamos alterar agora as cores diferentes quando a animação está sendo executada. Por exemplo, a animação muda de cor após os intervalos regulares de tempo, ou seja, a 25% de um determinado tempo, 50% do tempo e de 75% de um dado tempo. Finalmente 100% de um dado tempo.

Exemplo-2:

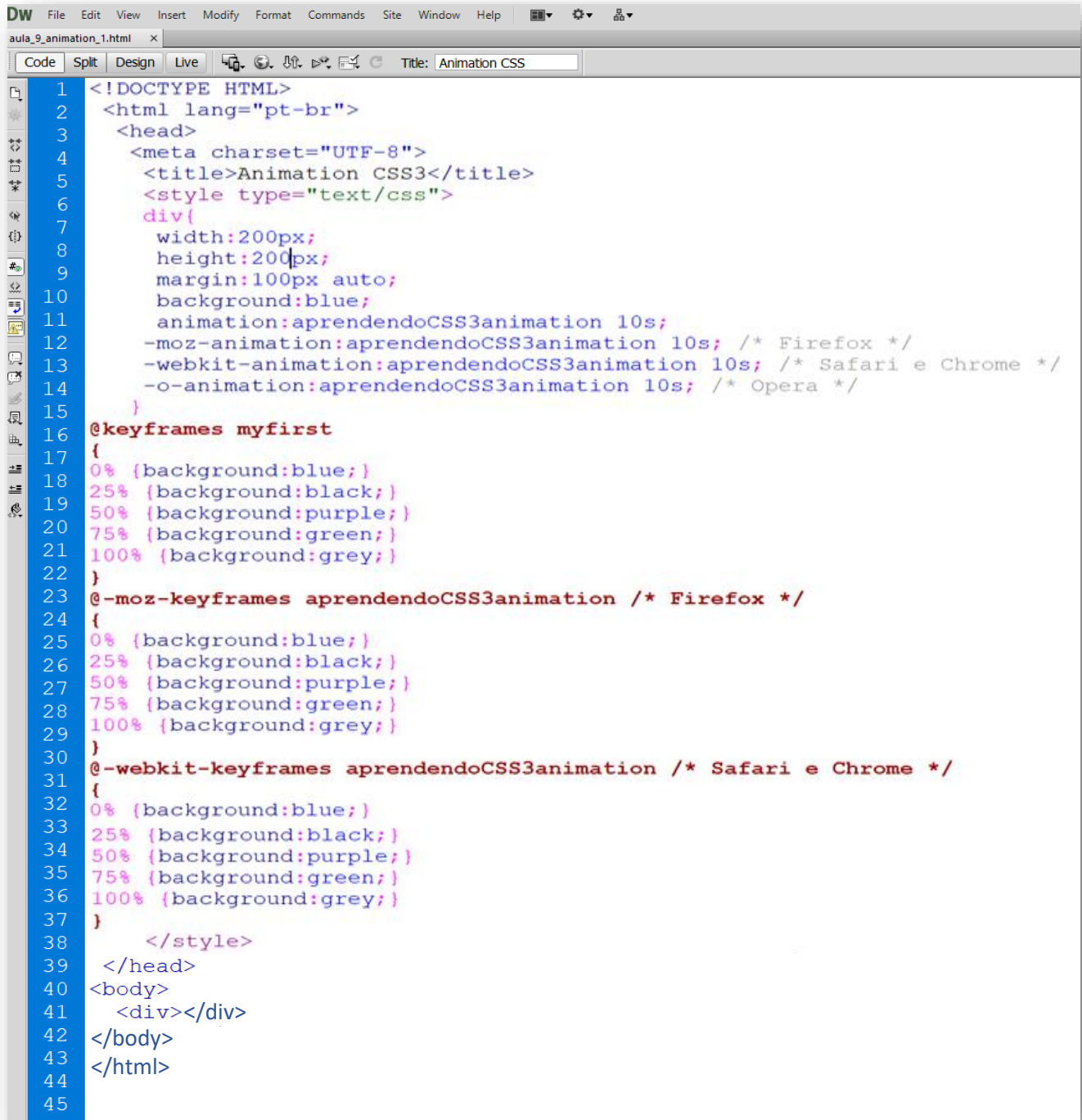
```
DW File Edit View Insert Modify Format Commands Site Window Help
aula_9_animation_1.html x
Code Split Design Live Title: Animation CSS
1 <!DOCTYPE HTML>
2 <html lang="pt-br">
3 <head>
4 <meta charset="UTF-8">
5 <title>Animation CSS3</title>
6 <style type="text/css">
7   div{
8     width:200px;
9     height:200px;
10    margin:100px auto;
11    background:blue;
12    animation:aprendendoCSS3animation 10s;
13    -moz-animation:aprendendoCSS3animation 10s; /* Firefox */
14    -webkit-animation:aprendendoCSS3animation 10s; /* Safari e Chrome */
15    -o-animation:aprendendoCSS3animation 10s; /* Opera */
16  }
17  @keyframes myfirst
18  {
19    0% {background:blue;}
20    25% {background:black;}
21    50% {background:purple;}
22    75% {background:green;}
23    100% {background:grey;}
24  }
25  @-moz-keyframes aprendendoCSS3animation /* Firefox */
26  {
27    0% {background:blue;}
28    25% {background:black;}
29    50% {background:purple;}
30    75% {background:green;}
31    100% {background:grey;}
32  }
33  @-webkit-keyframes aprendendoCSS3animation /* Safari e Chrome */
34  {
35    0% {background:blue;}
36    25% {background:black;}
37    50% {background:purple;}
38    75% {background:green;}
39    100% {background:grey;}
40  }
41  </style>
42 </head>
43 <body>
44   <div></div>
45 </body>
</html>
```


Salvando e abrindo no navegador vai ficar assim:



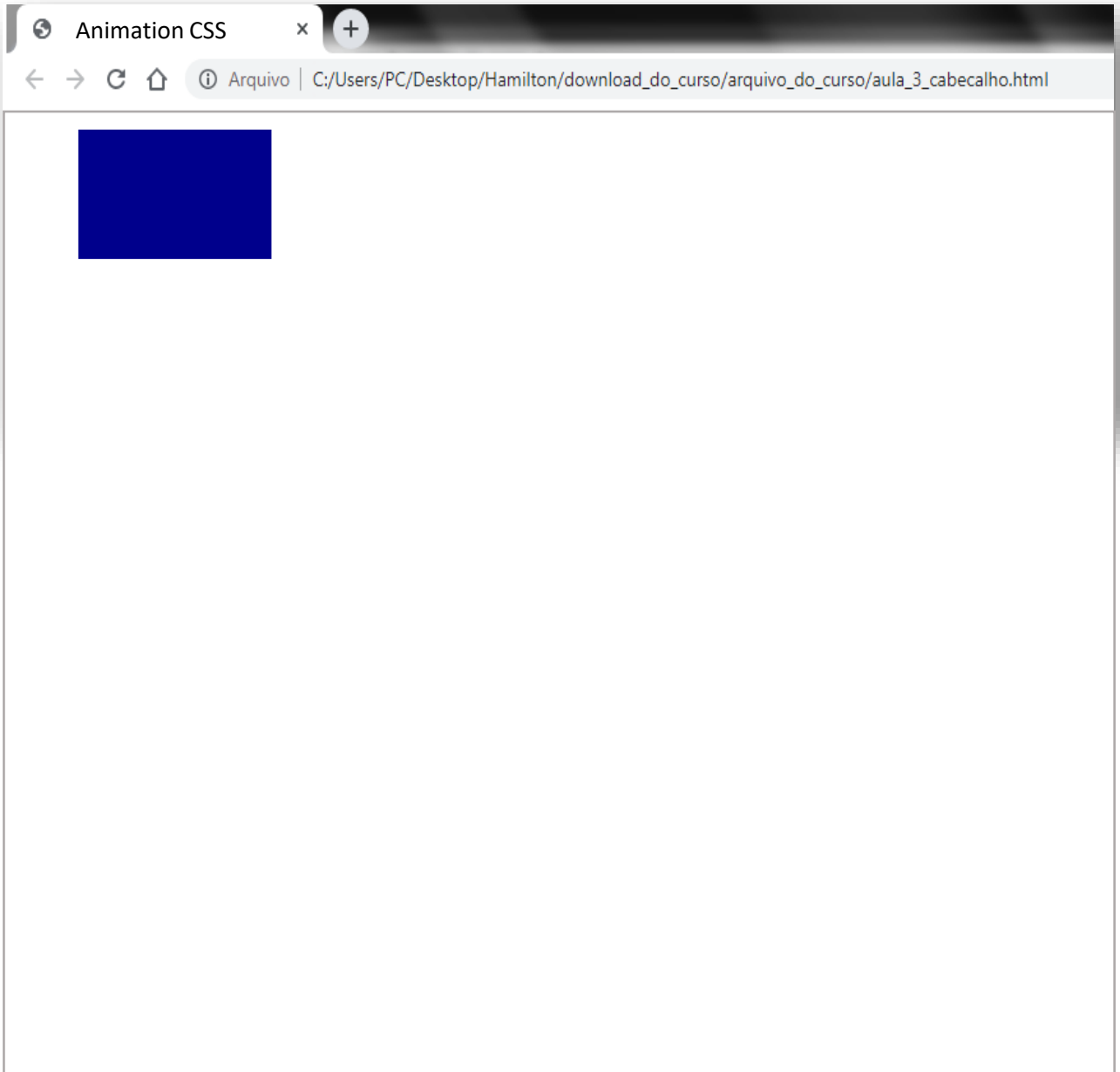
Podemos também fazer o movimento de animação em locais diferentes e em momentos diferentes, como mostra o código a seguir.

Exemplo-3:



```
1 <!DOCTYPE HTML>
2 <html lang="pt-br">
3   <head>
4     <meta charset="UTF-8">
5     <title>Animation CSS3</title>
6     <style type="text/css">
7       div{
8         width:200px;
9         height:200px;
10        margin:100px auto;
11        background:blue;
12        animation:aprendendoCSS3animation 10s;
13        -moz-animation:aprendendoCSS3animation 10s; /* Firefox */
14        -webkit-animation:aprendendoCSS3animation 10s; /* Safari e Chrome */
15        -o-animation:aprendendoCSS3animation 10s; /* Opera */
16      }
17      @keyframes myfirst
18      {
19        0% {background:blue;}
20        25% {background:black;}
21        50% {background:purple;}
22        75% {background:green;}
23        100% {background:grey;}
24      }
25      @-moz-keyframes aprendendoCSS3animation /* Firefox */
26      {
27        0% {background:blue;}
28        25% {background:black;}
29        50% {background:purple;}
30        75% {background:green;}
31        100% {background:grey;}
32      }
33      @-webkit-keyframes aprendendoCSS3animation /* Safari e Chrome */
34      {
35        0% {background:blue;}
36        25% {background:black;}
37        50% {background:purple;}
38        75% {background:green;}
39        100% {background:grey;}
40      }
41    </style>
42  </head>
43  <body>
44    <div></div>
45  </body>
46 </html>
```

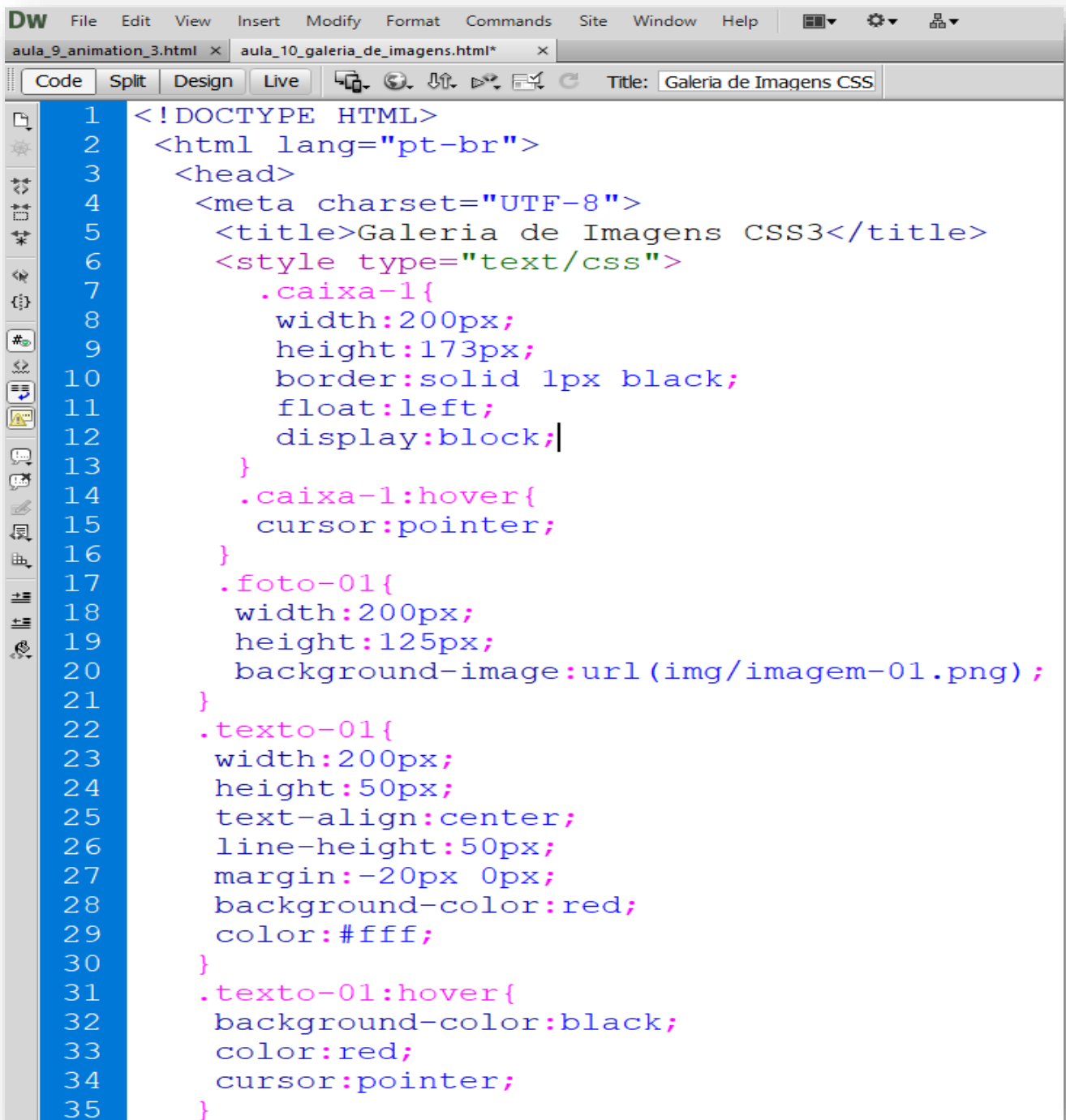
Salvando e abrindo no navegador vai ficar assim:



Quando a animação estiver finalizada, irá voltar ao estado original inicial.

Galeria de imagens CSS3

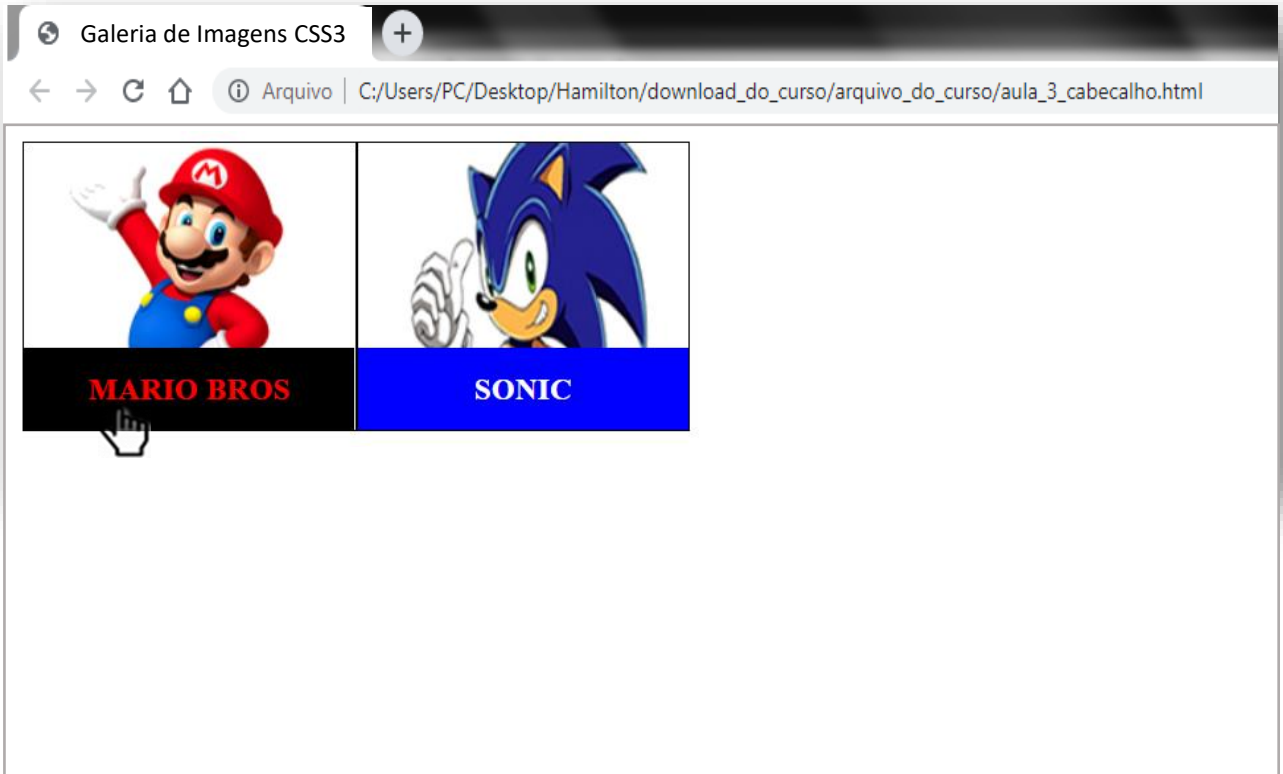
Veja como é bem simples também criar uma galeria de imagens com o **CSS3**:



```
1 <!DOCTYPE HTML>
2 <html lang="pt-br">
3 <head>
4 <meta charset="UTF-8">
5 <title>Galeria de Imagens CSS3</title>
6 <style type="text/css">
7   .caixa-1{
8     width:200px;
9     height:173px;
10    border:solid 1px black;
11    float:left;
12    display:block;|
13  }
14  .caixa-1:hover{
15    cursor:pointer;
16  }
17  .foto-01{
18    width:200px;
19    height:125px;
20    background-image:url (img/imagen-01.png) ;
21  }
22  .texto-01{
23    width:200px;
24    height:50px;
25    text-align:center;
26    line-height:50px;
27    margin:-20px 0px;
28    background-color:red;
29    color:#fff;
30  }
31  .texto-01:hover{
32    background-color:black;
33    color:red;
34    cursor:pointer;
35  }
```

```
36     .caixa-2{
37         width:200px;
38         height:173px;
39         border:solid 1px black;
40         float:left;
41         display:block;
42     }
43     .caixa-2:hover{
44         cursor:pointer;
45     }
46     .foto-02{
47         width:200px;
48         height:125px;
49         background-image:url (img/imagem-02.png) ;
50     }
51     .texto-02{
52         width:200px;
53         height:50px;
54         text-align:center;
55         line-height:50px;
56         margin:-20px 0px;
57         background-color:blue;
58         color:#fff;
59     }
60     .texto-02:hover{
61         background-color:black;
62         color:red;
63         cursor:pointer;
64     }
65 </style>
66 </head>
67 <body>
68     <div class="caixa-1">
69         <div class="foto-01"></div>
70         <div class="texto-01">
71             <h3>MARIO BROS</h3>
72         </div>
73     </div>
74     <div class="caixa-2">
75         <div class="foto-02"></div>
76         <div class="texto-02">
77             <h3>SONIC</h3>
78         </div>
79     </div>
80 </body>
81 </html>
```

Salvando e abrindo no navegador vai ficar assim:



Nota: Para acrescentar mais imagens na galeria você pode copiar o código das divs caixa-1 e caixa-2, substituir os arquivos: caixa-1 e caixa-2 por caixa-3 e caixa-4, substituir os textos e as imagens também e depois colar abaixo das divs já existentes no documento HTML. Para estilizar o processo é mesmo, basta substituir os arquivos também.

Obs: Lembre-se, para substituir os as imagens, você deve acrescentar as imagens na pasta img.

BOTÕES CSS3

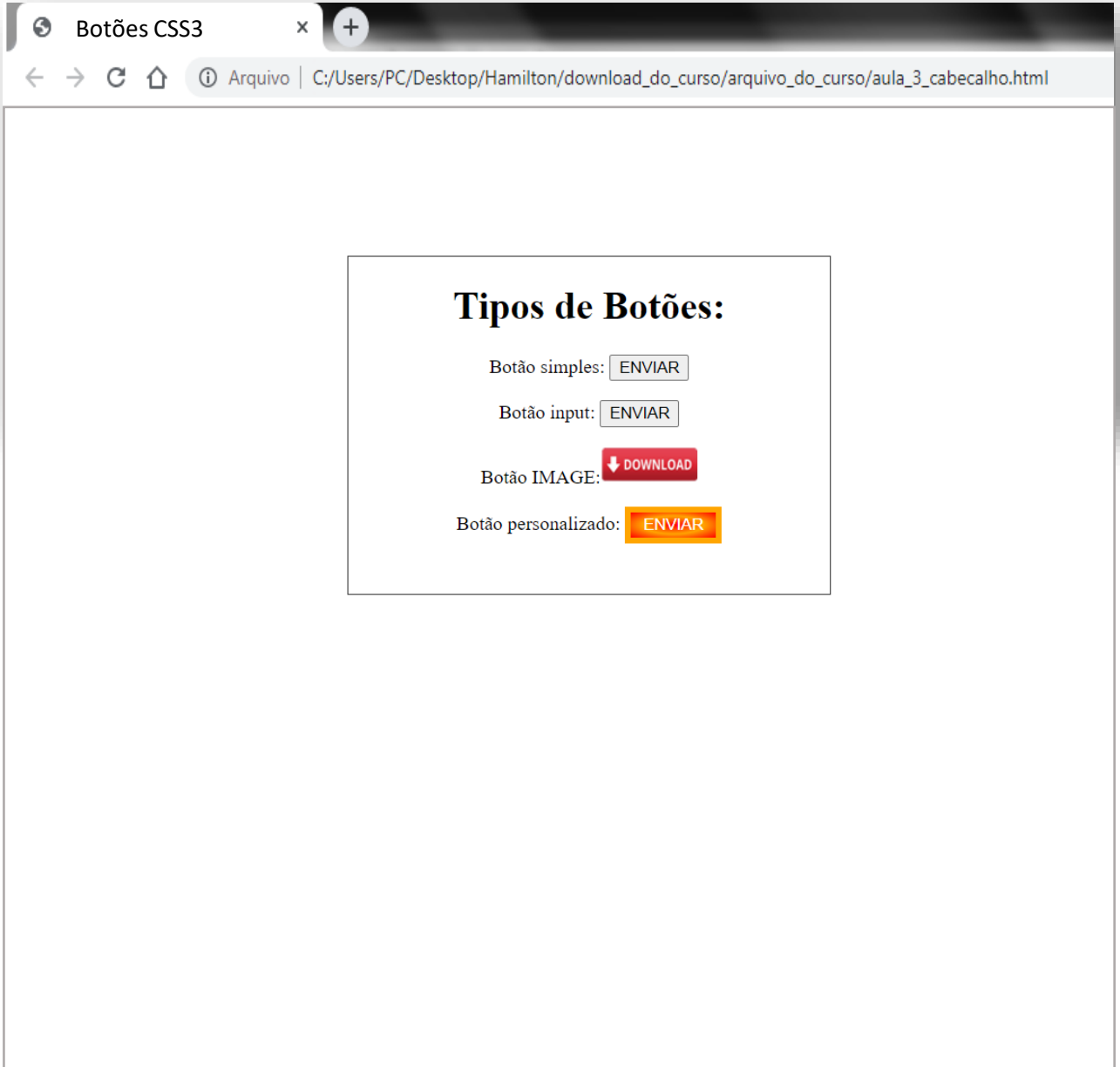
Veja também com é simples e divertido criar botões com a **CSS3** e deixa-los com aparência bem elegante:

```

1 <!DOCTYPE HTML>
2 <html lang="pt-br">
3 <head>
4 <meta charset="UTF-8">
5 <title>Botões CSS3</title>
6 <style type="text/css">
7   .personalizado{
8     width:80px;
9     height:30px;
10    cursor:pointer;
11    background-image:radial-gradient(red, orange, red);
12    color:#FFF;
13    border:5px orange solid;
14  }
15  .personalizado:hover{
16    color:#000;
17    font-weight:bolder;
18  }
19  .box{
20    border:#333 1px solid;
21    width:400px;
22    height:280px;
23    margin:100px auto;
24    padding:0px 10px;
25    box-sizing:border-box;
26    text-align:center;
27  }
28 </style>
29 </head>
30 <body>
31 <div class="box">
32 <h1>Tipos de Botões:</h1>
33   Botão simples:
34   <button type="submit">ENVIAR</button>
35   <p>Botão input:
36   <input type="button" value="ENVIAR" /></p>
37   <p>Botão IMAGE:<input type="image" src="img/btn.png" value="ENVIAR"/></p>
38   <p>Botão personalizado:
39   <input class="personalizado" type="button" value="ENVIAR" /></p>
40 </div>
41 </body>
42 </html>

```

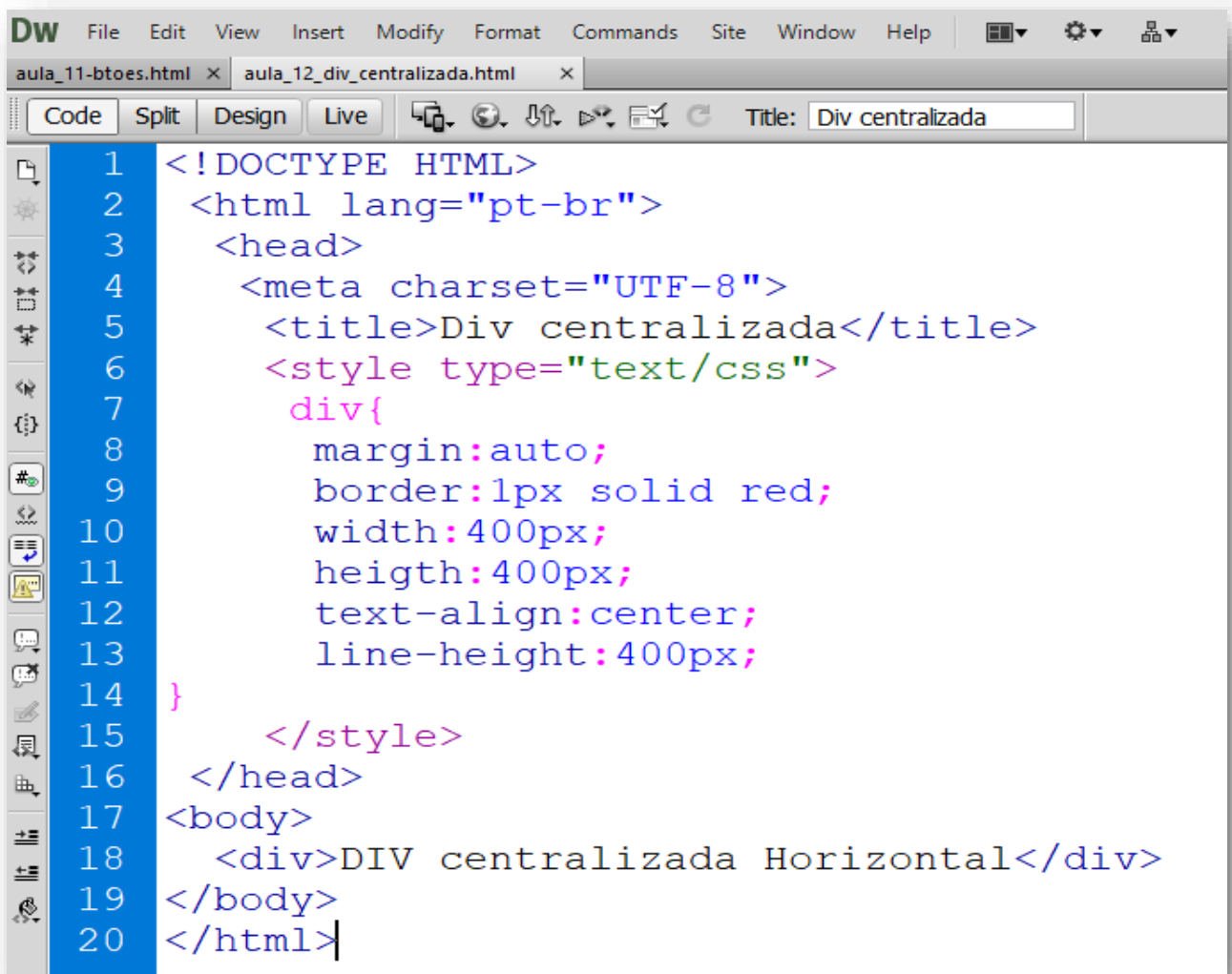
Salvando e abrindo no navegador vai ficar assim:



Centralizando elementos na página

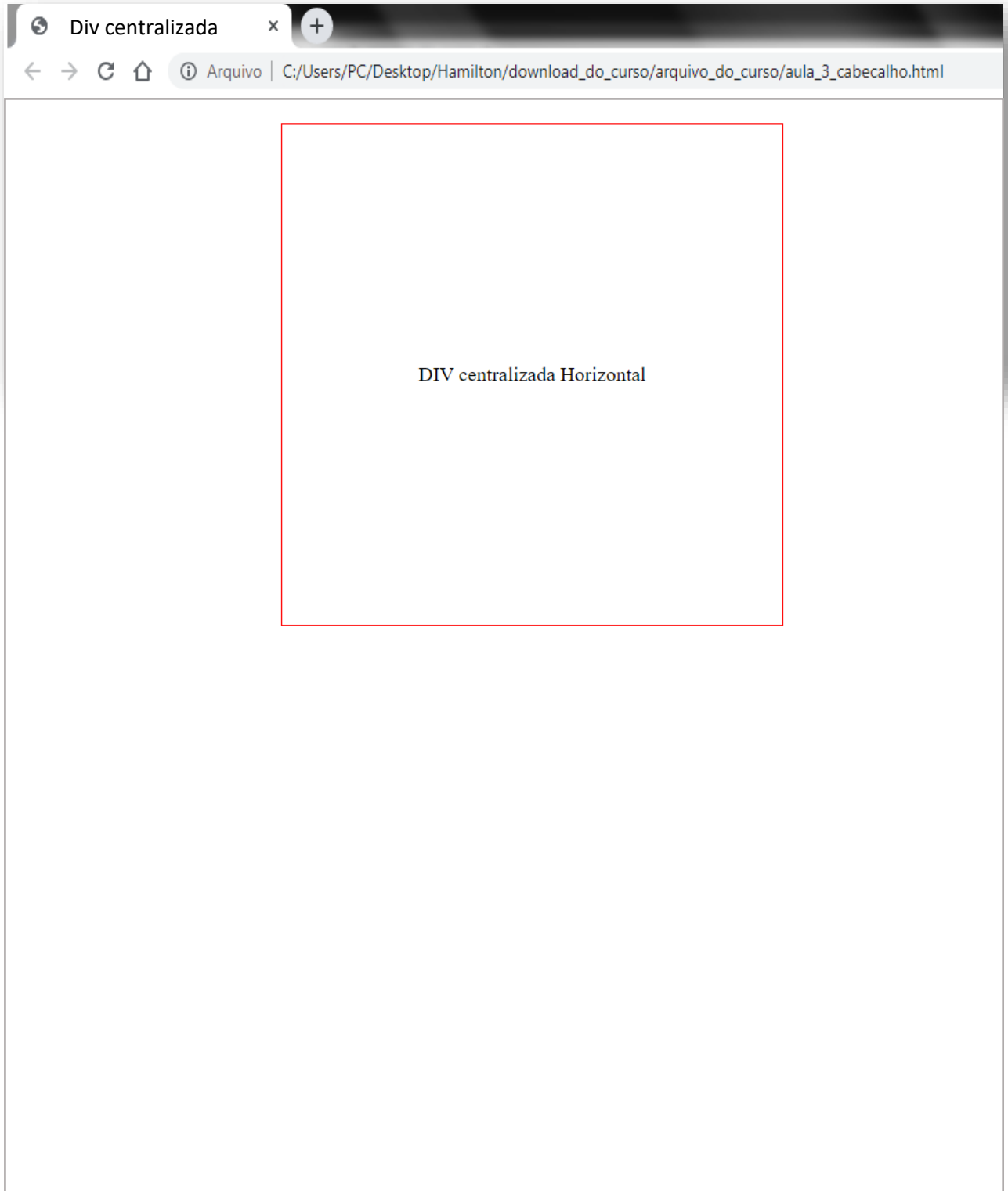
Vamos aprender agora um conceito muito importante: Como centralizar elementos na página. Vamos começar centralizando uma div na horizontal. Preste bem atenção na sintaxe: Para que os elementos fiquem na página de horizontal, o processo é bem simples. Basta deixar com que as margens do elemento fiquem automática.

Vamos ver como isso se comporta na prática:

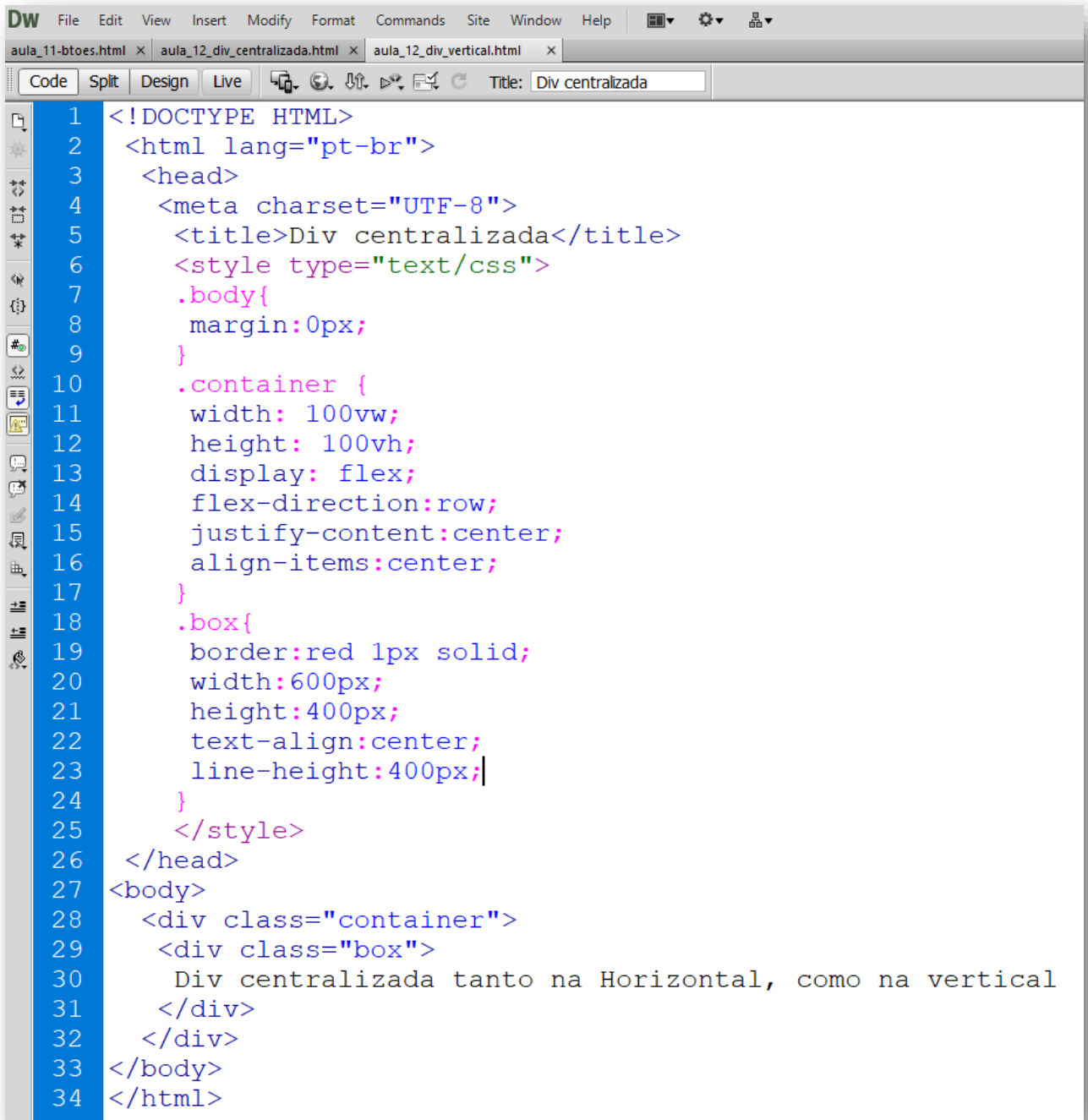


```
1 <!DOCTYPE HTML>
2 <html lang="pt-br">
3 <head>
4 <meta charset="UTF-8">
5 <title>Div centralizada</title>
6 <style type="text/css">
7   div{
8     margin:auto;
9     border:1px solid red;
10    width:400px;
11    height:400px;
12    text-align:center;
13    line-height:400px;
14  }
15 </style>
16 </head>
17 <body>
18   <div>DIV centralizada Horizontal</div>
19 </body>
20 </html>
```

Salvando e abrindo no navegador vai ficar assim:

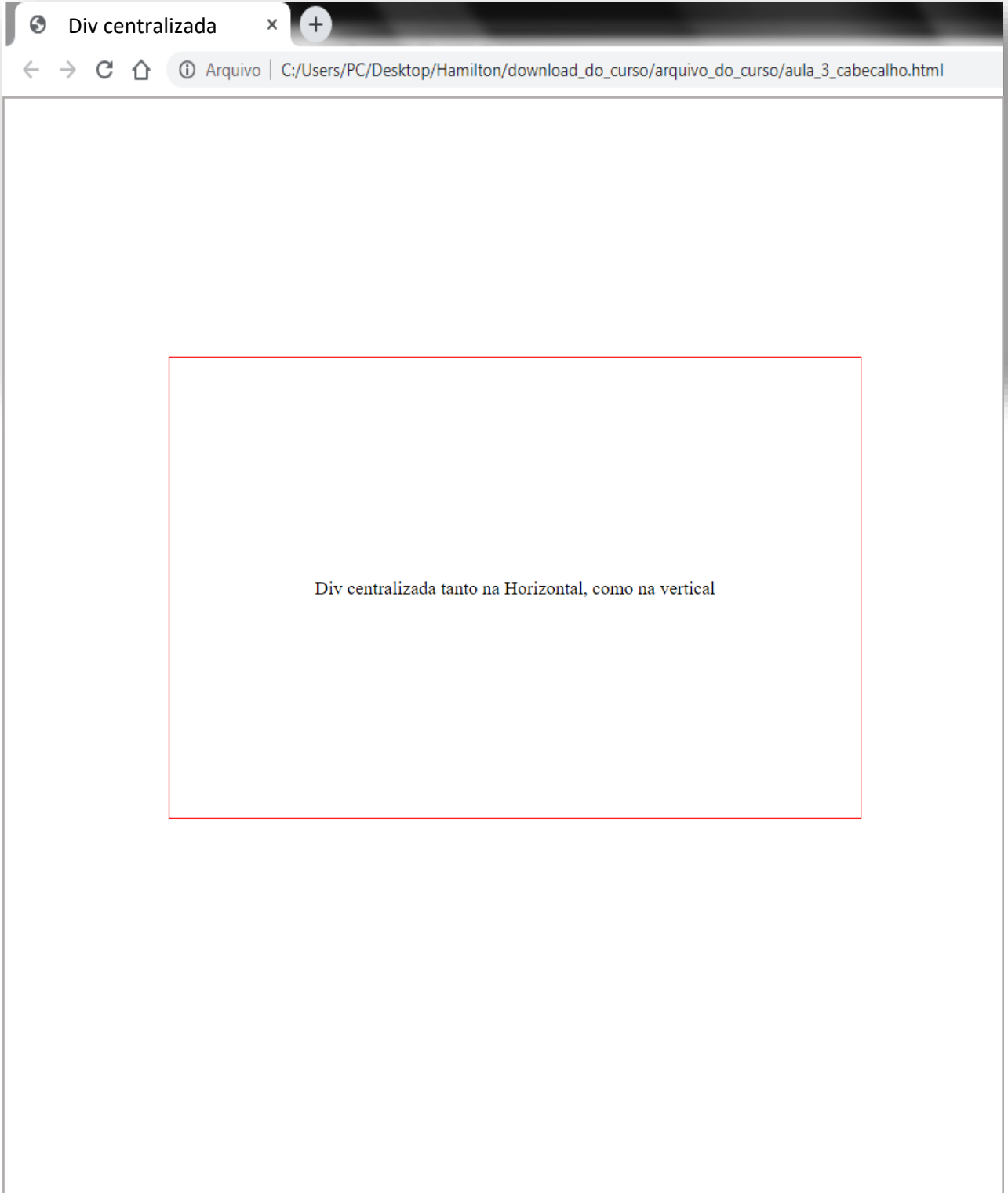


Perceba que a div esta centralizada na horizontal conforme estilizamos na folha de estilo. Porém também queremos que a div fique centralizada tanto na vertical quanto na horizontal, como esta o seu conteúdo. É exatamente isso que faremos agora: Preste atenção que agora a sintaxe é diferente:



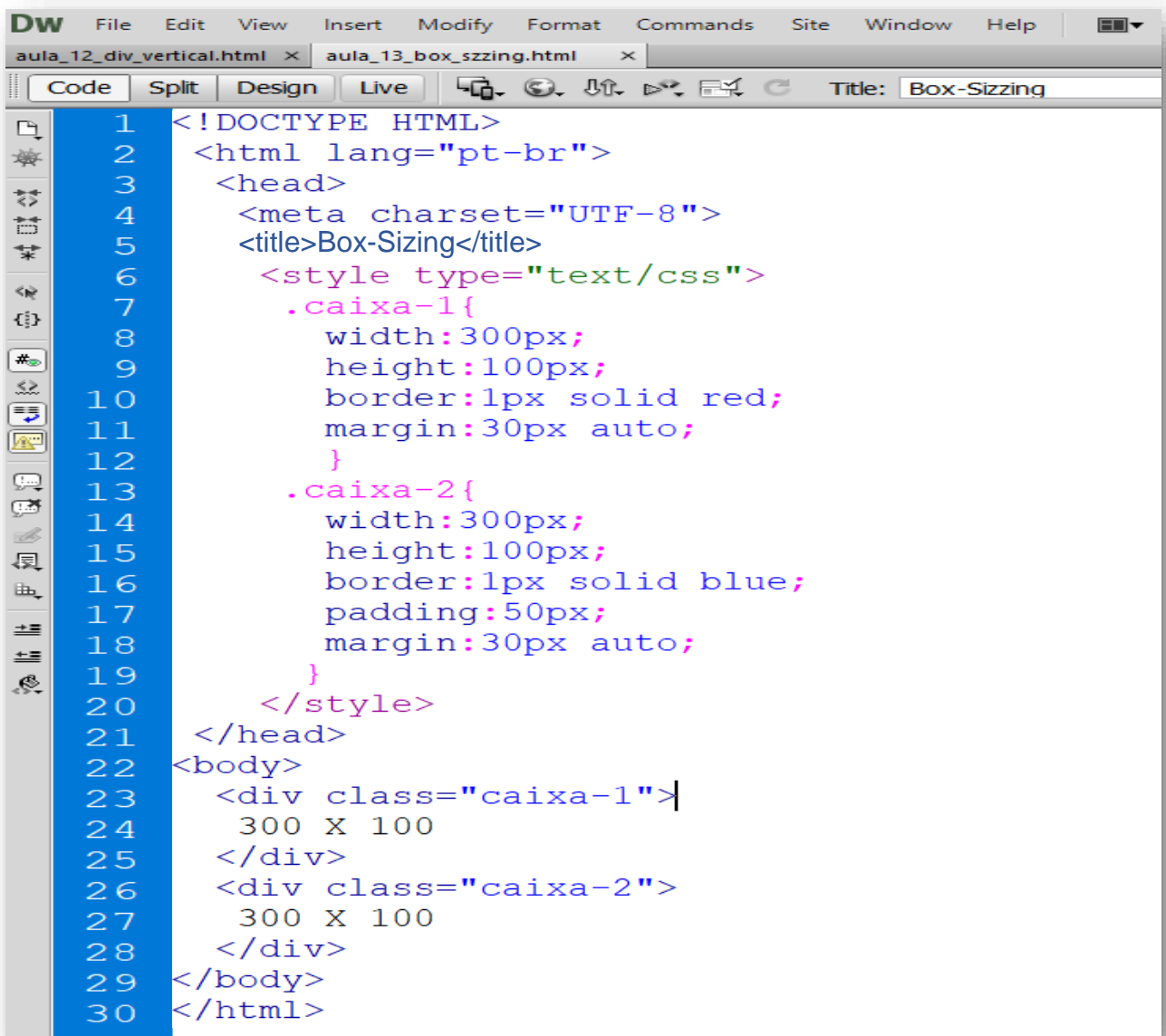
```
1 <!DOCTYPE HTML>
2 <html lang="pt-br">
3 <head>
4 <meta charset="UTF-8">
5 <title>Div centralizada</title>
6 <style type="text/css">
7 .body{
8   margin:0px;
9 }
10 .container {
11   width: 100vw;
12   height: 100vh;
13   display: flex;
14   flex-direction:row;
15   justify-content:center;
16   align-items:center;
17 }
18 .box{
19   border:red 1px solid;
20   width:600px;
21   height:400px;
22   text-align:center;
23   line-height:400px;|
24 }
25 </style>
26 </head>
27 <body>
28 <div class="container">
29 <div class="box">
30   Div centralizada tanto na Horizontal, como na vertical
31 </div>
32 </div>
33 </body>
34 </html>
```

Salvando e abrindo no navegador vai ficar assim:



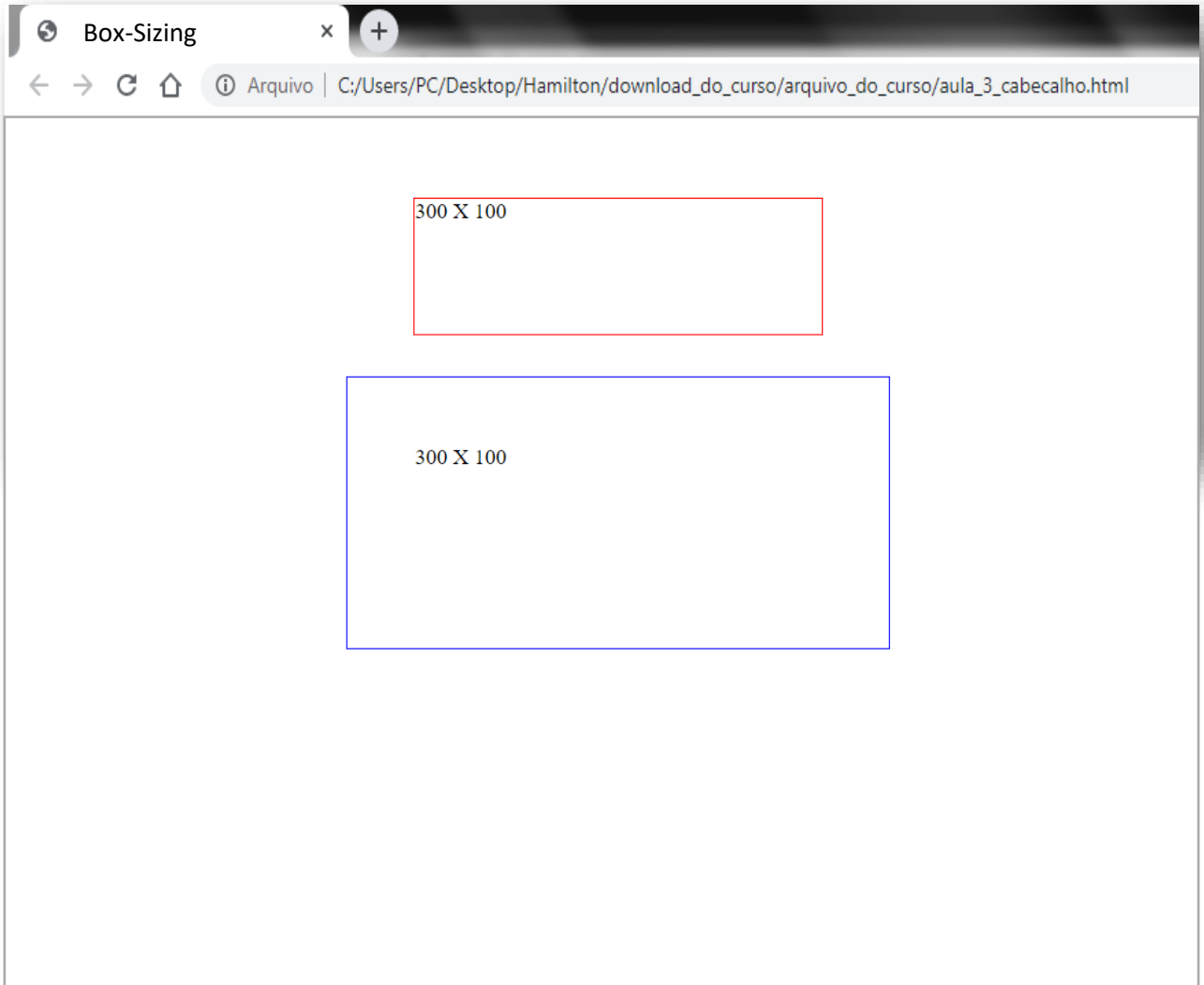
Atributo Box-Sizing

O atributo `box-sizing` (**Dimensão da caixa**) propriedade **CSS** nos permite incluir o preenchimento e a borda na largura e altura totais de um elemento. Por padrão, a largura e a altura de um elemento são calculadas da seguinte maneira: $\text{width} + \text{padding} + \text{border} = \text{largura real de um elemento}$ e $\text{height} + \text{padding} + \text{border} = \text{altura real de um elemento}$. Quando você define a largura / altura de um elemento, o elemento geralmente aparece maior do que o definido (porque a borda e o preenchimento do elemento são adicionados à largura / altura especificada do elemento). Vamos citar aqui um exemplo com duas divs de mesmo tamanho. Preste bem atenção na sintaxe:



```
1 <!DOCTYPE HTML>
2 <html lang="pt-br">
3   <head>
4     <meta charset="UTF-8">
5     <title>Box-Sizing</title>
6     <style type="text/css">
7       .caixa-1{
8         width:300px;
9         height:100px;
10        border:1px solid red;
11        margin:30px auto;
12      }
13      .caixa-2{
14        width:300px;
15        height:100px;
16        border:1px solid blue;
17        padding:50px;
18        margin:30px auto;
19      }
20    </style>
21  </head>
22  <body>
23    <div class="caixa-1">
24      300 X 100
25    </div>
26    <div class="caixa-2">
27      300 X 100
28    </div>
29  </body>
30 </html>
```

Salvando e abrindo no navegador vai ficar assim:



Ao salvar e visualizar no navegador/Browser perceba que temos um problema: Uma grande diferença nas dimensões das caixas, sendo que suas larguras e alturas são as mesmas.

Solucionando o problema:

O atributo **box-sizing** nos permite incluir o preenchimento e a borda na largura e altura totais de um elemento. Basta definir no **box-sizing** o elemento **border-box**.

Preenchimento e borda estão incluídos na largura e altura: Vamos ver como se aplica isso na pratica:

```

DW  File  Edit  View  Insert  Modify  Format  Commands  Site  Window  Help
aula_13_box_szzing_border_box.html  x
Code  Split  Design  Live  Title: Border-Box
1  <!DOCTYPE HTML>
2  <html lang="pt-br">
3  <head>
4  <meta charset="UTF-8">
5  <title>Border-Box</title>
6  <style type="text/css">
7  .caixa-1{
8  width:300px;
9  height:100px;
10 border:1px solid red;
11 margin:30px auto;
12 box-sizing:border-box;
13
14 }
15 .caixa-2{
16 width:300px;
17 height:100px;
18 border:1px solid blue;
19 padding:50px;
20 margin:30px auto;
21 box-sizing:border-box;
22 }
23 </style>
24 </head>
25 <body>
26 <div class="caixa-1">
27 300 X 100
28 </div>
29 <div class="caixa-2">
30 As divs estão ambas do mesmo tamanho
31 </div>
32 </body>
33 </html>

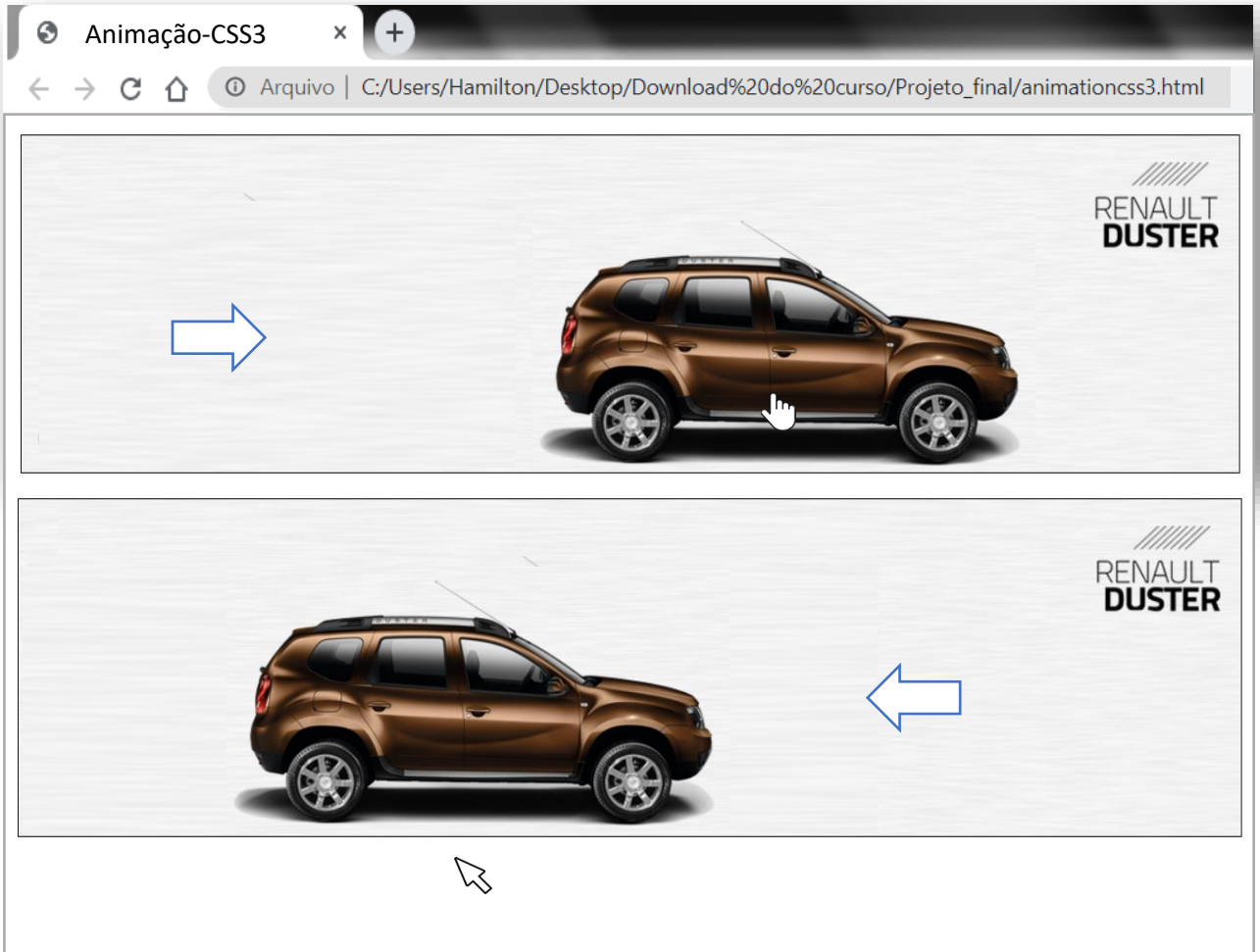
```

Salvando e abrindo no navegador vai ficar assim:



Animação CSS3 – Projeto Final

Para finalizar o curso CSS3 para iniciantes vamos fazer uma animação com o carro Renault Duster. A ideia é: Quando o mouse se sobrepõe na imagem, o carro começa a andar. Ao sobrepõe o mouse fora da imagem, o carro começa andar de ré.



Vamos lá, então: Na pasta download do curso, abra a pasta Projeto_final. Abra o arquivo index.html para ver a animação. Perfeito, agora vamos começar a animação do zero. Abra a pasta arquivos_do_usuario. Abra o arquivo index.html e veja que a pagina esta vazia, pois não inserimos nenhum elemento no documento HTML.



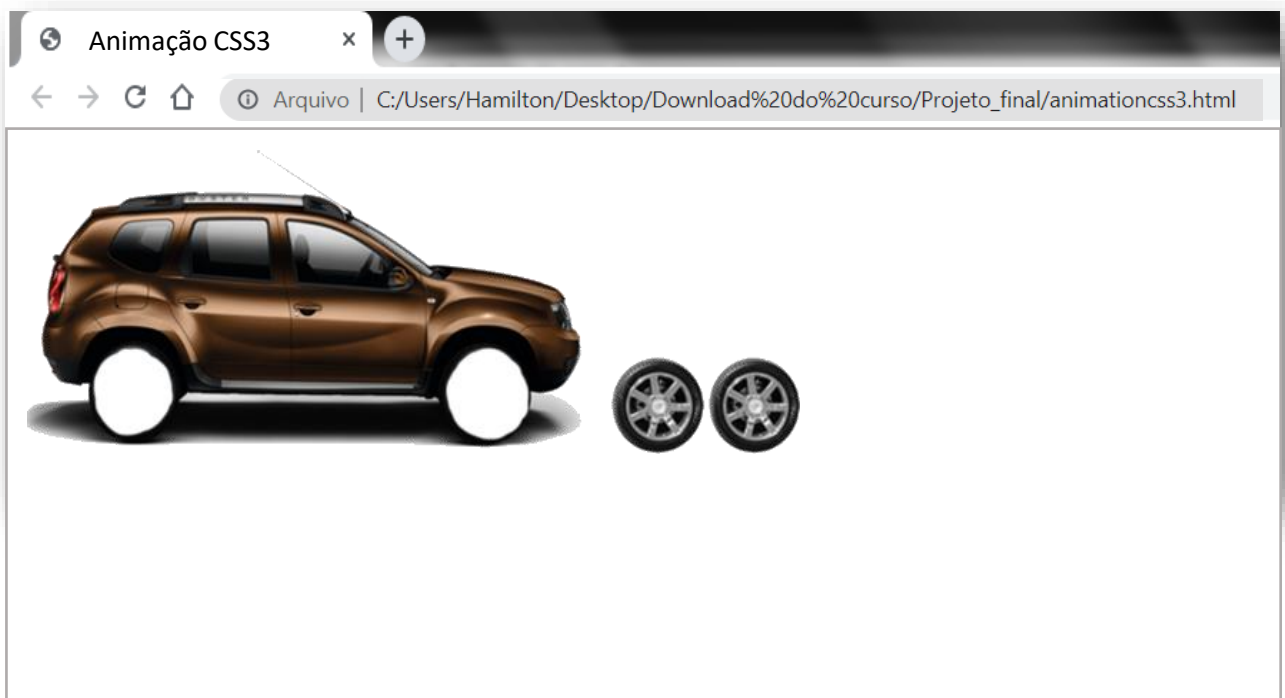
A screenshot of the Adobe Dreamweaver interface. The top menu bar includes 'DW', 'File', 'Edit', 'View', 'Insert', 'Modify', 'Format', 'Commands', 'Site', 'Window', and 'Help'. The title bar shows 'animationcss3.html'. The 'Code' tab is active, displaying the following HTML code:

```
1 <!doctype html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Animação CSS</title>
6     <link rel="stylesheet" href="css/estilo.css">
7   </head>
8   <body>
9     <div>
10
11   </div>
12 </body>
13 </html>
```

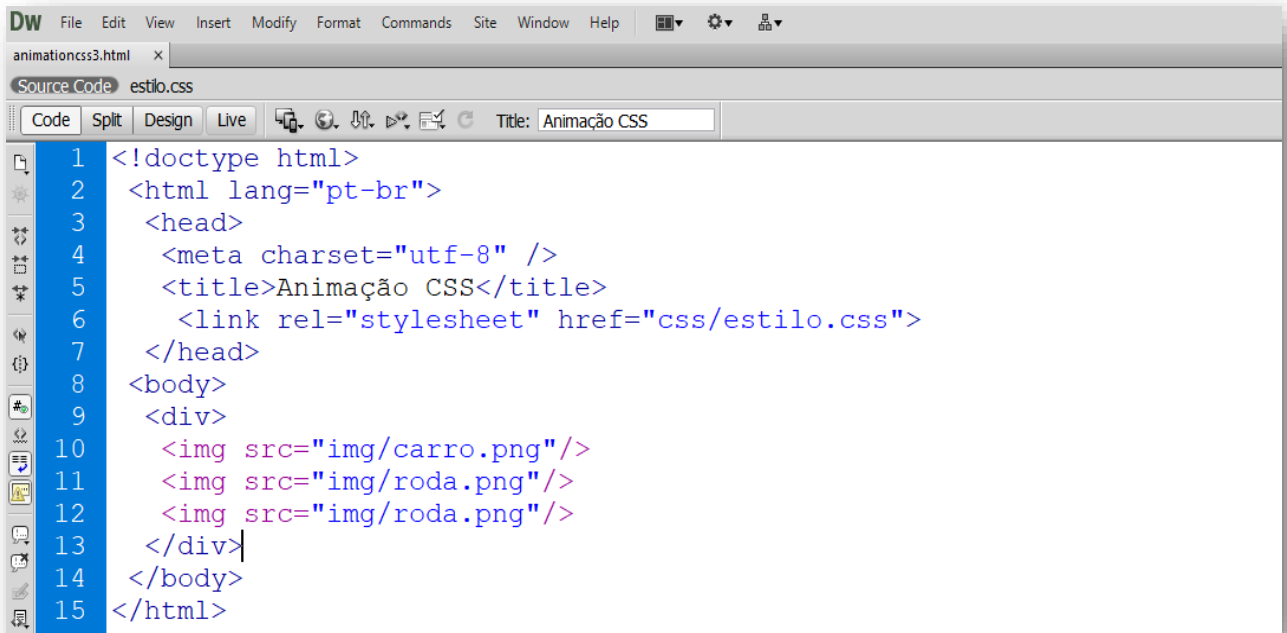
Com o arquivo aberto no software editor de código insira os elementos no documento. Faça conforme a descrição abaixo:

```
DW File Edit View Insert Modify Format Commands Site Window Help
animationcss3.html* x
Code Split Design Live Title: Animação CSS
1 <!doctype html>
2 <html lang="pt-br">
3 <head>
4 <meta charset="utf-8" />
5 <title>Animação CSS</title>
6
7 </head>
8 <body>
9 <div>
10 
11 
12 |
13 </div>
14 </body>
15 </html>
```

Salve o documento e abra no navegador para ver como esta ficando:

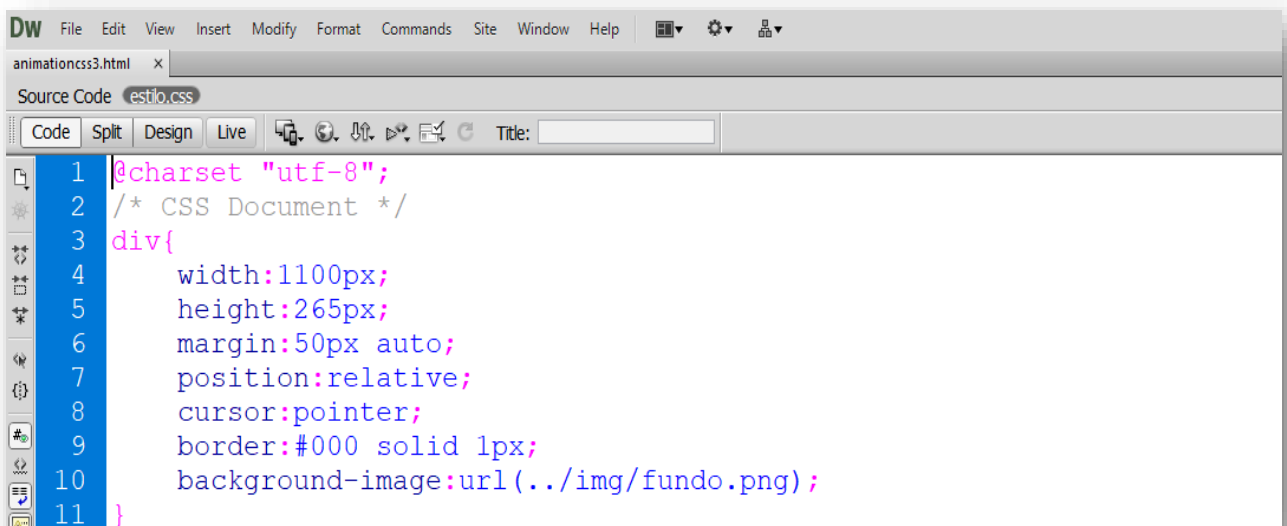


O próximo passo é linkar o documento HTML a folha de estilos, pois iremos utilizar o CSS externo. Vamos lá então. Faça conforme a descrição abaixo:



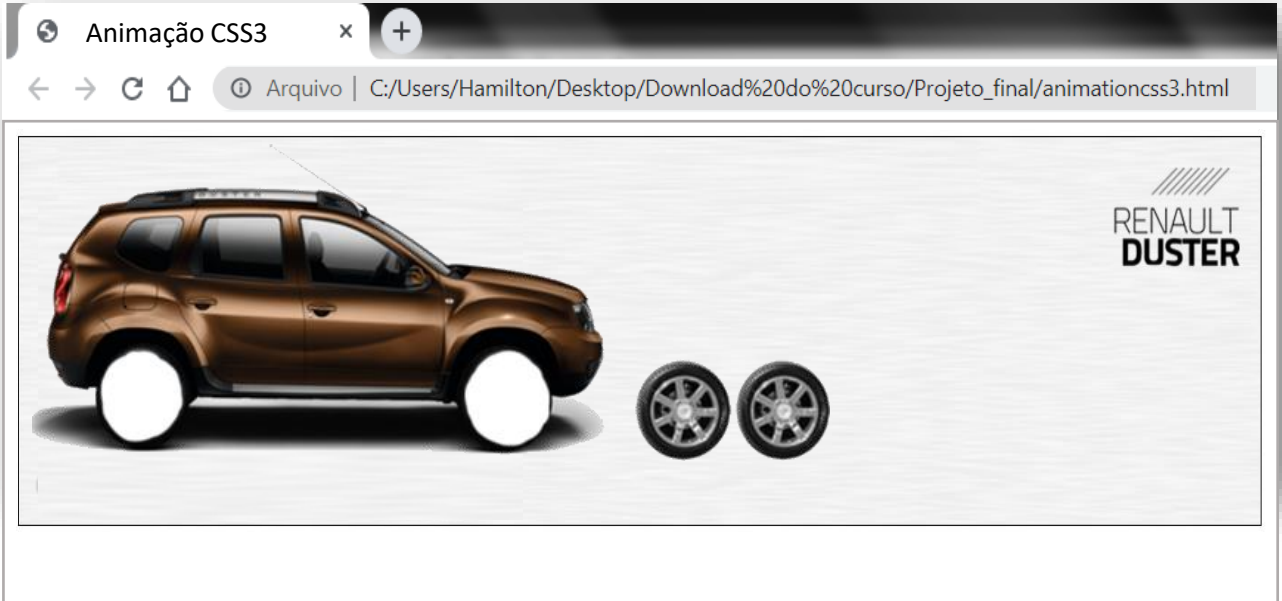
```
1 <!doctype html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8" />
5     <title>Animação CSS</title>
6     <link rel="stylesheet" href="css/estilo.css">
7   </head>
8   <body>
9     <div>
10    
11    
12    
13  </div>
14 </body>
15 </html>
```

Perfeito, agora vamos estilizar os elementos na folha de estilos. Vamos começar estilizando a div para que fique com um fundo elegante. Digite na folha de estilos conforme a descrição abaixo:



```
1 @charset "utf-8";
2 /* CSS Document */
3 div{
4   width:1100px;
5   height:265px;
6   margin:50px auto;
7   position:relative;
8   cursor:pointer;
9   border:#000 solid 1px;
10  background-image:url(..img/fundo.png);
11 }
```

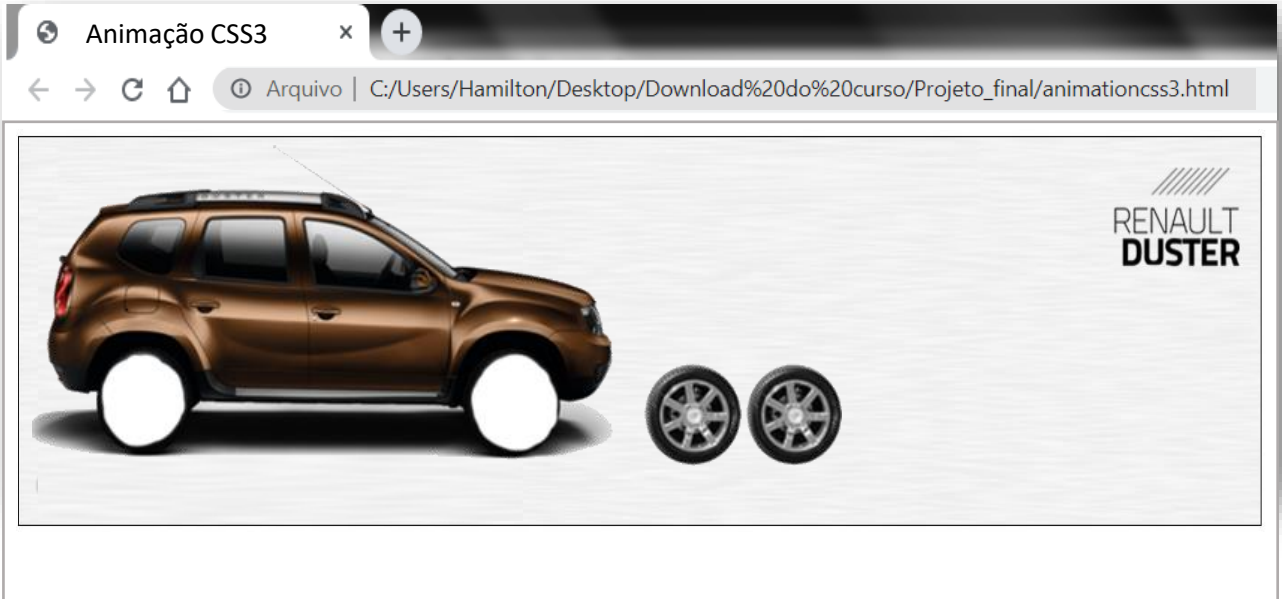
Salve o documento e abra no navegador para ver como esta ficando:



Veja que temos três imagens que compõem a animação, para deixar a imagem pronta para rodar a animação faremos um pequeno ajuste nas três imagens ao mesmo tempo. Vamos fazer com que a imagem do carro e as duas rodas comecem a movimentar com uma duração de 3 segundos ao sobrepor o mouse sobre a imagem. Para fazer isso é muito simples: Basta aplicar o efeito **trransition:all 3s ease-in-out;** (que significa: Transição para todas as entradas). Faça conforme a descrição abaixo:

```
DW File Edit View Insert Modify Format Commands Site Window Help
animationcss3.html x
Source Code estilo.css*
Code Split Design Live
1 @charset "utf-8";
2 /* CSS Document */
3 div{
4     width:1100px;
5     height:265px;
6     margin:50px auto;
7     position:relative;
8     cursor:pointer;
9     border:#000 solid 1px;
10    background-image:url(../img/fundo.png);
11 }
12 div img{
13     trransition:all 3s ease-in-out;
14     -moz-transition:all 3s ease-in-out; /* para rodar no mozilla */
15     -webkit-transition:all 3s ease-in-out; /* para rodar no ggole e safari */
16     -o-transition:all 3s ease-in-out; /* para rodar no opera */
17     -ms-transition:all 3s ease-in-out; /* para rodar no explorer */
18 }
```

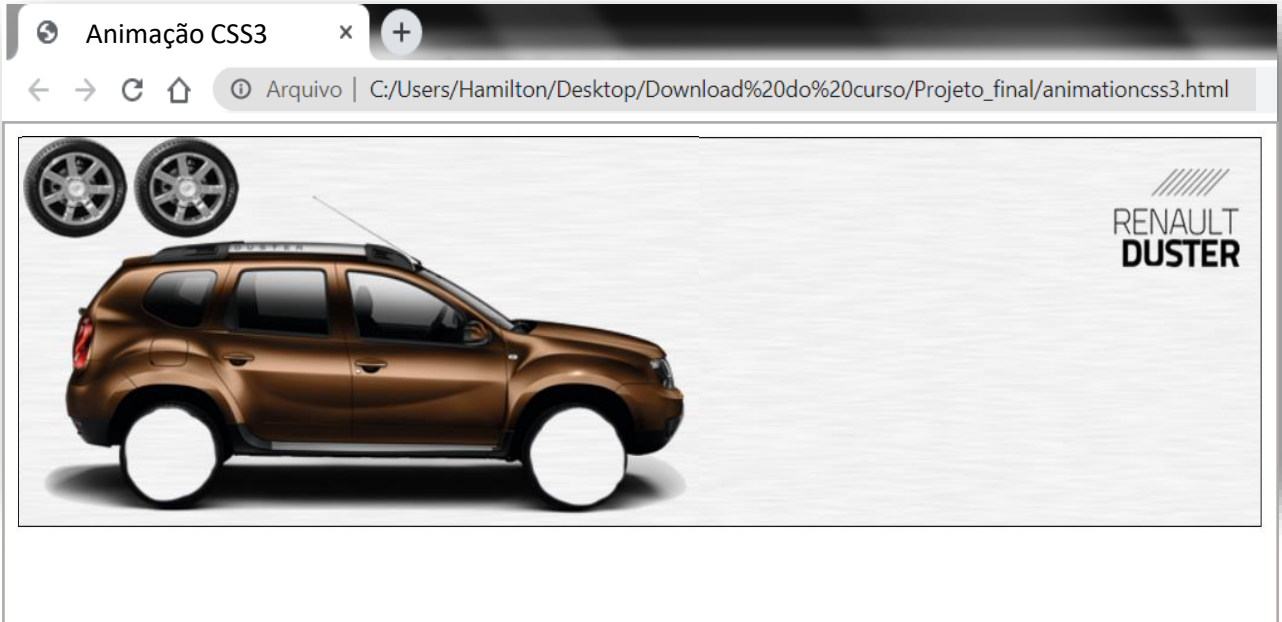
Ao salvar o documento e abrir no navegador, nada acontece. Isso porque ainda não estilizamos os elementos que compõem a animação.



Faremos isso agora, vamos começar com a imagem do carro - **img:nth-of-type(1)**
Digite na folha de estilos conforme a descrição abaixo:

```
DW File Edit View Insert Modify Format Commands Site Window Help
animationcss3.html
Source Code estilo.css
Code Split Design Live
1 @charset "utf-8";
2 /* CSS Document */
3 div{
4     width:1100px;
5     height:265px;
6     margin:50px auto;
7     position:relative;
8     cursor:pointer;
9     border:#000 solid 1px;
10    background-image:url(../img/fundo.png);
11 }
12 div img{
13     transition:all 3s ease-in-out;
14     -moz-transition:all 3s ease-in-out; /* para rodar no mozilla */
15     -webkit-transition:all 3s ease-in-out; /* para rodar no ggole e safari */
16     -o-transition:all 3s ease-in-out; /* para rodar no opera */
17     -ms-transition:all 3s ease-in-out; /* para rodar no explorer */
18 }
19 div img:nth-of-type(1){
20     position:absolute;
21     top:30px;
22     left:0px;
23 }
```

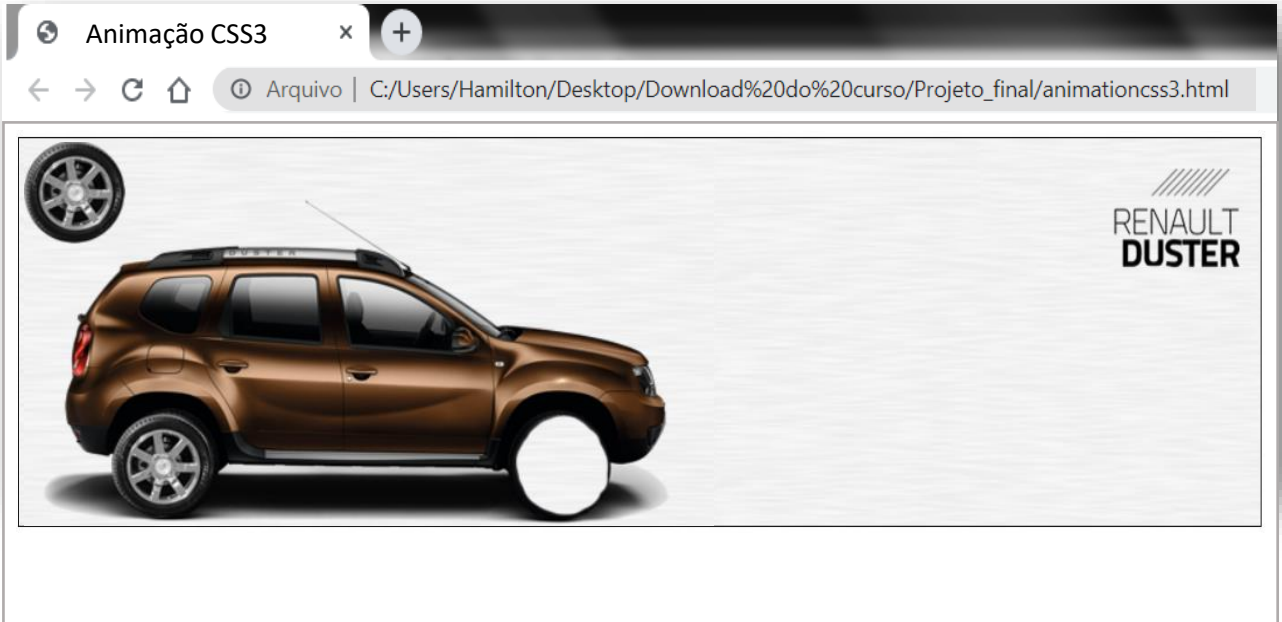
Salve e veja no navegador para ver com fica:



Agora vamos estilizar a imagem da roda direita - **img:nth-of-type(2)**:

```
DW File Edit View Insert Modify Format Commands Site Window Help
animationcss3.html x
Source Code estilo.css*
Code Split Design Live
1 @charset "utf-8";
2 /* CSS Document */
3 div{
4     width:1100px;
5     height:265px;
6     margin:50px auto;
7     position:relative;
8     cursor:pointer;
9     border:#000 solid 1px;
10    background-image:url(../img/fundo.png);
11 }
12 div img{
13     trransition:all 3s ease-in-out;
14     -moz-transition:all 3s ease-in-out; /* para rodar no mozilla */
15     -webkit-transition:all 3s ease-in-out; /* para rodar no google e safari */
16     -o-transition:all 3s ease-in-out; /* para rodar no opera */
17     -ms-transition:all 3s ease-in-out; /* para rodar no explorer */
18 }
19 div img:nth-of-type(1){ div img:nth-of-type(2){
20     position:absolute;         position:absolute;|
21     top:30px;                 top:185px;
22     left:0px;                 left:60px;
23 }
```

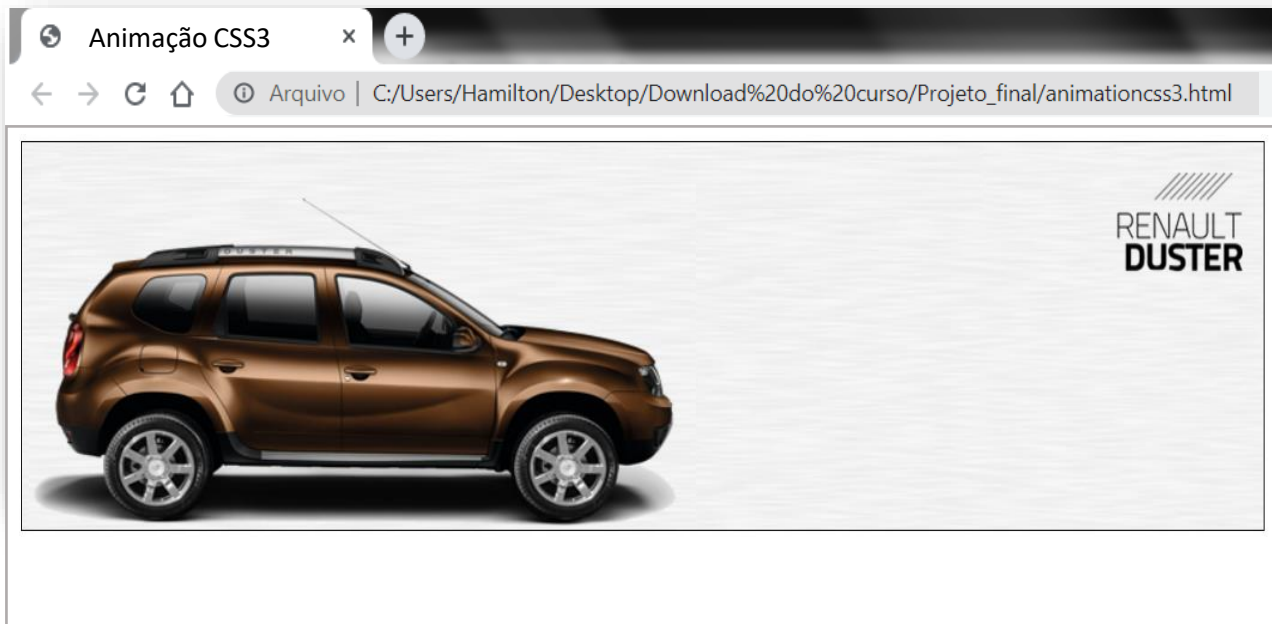
Salve e veja no navegador para ver com fica:



Agora vamos estilizar a imagem da roda esquerda - **img:nth-of-type(3)**:

```
DW File Edit View Insert Modify Format Commands Site Window Help
animationcss3.html x
Source Code estilo.css*
Code Split Design Live
1 @charset "utf-8";
2 /* CSS Document */
3 div{
4     width:1100px;
5     height:265px;
6     margin:50px auto;
7     position:relative;
8     cursor:pointer;
9     border:#000 solid 1px;
10    background-image:url(../img/fundo.png);
11 }
12 div img{
13     transition:all 3s ease-in-out;
14     -moz-transition:all 3s ease-in-out; /* para rodar no mozilla */
15     -webkit-transition:all 3s ease-in-out; /* para rodar no google e safari */
16     -o-transition:all 3s ease-in-out; /* para rodar no opera */
17     -ms-transition:all 3s ease-in-out; /* para rodar no explorer */
18 }
19 div img:nth-of-type(1){ position:absolute; }
20 div img:nth-of-type(2){ position:absolute; }
21 div img:nth-of-type(3){ position:absolute;
22     top:30px; top:185px; top:185px;
23     left:0px; left:60px; left:335px;
}
```


Salve e veja no navegador para ver com fica:



E assim terminamos a montagem da imagem do carro. O que faremos agora é dar um outro efeito na imagem para que o carro possa se movimentar conforme o mouse estiver sobreposto na imagem.

Vamos utilizar o efeito: **CSS transform:translate() rotate();**

A função **CSS translate()** reposiciona um elemento na direção horizontal e/ou vertical. O seu resultado é um tipo de dado `<transform-function>`. Esta transformação é caracterizada por um vetor bidimensional. Suas coordenadas definem o quanto o elemento se move em cada direção.

A função **CSS rotate()** define uma transformação que gira um elemento em torno de um ponto fixo no plano 2D, sem deformá-lo. O resultado é um tipo de dados `<transform-function>`. O eixo de rotação passa por uma origem, definido pela propriedade **CSS transform-origin** (en-US).

Vamos ver como isso se comporta na pratica. Faça conforme a descrição baixo:

```

1 @charset "utf-8";
2 /* CSS Document */
3
4 div{
5     width:1100px;
6     height:265px;
7     margin:50px auto;
8     position:relative;
9     cursor:pointer;
10    border:#000 solid 1px;
11    background-image:url(../img/fundo.png);
12
13 }
14
15 div img{
16    trransition:all 3s ease-in-out;
17    -moz-transition: 3s ease-in-out; /* para rodar no mozilla */
18    -webkit-transition:all 3s ease-in-out; /* para rodar no google e safari */
19    -o-transition:all 3s ease-in-out; /* para rodar no opera */
20    -ms-transition:all 3s ease-in-out; /* para rodar no explorer */
21
22 }
23
24 div img:nth-of-type(1){          div img:nth-of-type(2){          }
25    position:absolute;            position:absolute;|          div img:nth-of-type(3){
26    top:30px;                     top:185px;                    position:absolute;
27    left:0px;                     left:60px;                     top:185px;
28                                     left:335px;
29 }                                     }
30
31 div:hover img:nth-of-type(1){
32     transform:translate(600px,0px) rotate(0deg);
33     -moz-transition:translate(600px,0px) rotate(0deg); /* mozilla */
34     -webkit-transition:translate(600px,0px) rotate(0deg); /* google e safari */
35     -o-transition:translate(600px,0px) rotate(0deg); /* opera */
36     -ms-transition:translate(600px,0px) rotate(0deg); /* explore */
37 }
38
39 div:hover img{
40     transform:translate(600px,0px) rotate(360deg);
41     -moz-transition:translate(600px,0px) rotate(360deg); /* mozilla */
42     -webkit-transition:translate(600px,0px) rotate(360deg); /* google e safari */
43     -o-transition:translate(600px,0px) rotate(360deg); /* opera */
44     -ms-transition:translate(600px,0px) rotate(360deg); /* explore */
45 }
46
47

```

Chegamos ao final de mais uma etapa. Estou muito feliz por ter chegado até a etapa final deste módulo.

Parabenizo você, pois apesar das demandas do dia a dia, permaneceu até o **final!** Percebo que esse **curso** ultrapassou minhas expectativas, pois tive você, que abraçou a causa e trouxe valiosas contribuições, deixando a sua essência. Isso é o que encontro de mais valioso nesse **curso!**

Um forte abraço e te aguardo na próxima etapa!

Porque é importante aprender CSS?

Aprender CSS permite atuar em diversos estilos de projetos, como no desenvolvimento de sites para web, aplicativos online ou mesmo em conjunto com programadores e designers para criação de temas, templates e outras soluções visuais para sites.

O que exatamente o CSS faz?

CSS3 é um dos elementos mais importantes para o desenvolvimento de páginas web, o que faz com que ela seja uma linguagem fundamental para quem pretende trabalhar nessa área. CSS3 permite atuar em diversos estilos de projetos, como no desenvolvimento de sites para web, aplicativos online ou mesmo em conjunto com programadores e designers para criação de temas, templates e outras soluções visuais para sites.

Como está o mercado para quem aprende CSS?

Se você deseja desempenhar atividades como desenvolvimento web e aprender CSS3, essa é uma excelente hora para começar a estudar. Isso porque o mercado possui uma grande demanda por programadores que tenham conhecimentos sólidos na linguagem. Como a linguagem é utilizada nos mais diversos tipos de aplicações web, esse é um conhecimento que agrega grande valor profissional.

