

```
<!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>justify-self</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: 50%;
        padding: 5px;
        display: flex;
        justify-content: space-between;
        align-items: center;
    }

    .bloco {
```

```
width: 16%;
height: 100%;
padding: 2px;
margin: 4px;
box-shadow: 10px 10px 10px rgba(0, 0, 0, .2);
display: flex;
justify-content: flex-end;
flex-direction: column;
}

.bloco > div {
  background-color: white;
}

.div1 {
  background-color: aqua;
  width: 30%;
  height: 50%;
  font-size: x-large;
  text-align: center;
}

.div2 {
  background-color: violet;
  width: 30%;
  height: 50%;
  font-size: x-large;
  text-align: center;
}

.div3 {
  background-color: rgb(248, 203, 121);
  width: 30%;
  height: 50%;
  font-size: x-large;
  text-align: center;
}
```

```
}

.div4 {
  background-color: turquoise;
  width: 30%;
  height: 50%;
  font-size: x-large;
  text-align: center;
}

#stretch {
  width: 100%;
  height: 80%;
  background-color: rgb(252, 248, 213);
  display: grid;
  place-items: center;
}

#stretch > .div3 {
  width: unset;
  justify-self: stretch;
}

#flex-start {
  width: 100%;
  height: 80%;
  background-color: rgb(252, 248, 213);
  display: grid;
  place-items: center;
}

#flex-start > .div3 {
  justify-self: flex-start;
}

#flex-end {
  width: 100%;
```

```
    height: 80%;
    background-color: rgb(252, 248, 213);
    display: grid;
    place-items: center;
}
#flex-end > .div3 {
    justify-self: flex-end;
}

#center {
    width: 100%;
    height: 80%;
    background-color: rgb(252, 248, 213);
    display: grid;
    place-items: start;
}
#center > .div3 {
    justify-self: center;
}

#baseline {
    width: 100%;
    height: 80%;
    background-color: rgb(252, 248, 213);
    display: grid;
    place-items: center;
}
#baseline > .div3 {
    width: unset;
    justify-self: baseline;
}

#self-start {
    width: 100%;
    height: 80%;
    background-color: rgb(252, 248, 213);
```

```
    display: grid;
    place-items: center;
}
#self-start > .div3 {
    justify-self: self-start;
}

#self-end {
    width: 100%;
    height: 80%;
    background-color: rgb(252, 248, 213);
    display: grid;
    place-items: center;
}
#self-end > .div3 {
    justify-self: self-end;
}

#start {
    width: 100%;
    height: 80%;
    background-color: rgb(252, 248, 213);
    display: grid;
    place-items: center;
}
#start > .div3 {
    justify-self: start;
}

#end {
    width: 100%;
    height: 80%;
    background-color: rgb(252, 248, 213);
    display: grid;
    place-items: center;
}
```

```
#end > .div3 {
  justify-self: end;
}

#right {
  width: 100%;
  height: 80%;
  background-color: rgb(252, 248, 213);
  display: grid;
  place-items: center;
}
#right > .div3 {
  justify-self: right;
}

#left {
  width: 100%;
  height: 80%;
  background-color: rgb(252, 248, 213);
  display: grid;
  place-items: center;
}
#left > .div3 {
  justify-self: left;
}
```

</style>

</head>

<body>

<!--

Propriedade: justify-self (Jutificar a Si Mesmo)

Objetivo: Essa propriedade define um alinhamento sobre o eixo (linha) primário do grid sobre um grid-item

individualmente, ou seja, diferente do "justify-items" que define um alinhamento horizontal para todos os grid-items dentro de um container, o "justify-self" vai definir um comportamento para um único grid-item individualmente.

Por esse motivo, a propriedade "justify-self" só pode ser aplicada sobre o grid-item e NUNCA SOBRE O CONTAINER!!!

DETALHE IMPORTANTE!!! Essa propriedade só é capaz de alinhar os nossos grid-items dentro do eixo (linha) do grid.

- Valores Aceitáveis:

- first: Ainda não está habilitado para alguns navegadores!!;
- last: Ainda não está habilitado para alguns navegadores!!;
- normal: Segue o alinhamento padrão do grid;
- stretch: Estica o grid-item se eles não tiverem tamanho pré definido onde o esticamento acontece;
- flex-start: Funciona em apenas alguns navegadores que já se adaptaram ao uso do flex como valor para grids (Já funciona no Chrome)!!! Alinha o grid-item no início de um eixo embutido do grid;
- flex-end: Funciona em apenas alguns navegadores que já se adaptaram ao uso do flex como valor para grids (Já funciona no Chrome)!!! Alinha o grid-item a partir de onde termina o eixo embutido do grid;
- center: Alinha o grid-item no centro do eixo do grid;
- baseline: Alinha o grid-item ao tamanho da maior letra dentro de um elemento;
- first-baseline: Alinha o grid-item ao tamanho da maior letra de um primeiro parágrafo - porém esse valor ainda é INVÁLIDO em alguns navegadores para esta propriedade;

- last-baseline: Alinha o grid-item ao tamanho da menor letra do último parágrafo - porém esse valor ainda é INVÁLIDO em alguns navegadores para esta propriedade;
- auto: Segue automaticamente o valor padrão ou do elemento pai;
- self-start: Posiciona o grid-item no começo de cada linha grid;
- self-end: Posiciona o grid-item no final de cada linha grid;
- start: Posiciona o grid-item no começo de cada linha grid;
- end: Posiciona o grid-item no final de cada linha grid;
- right: Ainda não está disponível para todos os navegadores (Já está disponível no Chrome)!!! Posiciona o grid-item á direita de cada linha grid;
- left: Ainda não está disponível para todos os navegadores (Já está disponível no Chrome)!!! Posiciona o grid-item á esquerda de cada linha grid;
- 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: Essa propriedade tem precedência sobre a propriedade "justify-items"!!!;

-->

```
<div class="container">
```

```
  <div class="bloco" style="justify-content: center;">
```


Veja exemplos de uso da propriedade: `
justify-self
` nos nossos grids `
⇒`
`</div>`

```
<div class="bloco">
  justify-self: stretch
  <span style="font-weight: normal;font-size: 10pt;">Perceba que somente o terceiro grid-item está sendo esticado
    horizontalmente enquanto os outros permanecem alinhados ao centro...
  </span>
  <div id="stretch">
    <div class="div1"></div>
    <div class="div2"></div>
    <div class="div3"></div>
    <div class="div4"></div>
  </div>
  <span style="font-weight: bold;font-size: 8pt;">Atenção: Valores aplicados sobre um place-item: center;</span>
</div>
```

```
<div class="bloco">
  justify-self: flex-start
  <span style="font-weight: normal;font-size: 10pt;">Perceba que somente o terceiro grid-item está sendo
    posicionado no começo do eixo grid, enquanto os outros permanecem alinhados ao centro...
  </span>
  <div id="flex-start">
    <div class="div1"></div>
    <div class="div2"></div>
    <div class="div3"></div>
    <div class="div4"></div>
  </div>
  <span style="font-weight: bold;font-size: 8pt;">Atenção: Valores aplicados sobre um place-item: center;</span>
</div>
```

```
<div class="bloco">
  justify-self: flex-end
  <span style="font-weight: normal;font-size: 10pt;">Perceba que somente o terceiro grid-item está sendo
    posicionado ao final do eixo grid, enquanto os outros permanecem alinhados ao centro...
  </span>
```

```
<div id="flex-end">
  <div class="div1"></div>
  <div class="div2"></div>
  <div class="div3"></div>
  <div class="div4"></div>
</div>
<span style="font-weight: bold;font-size: 8pt;">Atenção: Valores aplicados sobre um place-item: center;</span>
</div>
```

```
<div class="bloco">
  justify-self: center
  <span style="font-weight: normal;font-size: 10pt;">Perceba que somente o terceiro grid-item está sendo
    posicionado ao centro do eixo grid, enquanto os outros permanecem alinhados ao centro...
  </span>
  <div id="center">
    <div class="div1"></div>
    <div class="div2"></div>
    <div class="div3"></div>
    <div class="div4"></div>
  </div>
  <span style="font-weight: bold;font-size: 8pt;">Atenção: Valores aplicados sobre um place-item: center;</span>
</div>
```

```
<div class="bloco">
  justify-self: baseline
  <span style="font-weight: normal;font-size: 10pt;">Perceba que somente o terceiro grid-item está sendo
    posicionado em relação a área da letra, enquanto os outros permanecem alinhados ao centro...
  </span>
  <div id="baseline">
    <div class="div1"><span style="font-size: 10pt;">A</span></div>
    <div class="div2"><span style="font-size: 50pt;">B</span></div>
    <div class="div3"><span style="font-size: 32pt;">C</span></div>
    <div class="div4"><span style="font-size: 22pt;">D</span></div>
  </div>
  <span style="font-weight: bold;font-size: 8pt;">Atenção: Valores aplicados sobre um place-item: center;</span>
</div>
```



```

    posicionado no começo e ao topo do eixo grid, enquanto os outros permanecem alinhados ao centro...
</span>
<div id="start">
    <div class="div1"></div>
    <div class="div2"></div>
    <div class="div3"></div>
    <div class="div4"></div>
</div>
<span style="font-weight: bold;font-size: 8pt;">Atenção: Valores aplicados sobre um place-item: center;</span>
</div>

<div class="bloco">
    justify-self: end
    <span style="font-weight: normal;font-size: 10pt;">Perceba que somente o terceiro grid-item está sendo
        posicionado no final e na parte de baixo do eixo grid, enquanto os outros permanecem alinhados ao centro...
    </span>
    <div id="end">
        <div class="div1"></div>
        <div class="div2"></div>
        <div class="div3"></div>
        <div class="div4"></div>
    </div>
    <span style="font-weight: bold;font-size: 8pt;">Atenção: Valores aplicados sobre um place-item: center;</span>
</div>

<div class="bloco">
    justify-self: right
    <span style="font-weight: normal;font-size: 10pt;">Perceba que somente o terceiro grid-item está sendo
        posicionado á direita do eixo grid, enquanto os outros permanecem alinhados ao centro...
    </span>
    <div id="right">
        <div class="div1"></div>
        <div class="div2"></div>
        <div class="div3"></div>
        <div class="div4"></div>
    </div>

```

```
<span style="font-weight: bold;font-size: 8pt;">Atenção: Valores aplicados sobre um place-item: center;</span>
</div>

<div class="bloco">
  justify-self: left
  <span style="font-weight: normal;font-size: 10pt;">Perceba que somente o terceiro grid-item está sendo
    posicionado á esquerda do eixo grid, enquanto os outros permanecem alinhados ao centro...
  </span>
  <div id="left">
    <div class="div1"></div>
    <div class="div2"></div>
    <div class="div3"></div>
    <div class="div4"></div>
  </div>
  <span style="font-weight: bold;font-size: 8pt;">Atenção: Valores aplicados sobre um place-item: center;</span>
</div>

</div>

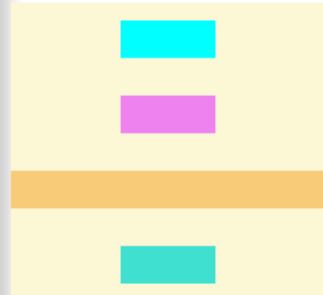
</body>
</html>
```

RESULTADO LOGO ABAIXO...

Veja exemplos de uso da propriedade: **justify-self** nos nossos grids
⇒

justify-self: stretch

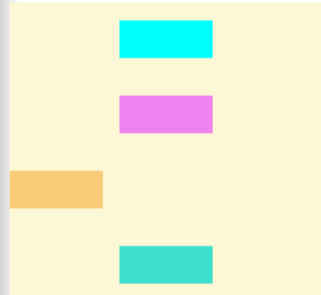
Perceba que somente o terceiro grid-item está sendo esticado horizontalmente enquanto os outros permanecem alinhados ao centro...



Atenção: Valores aplicados sobre um place-item: center;

justify-self: flex-start

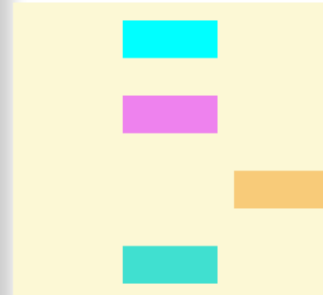
Perceba que somente o terceiro grid-item está sendo posicionado no começo do eixo grid, enquanto os outros permanecem alinhados ao centro...



Atenção: Valores aplicados sobre um place-item: center;

justify-self: flex-end

Perceba que somente o terceiro grid-item está sendo posicionado ao final do eixo grid, enquanto os outros permanecem alinhados ao centro...



Atenção: Valores aplicados sobre um place-item: center;

justify-self: center

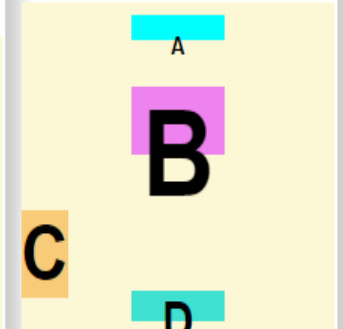
Perceba que somente o terceiro grid-item está sendo posicionado ao centro do eixo grid, enquanto os outros permanecem alinhados ao centro...



Atenção: Valores aplicados sobre um place-item: center;

justify-self: baseline

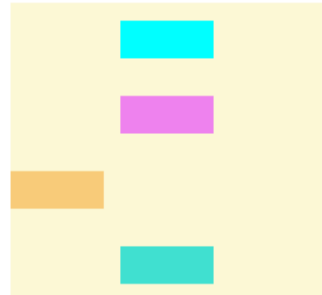
Perceba que somente o terceiro grid-item está sendo posicionado em relação a área da letra, enquanto os outros permanecem alinhados ao centro...



Atenção: Valores aplicados sobre um place-item: center;

justify-self: self-start

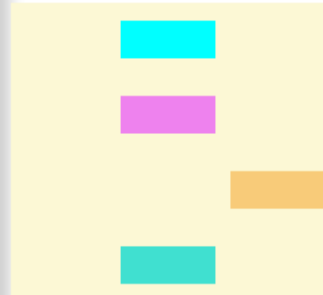
Perceba que somente o terceiro grid-item está sendo posicionado ao início da sua área de eixo grid, enquanto os outros permanecem alinhados ao centro...



Atenção: Valores aplicados sobre um place-item: center;

justify-self: self-end

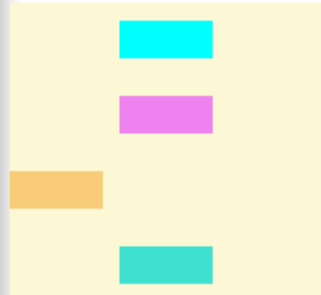
Perceba que somente o terceiro grid-item está sendo posicionado ao final da sua área de eixo grid, enquanto os outros permanecem alinhados ao centro...



Atenção: Valores aplicados sobre um place-item: center;

justify-self: start

Perceba que somente o terceiro grid-item está sendo posicionado no começo e ao topo do eixo grid, enquanto os outros permanecem alinhados ao centro...



Atenção: Valores aplicados sobre um place-item: center;

justify-self: end

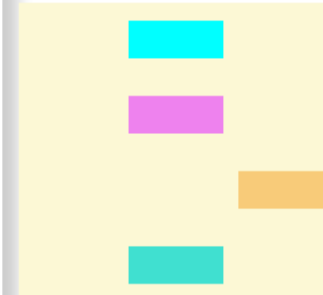
Perceba que somente o terceiro grid-item está sendo posicionado no final e na parte de baixo do eixo grid, enquanto os outros permanecem alinhados ao centro...



Atenção: Valores aplicados sobre um place-item: center;

justify-self: right

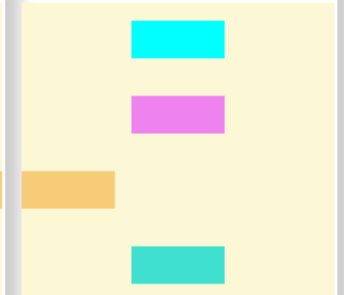
Perceba que somente o terceiro grid-item está sendo posicionado à direita do eixo grid, enquanto os outros permanecem alinhados ao centro...



Atenção: Valores aplicados sobre um place-item: center;

justify-self: left

Perceba que somente o terceiro grid-item está sendo posicionado à esquerda do eixo grid, enquanto os outros permanecem alinhados ao centro...



Atenção: Valores aplicados sobre um place-item: center;