



Exercícios de CSS

1. Métodos para adicionar CSS

Neste exercício, você praticará a adição de CSS a um arquivo HTML usando todos os três métodos: CSS externo, CSS interno e CSS embutido. Você só deve usar seletores de tipo para este exercício ao adicionar estilos por meio dos métodos externo e interno. Você também deve usar palavras-chave para cores (por exemplo, "azul") em vez de usar valores RGB ou HEX.

Existem três elementos para você adicionar estilos, cada um dos quais usa um método diferente de adicionar CSS a ele, conforme observado na imagem de resultado abaixo. Todos os outros exercícios desta seção terão um arquivo CSS fornecido e vinculado para você, mas para este exercício você terá que criar o arquivo e vinculá-lo no arquivo HTML. Trata-se de praticar o uso desses métodos diferentes e obter a sintaxe correta.

As propriedades que você precisa adicionar a cada elemento são:

div: fundo vermelho, texto branco, tamanho de fonte de 32px, alinhamento central e negrito;

p: fundo verde, texto branco e tamanho de fonte de 18px;

botão: fundo laranja e tamanho de fonte de 18px.

dica rápida:

Não se preocupe com detalhes nestes exercícios que não são especificamente mencionados na seção de exercícios ou autoverificação. Como os resultados desejados são capturas de tela, seu navegador pode mostrar uma fonte diferente, as cores podem parecer diferentes em sua máquina ou o espaçamento entre os elementos pode parecer diferente. Preocupe-se apenas com os itens específicos que você deveria aprender para cada exercício.

Resultado desejado:



Style me via the external method!

I would like to be styled with the internal method, please.

Inline Method

index.html

```
<div>Style me via the external method!</div>
<p>I would like to be styled with the internal method,
please.</p>
<button>Inline Method</button>
```

Autoverificação

- Você usou todos os três métodos de adicionar CSS a um arquivo HTML?
- Você vinculou corretamente o arquivo CSS externo no arquivo HTML?
- O elemento div tem CSS adicionado por meio do método externo?
- O elemento p tem CSS adicionado por meio do método interno?
- O elemento de botão tem CSS adicionado por meio do método inline?

2. Seletores de classe e ID

Saber como adicionar atributos de classe e ID a elementos HTML, bem como usar seus respectivos seletores, é inestimável. É importante praticar o uso deles.

Existem vários elementos no arquivo HTML fornecido, aos quais você terá que adicionar atributos de classe ou ID, conforme observado na imagem de resultado



abaixo. Em seguida, você terá que adicionar regras no arquivo CSS fornecido usando a sintaxe correta do seletor. Examine a imagem do resultado com cuidado e tente ter em mente quais elementos têm estilo semelhante (classes), quais podem ser completamente únicos do resto (ID) e quais têm pequenas variações de outros (múltiplas classes).

Não é totalmente importante quais valores de classe ou ID você usa, pois o foco aqui é poder adicionar os atributos e usar a sintaxe correta do seletor para estilizar elementos. Para as cores neste exercício, tente usar um valor que não seja de palavra-chave (RGB, HEX ou HSL). As propriedades que você precisa adicionar a cada elemento são:

Todos os elementos numerados ímpares: um fundo vermelho/rosa claro e uma lista de fontes contendo Verdana e DejaVu Sans com sans-serif como substituto

O segundo elemento: texto azul e tamanho de fonte de 36px

O terceiro elemento: além dos estilos para todos os elementos ímpares, adicione um tamanho de fonte de 24px

O quarto elemento: um fundo verde claro, um tamanho de fonte de 24px e negrito

Dica rápida: no VS Code, você pode alterar quais cores de formato são exibidas em RGB, HEX ou HSL passando o mouse sobre o valor da cor no CSS e clicando na parte superior do pop-up que aparece!

Observação: Parte de sua tarefa é adicionar uma fonte a alguns desses itens. Não se preocupe com a fonte do resto deles. A fonte padrão do seu navegador pode ser diferente da exibida e tudo bem para este exercício.

Resultado desejado:



Number 1 - I'm a class!

Number 2 - I'm one ID.

Number 3 - I'm a class, but cooler!

Number 4 - I'm another ID.

Number 5 - I'm a class!

index.html

```
<p>Number 1 - I'm a class!</p>
<div>Number 2 - I'm one ID.</div>
<p>Number 3 - I'm a class, but cooler!</p>
<div>Number 4 - I'm another ID.</div>
<p>Number 5 - I'm a class!</p>
```

3. Seletores de Agrupamento

Vamos construir um pouco do exercício anterior, no qual você (espero) adicionou várias classes a um único elemento para aplicar duas regras diferentes a ele.

Em vez de ter um único elemento com duas regras diferentes aplicadas, você dará a dois elementos um nome de classe exclusivo e, em seguida, adicionará regras para estilos que ambos os elementos compartilham, bem como seus próprios estilos exclusivos. Certifique-se de dar uma boa olhada na imagem de resultado abaixo para ver exatamente o que é único em cada elemento e o que ambos os elementos têm em comum.

Isso ajudará você a praticar ainda mais a adição de classes e o uso de seletores de classe, portanto, certifique-se de adicionar o atributo class no arquivo HTML. Para o restante desses exercícios, o formato de qualquer cor fica a seu critério; confiamos



que você praticará usando os diferentes valores! As propriedades que você precisa adicionar a cada elemento são:

- **O primeiro elemento:** um fundo preto e texto branco
- **O segundo elemento:** um fundo amarelo
- **Ambos os elementos:** um tamanho de fonte de 28px e uma lista de fontes contendo Helvetica e Times New Roman, com sans-serif como substituto

Resultado desejado:



index.html

```
<button>Click Me!</button>
<button>No, Click Me!</button>
```

Autoverificação:

- Cada elemento tem um nome de classe exclusivo?
- Você usou o seletor de agrupamento para estilos que ambos os elementos compartilham?
- Você fez regras separadas para os estilos exclusivos para cada elemento?

4. Ajuste o padding de um elemento

styles.css

```
.texto-injetado {
  margin-bottom: -25px;
  text-align: center;
}
```



```
.caixa {  
    border-style: solid;  
    border-color: black;  
    border-width: 5px;  
    text-align: center;  
}  
  
.caixa-amarela {  
    background-color: yellow;  
    padding: 10px;  
}  
  
.caixa-vermelha {  
    background-color: crimson;  
    color: #fff;  
    padding: 20px;  
}  
  
.caixa-azul {  
    background-color: blue;  
    color: #fff;  
    padding: 10px;  
}
```

index.html

```
<h5 class="texto-injetado">margin</h5>  
  
<div class="caixa caixa-amarela">  
    <h5 class="caixa caixa-vermelha">padding</h5>  
    <h5 class="caixa caixa-azul">padding</h5>  
</div>
```

5. Ajuste a margin de um elemento



style.css

```
.texto-injetado {  
    margin-bottom: -25px;  
    text-align: center;  
}  
  
.caixa {  
    border-style: solid;  
    border-color: black;  
    border-width: 5px;  
    text-align: center;  
}  
  
.caixa-amarela {  
    background-color: yellow;  
    padding: 10px;  
}  
  
.caixa-vermelha {  
    background-color: crimson;  
    color: #fff;  
    padding: 20px;  
    margin: 20px;  
}  
  
.caixa-azul {  
    background-color: blue;  
    color: #fff;  
    padding: 20px;  
    margin: 10px;  
}
```

index.html

```
<h5 class="texto-injetado">margin</h5>
```

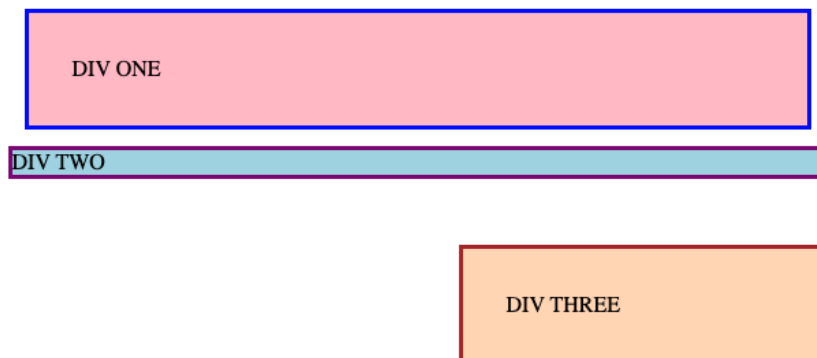


```
<div class="caixa caixa-amarela">
  <h5 class="caixa caixa-vermelha">padding</h5>
  <h5 class="caixa caixa-azul">padding</h5>
</div>
```

6. Prática 1 de margin e padding

Para este primeiro exercício, simplesmente edite o arquivo style.css para que as divs se pareçam com a imagem abaixo. Edite o CSS apenas onde instruído no arquivo. Você só deve alterar os valores da margem e do preenchimento para este exercício. Você não deve adicionar ou remover propriedades no CSS ou tocar no HTML.

Resultado desejado:



style.css

```
body {
  max-width: 600px;
  margin: 0 auto;
}

.one {
  background: pink;
  border: 3px solid blue;
  /* ALTERE-ME */
  padding: 0px;
```




```
margin: 0px;
}

.two {
  background: lightblue;
  border: 3px solid purple;
  /* ALTERE-ME */
  margin-bottom: 0px;
}

.three {
  background: peachpuff;
  border: 3px solid brown;
  width: 200px;
  /* ALTERE-ME */
  padding: 0px;
  margin-left: 0px;
}
```

index.html

```
<div class="one">
  DIV ONE
</div>
<div class="two">
  DIV TWO
</div>
<div class="three">
  DIV THREE
</div>
```



Autoverificação

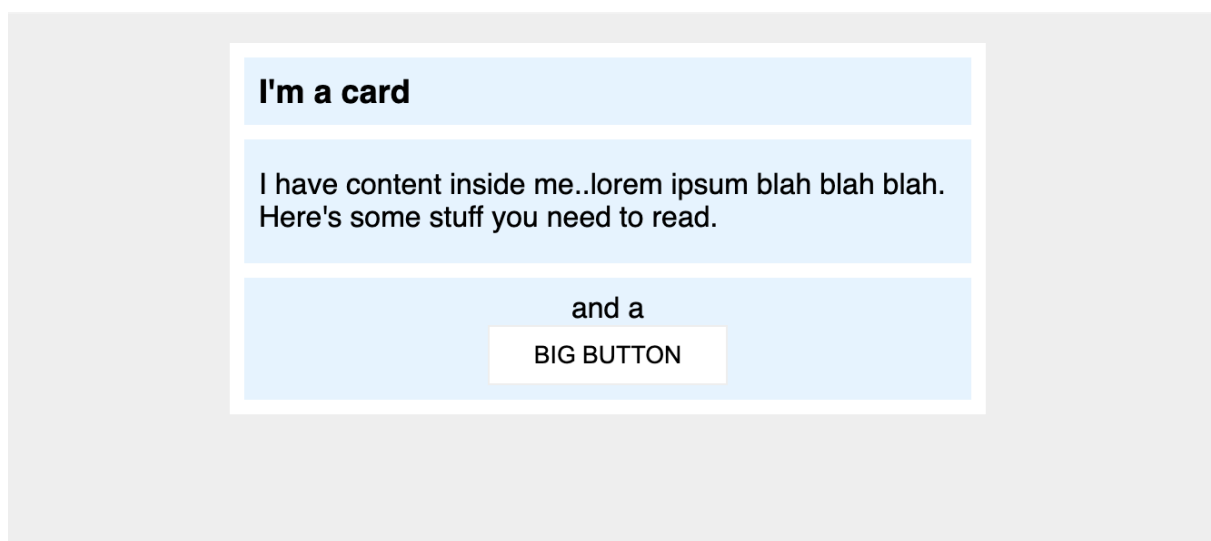
Use esta seção para verificar seu trabalho. Nesses projetos, seu objetivo não é atingir 100% de perfeição de pixel, mas usar as ferramentas que você aprendeu para chegar relativamente perto da saída desejada.

- Div One e Div Three têm 32px entre o texto e a borda.
- Div One tem 12px entre ele e qualquer outro elemento na página.
- Há uma lacuna de 48px entre a Div Two e a Div Three.
- A Div Three está alinhada à direita.
- O alinhamento da Div Three é obtido usando margin (e não float, flexbox, etc.).

7. Prática 2 de margin e padding

Este é um pouco mais bonito e um pouco mais próximo de algo que você pode ver no mundo real. Você precisará alterar um pouco mais do que apenas a margem e o preenchimento para que pareça exatamente correto.

Resultado desejado:





style.css

```
body {  
    background: #eee;  
    font-family: sans-serif;  
}  
  
.card {  
    width: 400px;  
    background: #fff;  
    margin: 16px auto;  
}  
  
.title {  
    background: #e3f4ff;  
}  
  
.content {  
    background: #e3f4ff;  
}  
  
.button-container {  
    background: #e3f4ff;  
}  
  
button {  
    background: white;  
    border: 1px solid #eee;  
}
```

index.html

```
<div class="card">  
    <h1 class="title">I'm a card</h1>
```



```
<div class="content">I have content inside me..lorem  
ipsum blah blah blah. Here's some stuff you need to  
read.</div>  
  
<div class="button-container">and a <button>BIG  
BUTTON</button></div>  
  
</div>
```

Autoverificação

Use esta seção para verificar seu trabalho. Nesses projetos, seu objetivo não é atingir 100% de perfeição de pixel, mas usar as ferramentas que você aprendeu para chegar relativamente perto da saída desejada.

- Há 8px entre a borda do cartão e seu conteúdo (as seções azuis).
- Há uma lacuna de 8px entre cada uma das seções azuis dentro do cartão.
- O título do cartão usa uma fonte de 16px.
- Há 8px entre o texto do título e a borda da seção do título.
- A seção de conteúdo tem espaço de 16px na parte superior e inferior e 8px em ambos os lados.
- Tudo dentro da seção *.button-container* é centralizado e há preenchimento de 8px.
- O Big Button está centrado em sua própria linha.
- O **Big Button** tem espaço de 24px nas laterais e 8px na parte superior e inferior.