



# CSS

## Flexbox



ou Flexible Box Layout, é um novo modo de layout em CSS3, projetado para elaboração de layouts complexos em páginas web.

Flexbox é flexível porque distribui o espaço e alinha o conteúdo de maneira mais fácil do que com o uso de floats. E ele também trabalha com alinhamento vertical, algo trabalhoso de fazer até então.

### Principais vantagens

Permite organização, reordenação, alinhamento e definição de espaços entre os elementos dentro de um container.

Itens dentro de um container poder ser esticados e contraídos de acordo com o espaço disponível.

Possui flex directions, que permite definir o layout em qualquer direção, horizontal ou vertical.



# CSS

## Flexbox - Propriedades principais

***flex-direction***

***flex-wrap***

***flex-flow***

***order***

***justify-content***

***align-items***

***align-content***

***align-self***

***flex-grow***

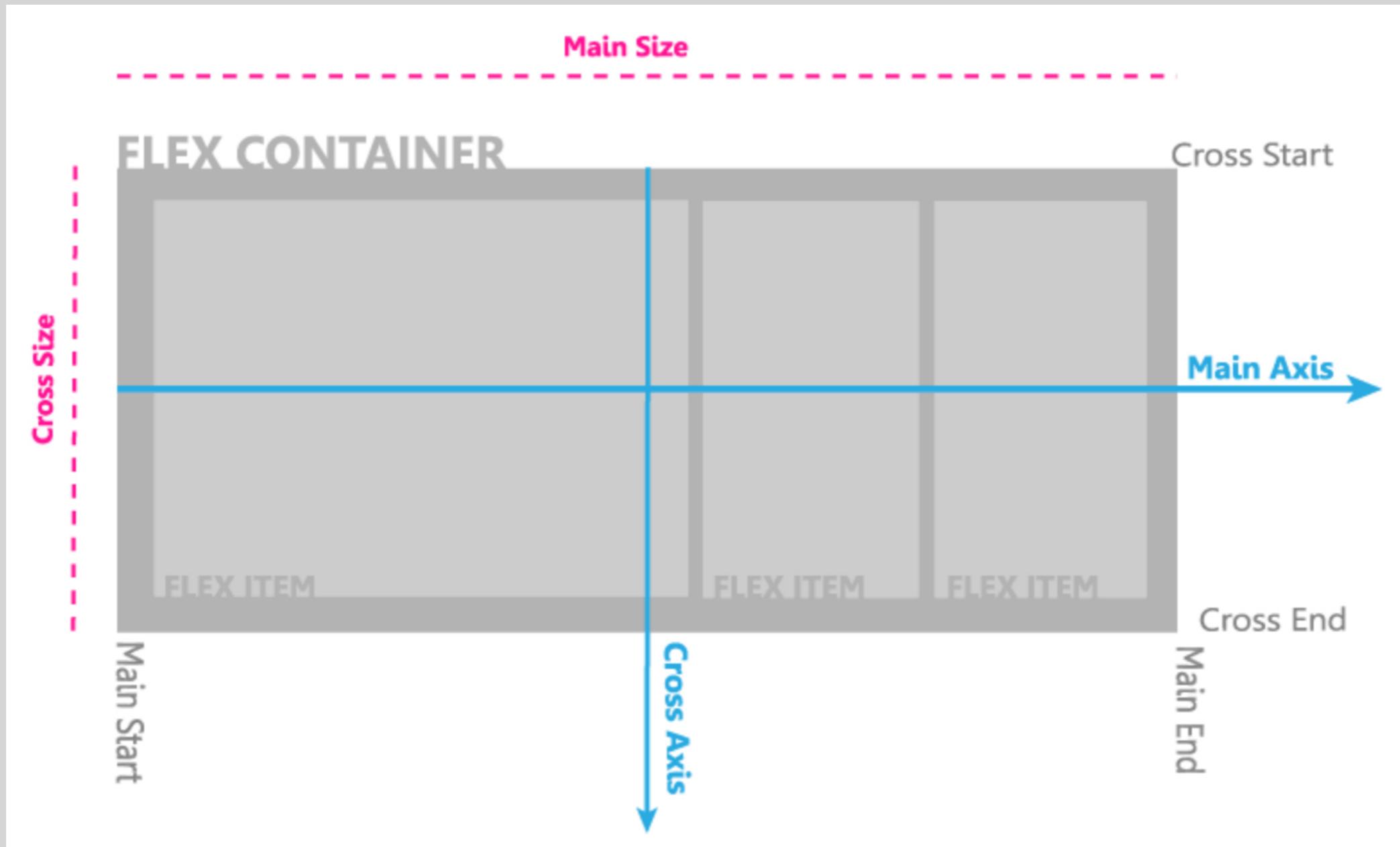
***flex-shrink***

***flex-basis***

***flex***

# CSS

## Eixos flex (Flex Axes) e Linhas flex (Flex Lines)



Absolutamente tudo em Flexbox é relativo a estes 2 eixos

# CSS

## Flex container



Para usar Flexbox é preciso criar um flex container. Os elementos filhos de um flex container são chamados de flex items.

Para criar um flex container deve usar a propriedade display com o valor **flex** ou **inline-flex**.

```
.flex-container{  
    display: flex;  
}
```

**Nível de bloco**

```
.flex-container{  
    display: inline-flex;  
}
```

**Nível de linha**

# CSS

## Flex container



O flex container pode ser single-line (todos os elementos são dispostos numa única linha, mesmo que isso “estoure” o conteúdo).

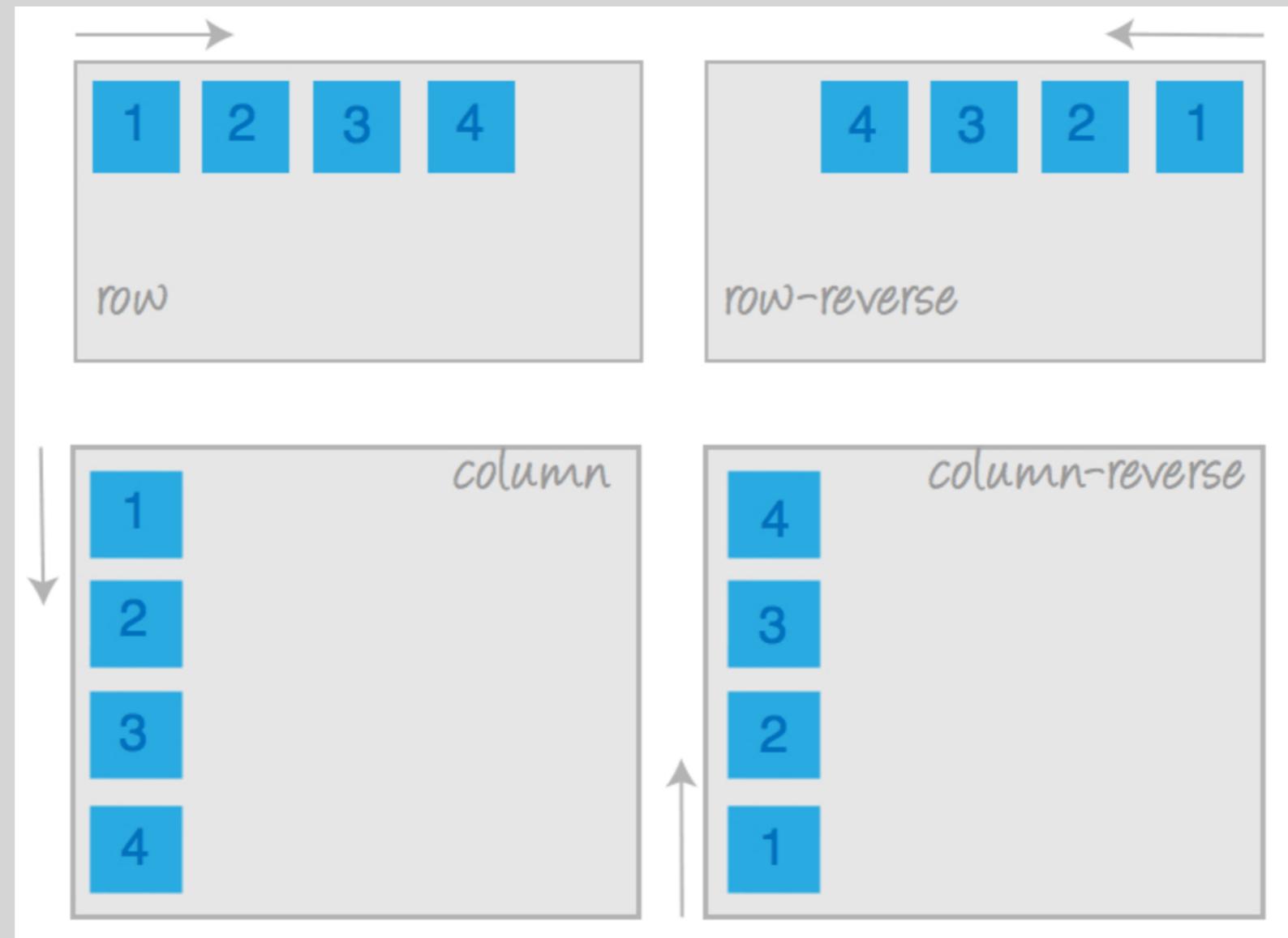
Pode ser também multi line (os elementos são dispostos em múltiplas linhas, linhas adicionais são criadas no container).

# CSS

## flex-direction



row | row-reverse | column | column-reverse

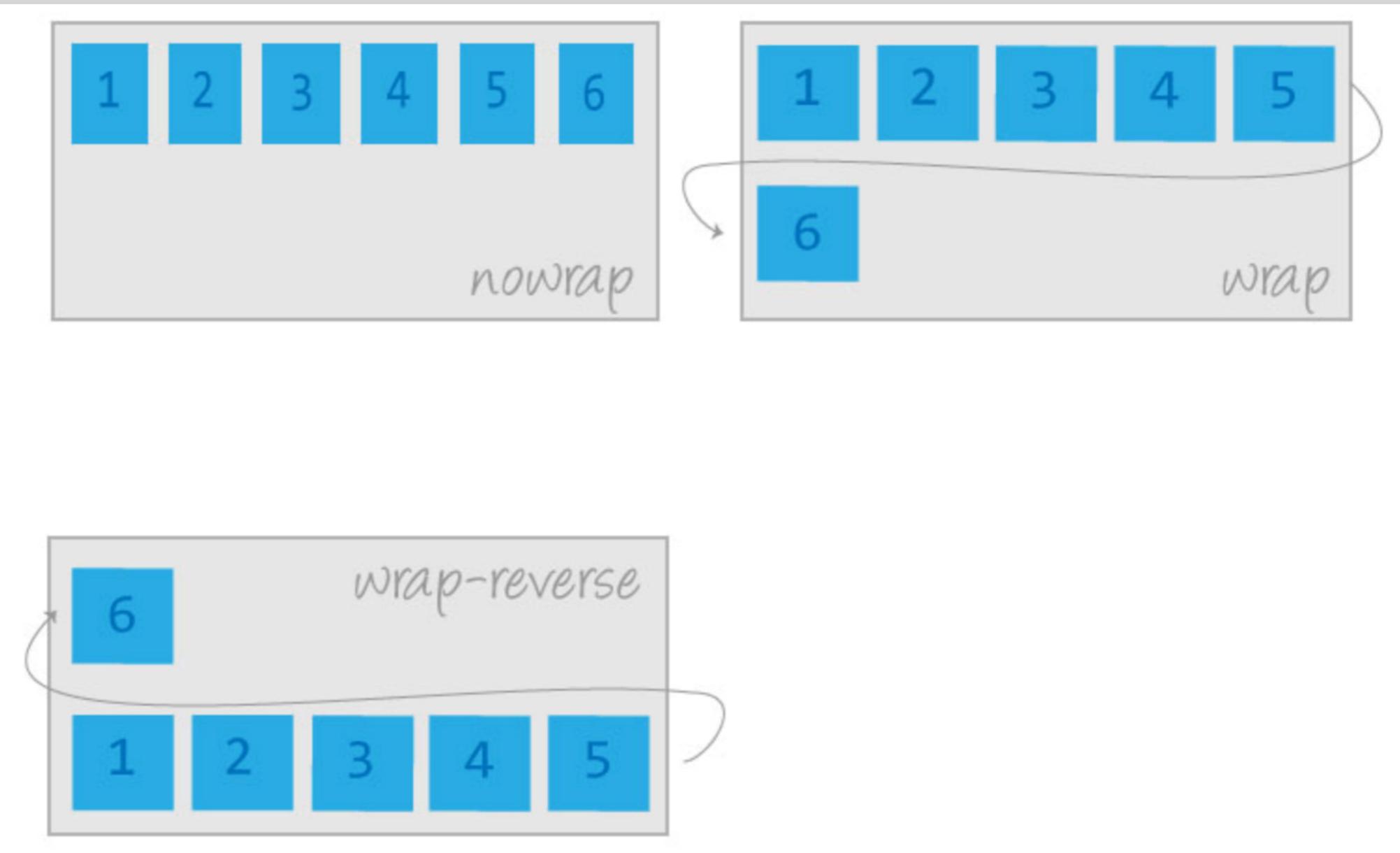


# CSS

## flex-wrap



nowrap | wrap | wrap-reverse



# CSS

## justify-content



flex-start | flex-end | center | space-between | space-around

flex-start



flex-end



center



space-between



space-around



# CSS

## flex-direction



```
<div class="container" id="container">
  <div class="item">1</div>
  <div class="item">2</div>
  <div class="item">3</div>
  <div class="item">4</div>
  <div class="item">5</div>
</div>
```

# CSS

## flex-direction



```
body{  
background-color: #F5F5F5;  
color: #555;  
font-size: 1.1em;  
font-family: 'Helvetica Neue', helvetica, arial, sans-serif;  
}
```

# CSS

## flex-direction



```
.container{  
  margin: 150px auto;  
  max-width: 800px;  
  padding: 20px;  
  display: flex;  
  background-color: #594255;  
  flex-direction: row;           row  
}  
  
flex-direction: row-reverse;    row-reverse  
flex-direction: column;        column  
flex-direction: column-reverse; column-reverse
```

# CSS

## flex-direction



```
.item{  
    background-color: #fff;  
    width: 100px;  
    height: 100px;  
    margin: 2px;  
    text-align: center;  
    line-height: 100px;  
    font-weight: bold;  
    font-family: sans-serif;  
    font-size: 5em;  
    color: #594235;  
}
```

# CSS

## flex-wrap



```
<div class="container" id="container">  
    <div class="item">1</div>  
    <div class="item">2</div>  
    <div class="item">3</div>  
    <div class="item">4</div>  
    <div class="item">5</div>  
    <div class="item">6</div>  
    <div class="item">7</div>  
    <div class="item">8</div>  
</div>
```

# CSS

## flex-wrap



```
body{  
background-color: #F5F5F5;  
color: #555;  
font-size: 1.1em;  
font-family: 'Helvetica Neue', helvetica, arial, sans-serif;  
}
```

# CSS

## flex-wrap



```
.container{  
    margin: 150px auto;  
    max-width: 800px;  
    padding: 20px;  
    display: flex;  
    background-color: #594255;  
    flex-wrap = 'nowrap'  
}
```

**nowrap**

**wrap**

**wrap-reverse**

# CSS

## flex-wrap



```
.item{  
    background-color: #fff;  
    width: 100px;  
    height: 100px;  
    margin: 2px;  
    text-align: center;  
    line-height: 100px;  
    font-weight: bold;  
    font-family: sans-serif;  
    font-size: 7em;  
    color: #594235;  
}
```

# CSS

## justify-content



```
<div class="container" id="container">
  <div class="item big">1</div>
  <div class="item">2</div>
  <div class="item">3</div>
  <div class="item big">4</div>
  <div class="item">5</div>
</div>
```

# CSS

## justify-content



```
body{  
background-color: #F5F5F5;  
color: #555;  
font-size: 1.1em;  
font-family: 'Helvetica Neue', helvetica, arial, sans-serif;  
}
```

# CSS

## justify-content



```
.container{  
  margin: 150px auto;  
  max-width: 840px;  
  padding: 20px;  
  display: flex;  
  background-color: #594255;  
  justify-content = 'flex-start'  
}
```

**flex-start**  
**flex-end**  
**center**  
**space-between**  
**space-around**

# CSS

## justify-content



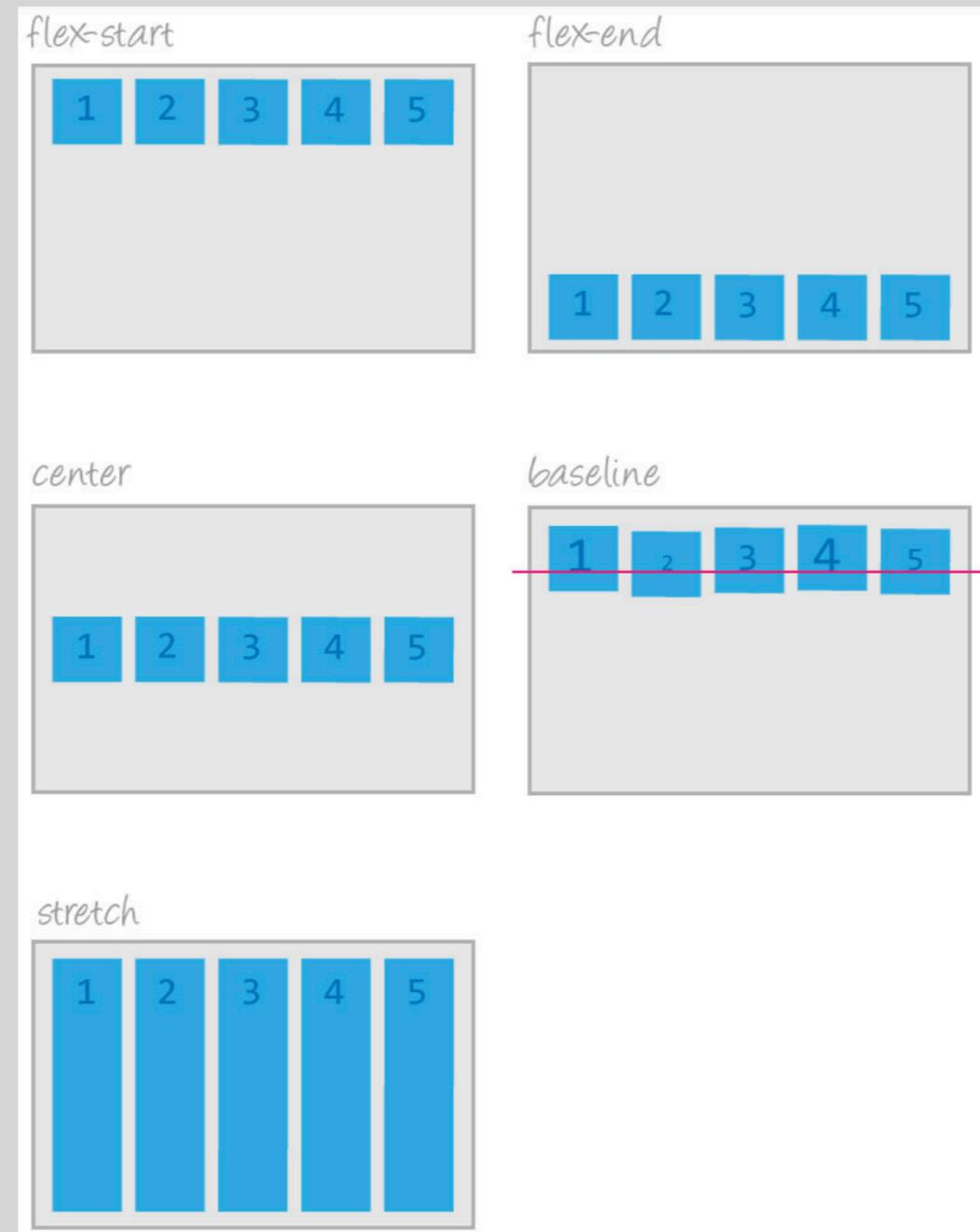
```
.item {  
    background-color: #fff;  
    width: 100px;  
    height: 100px;  
    margin: 1%;  
    text-align: center;  
    line-height: 100px;  
    font-weight: bold;  
    font-family: sans-serif;  
    font-size: 6em;  
    color: #594255;  
}  
  
.big {  
    width: 150px;  
}
```

# CSS

## align-items



flex-start | flex-end | center | baseline | stretch



# CSS

## align-items



```
<div class="container" id="container">
  <div class="item big">1</div>
  <div class="item">2</div>
  <div class="item small">3</div>
  <div class="item big">4</div>
  <div class="item">5</div>
  <div class="line" id="line"></div>
</div>
```

# CSS

## align-items



```
body{  
background-color: #F5F5F5;  
color: #555;  
font-size: 1.1em;  
font-family: 'Helvetica Neue', helvetica, arial, sans-serif;  
}
```

# CSS

## align-items



```
.container {  
    margin: 150px auto;  
    max-width: 840px;  
    height: 400px;  
    padding: 20px;  
    display: flex;  
    background-color: #594255;  
    position: relative;  
    align-items: flex-start;  
}
```

**flex-start**  
**flex-end**  
**center**  
**baseline**  
**stretch**

# CSS

## align-items



```
.line {  
    height: 1px;  
    width: 100%;  
    border-bottom: 2px solid deepPink;  
    position: absolute;  
    top: 120px;  
    left: 0;  
    visibility: hidden;  
}
```

# CSS

## align-items



```
.item {  
    background-color: #fff;  
    width: 100px;  
    margin: 1%;  
    text-align: center;  
    line-height: 100px;  
    font-weight: bold;  
    font-family: sans-serif;  
    font-size: 4em;  
    color: #594255;  
}
```

# CSS

## align-items



```
.big {  
    width: 150px;  
    font-size: 5em;  
}  
  
.small {  
    font-size: 2em;  
}
```



# Flexbox

## Soluções para problemas comuns

Centralização horizontal e vertical

1 - crie um container flex

```
.container {  
    display: flex;  
}
```

2 - com a propriedade justify-content,  
centralize o elemento no Main Axis (horizontal)  
e com align-items, no Cross Axis (vertical)

```
.container {  
    align-items: center;  
    display: flex;  
    justify-content: center;  
}
```

**Não importa o tamanho do elemento, ele sempre será  
alinulado na vertical e na horizontal.**

# CSS

## Centralização horizontal e vertical



```
<div class="container">  
  <div class="elemento">  
    Este elemento está centralizado. Não importa o quanto  
    seu tamanho seja alterado, ele sempre será centralizado  
    horizontal e verticalmente.  
  </div>  
</div>
```

Este elemento está centralizado. Não importa o quanto seu tamanho seja alterado, ele sempre será centralizado horizontal e verticalmente.

# CSS

## Centralização horizontal e vertical



```
body {  
    background-color: #F5F5F5;  
    color: #555;  
    font-size: 1.1em;  
    font-family: 'Helvetica Neue', Helvetica, Arial, sans-serif;  
}  
  
.container {  
    max-width: 800px;  
    margin: 3em auto;  
    background-color: white;  
    height: 500px;  
    border: 10px solid #594255;  
    display: flex;  
    align-items: center;  
    justify-content: center;  
}  
  
.element {  
    background-color: #594255;  
    color: white;  
    width: 250px;  
    padding: 20px;  
}
```



# Flexbox

## Soluções para problemas comuns

### Layout muticolumnas

```
<header>Cabeçalho</header>
<div class="container">
  <section class="main">
    <h1>Notícias</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Ducimus, qui totam placeat
    ullam eligendi itaque preferendis similique aliquam inventore tenetur. Eveniet quisquam unde
    vitae expedita voluptates soluta in obcaecati recusandae.</p>
    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Magnam, voluptas autem sed a
    rem impedit quidem facere dolore praesentium optio provident minus deleniti eveniet ex cum fuga
    preferendis illum molestiae!</p>
    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Asperiores, repellendus, nobis
    hic molestias eveniet preferendis nesciunt voluptate ipsum impedit quo a veritatis porro qui
    commodi expedita? Nostrum enim sed ab.</p>
    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Eos, aspernatur reiciendis
    animi sequi ducimus odit ad quos libero fugiat vitae natus optio dolores reprehenderit enim
    aliquid necessitatibus deleniti voluptatum quod.</p>
    <p>
      Lorem ipsum dolor sit amet, consectetur adipisicing elit. Nobis, cumque, eveniet,
      veritatis, quae esse delectus non omnis incident nihil saepe labore neque fugiat praesentium.
      Nulla quam pariatur eos hic tempore.
    </p>
  </section>
```



# Flexbox

## Soluções para problemas comuns

### Layout muticolumnas

```
<aside>
  <h1>Barra Lateral</h1>
  <p>A barra lateral pode conter um menu de opções ou pouco texto. Ela será redimensionada
igualmente a segunda coluna.</p>
</aside>
</div>
<footer>Rodapé</footer>
```



# Flexbox

```
body {  
  color: #555;  
  font-size: 1.1em;  
  font-family: 'Helvetica Neue', Helvetica, Arial, sans-serif;  
}  
  
header,  
footer {  
  height: 100px;  
  line-height: 100px;  
  text-align: center;  
  background-color: #ccc;  
}  
  
section,  
aside {  
  background-color: #594255;  
  color: #eee;  
  padding: 3em 1.5em;  
  margin: 1% 0;  
}  
  
section {  
  flex: 2 1 500px;  
  margin-right: 1%;  
}  
  
aside {  
  flex: 1 2 250px;  
}  
  
.container {  
  display: flex;  
  align-items: stretch;  
}  
  
@media screen and (max-width: 500px) {  
  .container {  
    flex-direction: column;  
  }  
  section {  
    margin-right: 0;  
    flex-basis: auto;  
  }  
}
```