

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>background-size</title>
<style>

    * {
        box-sizing: border-box;
        font-family: sans-serif;
        font-weight: bold;
    }

    body {
        margin: 0;
        padding: 1vw;
        width: 100vw;
        height: 100vh;
        display: flex;
        justify-content: space-between;
        align-items: center;
        flex-direction: column;
    }

    .container {
        width: 100%;
        height: 100%;
        padding: 5px;
        display: flex;
        justify-content: space-between;
        align-items: center;
    }

    .bloco {
```

```
width: 25%;
height: 100%;
padding: 2px;
margin: 4px;
box-shadow: 10px 10px 10px rgba(0, 0, 0, .2);
display: flex;
justify-content: center;
flex-direction: column;
}

.bloco > div {
  display: flex;
  justify-content: center;
  align-items: center;
  width: 100%;
  height: 80%;
  background-image: url("https://picsum.photos/1400/1000");
  /*Essa é uma imagem, que toda vez que é atualizada traz uma imgame diferente*/
}

#auto {
  background-size: auto;
}

#length {
  background-size: 100% 33%;
}

#cover {
  background-size: cover;
}

#contain {
  background-size: contain;
}
```

```
</style>
</head>
<body>

<!--
```

Propriedade: background-size (Tamanho do Fundo)

Objetivo: Essa propriedade define o tamanho que uma imagem de fundo pode ter sobre um elemento HTML, podemos usar qualquer tamanho relativo ou absoluto, além de poder atribuir comportamentos característicos dessa propriedade, como permitir que a imagem preencha todo o espaçamento do elemento, que ela seja maior ou menor que o elemento que o fundo do elemento e etc.

- Valores Aceitáveis:

- auto: Pega automaticamente o tamanho original da imagem;
- length: Qualquer valor relativo ou absoluto, podemos escolher entre definir um único valor ou 2 valores, se definirmos um único valor, ele vai ser aplicado sobre a largura do elemento, se definirmos 2 valores o 1º será aplicado sobre a largura e o 2º será aplicado sobre a altura;
- cover: "Cobrir". Vai cobrir toda a área disponível sem esticar a imagem, nem que para isso seja necessário que ela perca partes da imagem;
- contain: "Conter". Vai colocar toda a imagem dentro da área disponível sem perder partes da imagem, mas caso o container da imagem não seja adequado para ela, ele vai repetir a imagem mais de uma vez até cobrir o container;
- unset: Para desligar a propriedade;
- initial: Volta ao valor inicial;
- inherit: Para herdar o valor do elemento pai;

- Valor Padrão: auto;
- Possui valor herdado? - NÃO;

OBS: Devemos tomar cuidado ao usar valores absolutos e relativos. Geralmente usamos valores absolutos quando temos uma imagem onde não queremos que ela sofra nenhuma alteração de tamanho quanto ao responsivo. E usamos o relativo quando desejamos que a imagem se adeque ao layout responsivo onde ela se encontra;

-->

```
<div class="container">

  <div class="bloco">
    background-size: auto
    <span style="font-weight: normal;">Veja que a imagem é grande, por isso só carrega uma parte pequena da imagem.
    </span>
    <div id="auto">
    </div>

  </div>

  <div class="bloco">
    background-size: length
    <span style="font-weight: normal;">Veja que escolhermos dois valores para esse background, que é 100% X 33%,
    veja que o primeiro valor é 100% por isso ocupa toda a largura, enquanto o 2º que é 33% ocupa a altura repetindo
    a imagem 3 vezes.</span>
    <div id="length">
    </div>

  </div>

  <div class="bloco">
    background-size: cover
    <span style="font-weight: normal;">Perceba que ele pega toda a área da altura que tem disponível e corta a
    largura devido ao formato do container, o cover sempre vai renderizar o melhor formato para não esticar a
    imagem, por isso ele corta algumas partes.</span>
```

```
<div id="cover">
</div>

</div>

<div class="bloco">
  background-size: contain
  <span style="font-weight: normal;">Perceba que o contain vai fazer o máximo para que a imagem caíba inteira
  dentro do container, mesmo que para isso ele tenha que repetí-la.</span>
  <div id="contain">
  </div>

</div>

</div>

</body>
</html>
```

**RESULTADO LOGO ABAIXO...**

### background-size: auto

Veja que a imagem é grande, por isso só carrega uma parte pequena da imagem.



### background-size: length

Veja que escolhermos dois valores para esse background, que é 100% X 33%, veja que o primeiro valor é 100% por isso ocupa toda a largura, enquanto o 2º que é 33% ocupa a altura repetindo a imagem 3 vezes.



### background-size: cover

Perceba que ele pega toda a área da altura que tem disponível e corta a largura devido ao formato do container, o cover sempre vai renderizar o melhor formato para não esticar a imagem, por isso ele corta algumas partes.



### background-size: contain

Perceba que o contain vai fazer o máximo para que a imagem caiba inteira dentro do container, mesmo que para isso ele tenha que repeti-la.

