

```
<!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>Flexbox</title>
  <link rel="stylesheet" href="005-style.css">
</head>
<body>
```

```
  <!--
    Flexbox
```

Falando de maneira simples, imagine que as caixas dos nossos elementos HTML agora fossem flexíveis, que pudéssemos posicioná-las em qualquer lado da nossa página, em sentido de linha, de coluna, que pudéssemos expandí-las ou reduzi-las. Tudo isso é possível usando o conceito de flexbox.

Para que consigamos fazer isso, primeiro temos que ter um elemento que vá usar um "display: flex", o valor flex irá fazer com que um elemento possa atribuir os comportamentos de flexbox aos elementos amarrados dentro dele. Lembre-se sempre que todos os elementos que recebem flexbox se transformam em container dos elementos que vão ser flexionados, e os elementos recebem o nome de "flex-items", pois eles serão os itens que sofrerão com a flexibilidade. Caso os flex-items não possuam formatação específica, eles vão sofrer flexibilidade de acordo o seu elemento pai ou com a formatação dos seus elementos internos.

Em seguida vamos usar a propriedade "flex-direction", essa propriedade pode definir um eixo central, ou principal da nossa página, trabalhando em duas dimensões "x" (principal) e "y" (eixo secundário). O nosso eixo principal pode mudar de direção dependendo do valor que escolhermos para a nossa propriedade "flex-direction".

Temos os seguintes valores:

- row: O eixo principal (x) será em linha vertical indo da esquerda para a direita e o secundário (y) será transversal de cima para baixo; (Por exemplo um menu de um site em desktop pode usar o flex-direction: row);
- column: o eixo principal (x) será uma linha transversal de cima para baixo e o secundário (y) será uma linha vertical indo da esquerda para a direita; (Por exemplo um menu de um smart-phone em desktop pode usar o flex-direction: row);

```

- row-
reverse: O eixo principal (x) será em linha vertical indo da direita para a esquerda e o secundário (y) será transversal de cima para baixo; (Por exemplo um menu de um site em desktop pode usar o flex-direction: row);

- column-
reverse: o eixo principal (x) será uma linha transversal de baixo para cima e o secundário (y) será uma linha vertical indo da direita para a esquerda; (Por exemplo um menu de um site em desktop pode usar o flex-direction: row);

-->
<div id="posicao">

    <button type="button" class="pulse">Home</button>
    <button type="button" class="pulse">Contatos</button>
    <button type="button" class="pulse">Loja</button>
    <button type="button" class="pulse">Sobre</button>
    <button type="button" class="pulse">Produtos</button>

</div>

</body>
</html>

```

CSS

```

/*Usando Flexbox*/
/**/

body {
    background-color: #17181c;
    color: #fff;
    font-family: sans-serif;
    margin: 0; /*Tiramos as margens para que todos os elementos fiquem encostados*/
    height: 200vh; /*Dobramos o tamanho do body para que ele ocupe o dobro do tamanho da janela do navegador*/
    background: url(004-boku.png); /*Imagem para perceber que a página está andando*/
}

```

```
#posicao {
  position: sticky;
  top: 0;
  width: 100%;
  height: 400px;
  background-color: #000;
  display: flex;
  /* Ex:1 - Ative somente o display flex e veja que o todos os flex-items são ser flexionados para obedecer a ele, se os flex-
  items não tiverem valores pré-
  definidos, eles vão obedecer altura e largura que o flex determinar para eles. Perceba que os elementos por padrão estão alinhados em
  linha, e cada um ganhou uma largura compatível com o que foi escrito*/
  flex-direction: column;
  /* Ex: 2 - Perceba que quando usamos o "flex-direction: column" a ordem de apresentação dos botões muda*/
  flex-direction: column-reverse;
  /* Ex:3 - Perceba que no column-reverse a ordem deslocção dos botões muda de baixo para cima. */
  flex-direction: row-reverse;
  /* Ex: 4 - Perceba que os botões começam da direita para a esquerda agora. */
}

.pulse {
  background: none;
  color: #ff760c;
  border: 2px solid;
  margin: 0;
  padding: 15px 40px;
  transition: 0.25s;
}

.pulse:hover, .pulse:focus {
  color: #fff;
  border-color: #ef8f6e;
  box-shadow: 0 0 0 15px rgba(255, 255, 255, 0);
  animation: pulsar 1s;
}
```

```
@keyframes pulsar {  
  0% {  
    box-shadow: 0 0 0 0 #ef8e6e;  
  }  
}
```