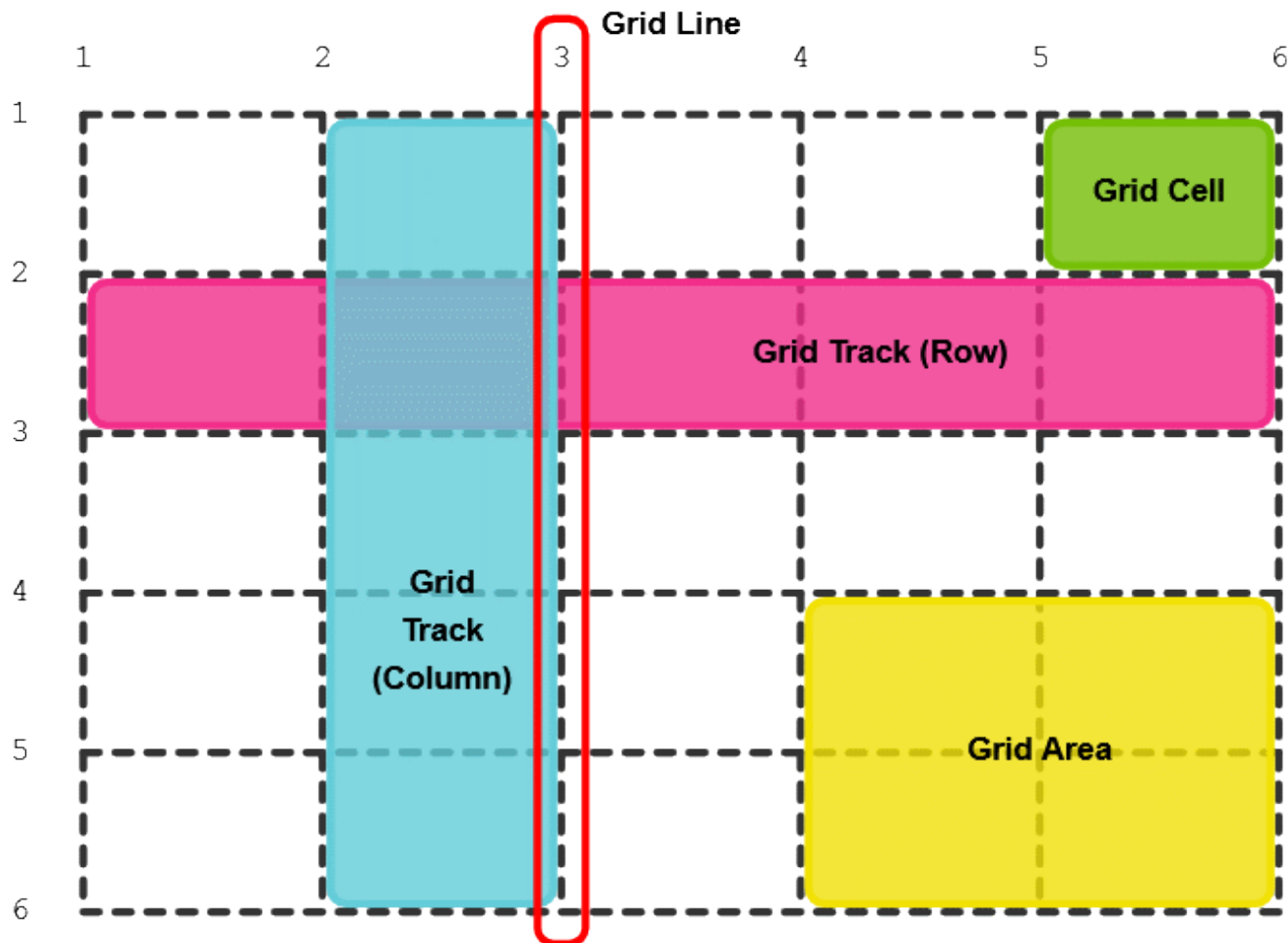


Grids

CSS Grid Layout é um sistema de layout bidimensional para a web. Ele permite que você disponha o conteúdo em linhas e colunas, auxiliando a construção de layouts complexos.



Elementos de uma grid

- **Grid line:** as linhas entre as linhas ou colunas de um grid.
- **Grid track:** uma linha ou uma coluna.
- **Track size:** a largura de uma coluna ou a altura de uma linha.
- **Grid cell:** a interseção de linhas com colunas.
- **Grid area:** uma área criada pela mesclagem de grid cells.

Usando grids

Para usar um grid, você aplica `display: grid` ao elemento **container** dos elementos que deseja dispor; todos os seus **items** diretos se tornam *items de grade*.

```
<!-- Código HTML de exemplo-->
<div class="container">
  <div class="itens">Um</div>
  <div class="itens">Dois</div>
  <div class="itens">Três</div>
  <div class="itens">Quatro</div>
  <div class="itens">Cinco</div>
  <div class="itens">Seis</div>
</div>
```

```
/* Código CSS de exemplo */
.container{
  display: grid;
}

.itens{
  background-color: orange;
  border: 2px solid black;
  border-radius: 5px;
  padding: 10px;
}
```

Propriedades do container

Linhas e colunas

Para acrescentar linhas e colunas à grade, utilize as propriedades `grid-template-columns` e `grid-template-rows`. É possível utilizar unidades fixas, porcentagens ou frações (que distribui o espaço disponível) na criação das colunas.

```
/* Código CSS de exemplo */
.container{
  display: grid;
  grid-template-columns: 200px 200px 200px;
}
```

```
/* Código CSS de exemplo */
.container{
  display: grid;
  grid-template-columns: 20% 30% 20%;
}
```

```
/* Código CSS de exemplo */
.container{
  display: grid;
  grid-template-columns: 2fr 1fr 1fr;
}
```

```
/* Código CSS de exemplo */
.container{
  display: grid;
  grid-template-columns: repeat(6, 1fr);
}
```

Espaçamento entre linhas ou colunas

Para adicionar intervalos entre as faixas, usamos a propriedade `columns-gap` para definir um espaço entre as colunas, a propriedade `row-gap` para definir um espaço entre as linhas ou a propriedade `gap` para definir um espaço tanto para linhas quanto para colunas.

```
/* Código CSS de exemplo */
.container{
  display: grid;
  grid-template-columns: repeat(6, 1fr);
  column-gap: 20px;
}

.container{
  display: grid;
  grid-template-rows: repeat(6, 1fr);
  row-gap: 20px;
}

.container{
  display: grid;
  grid-template-columns: repeat(3, 1fr);
  column-gap: 10px;
  row-gap: 20px;
}

.container{
  display: grid;
  grid-template-columns: repeat(3, 1fr);
  gap: 20px;
}
```

Alinhamento horizontal

O alinhamento horizontal dos itens **na célula** a partir da propriedade `justify-items`. Podem ser usados os valores `start`, `end`, `center` ou `stretch`.

```
/* Código CSS de exemplo */
.container{
  display: grid;
  grid-template-columns: repeat(3, 200px);
  justify-items: center;
}
```

Alinhamento vertical

O alinhamento vertical dos itens **na célula** a partir da propriedade `align-items`. Podem ser usados os valores `start`, `end`, `center` ou `stretch`.

```
/* Código CSS de exemplo */
.container{
  display: grid;
  height: 80vh;
  grid-template-rows: repeat(3, 200px);
  align-items: center;
}
```

Propriedades dos itens

Posicionamento de itens no grid

As propriedades `grid-column` e `grid-row` permitem definir, respectivamente, quais colunas ou linhas um determinado item ocupará.

```
<!-- Código HTML de exemplo-->
<div class="container">
  <div class="itens item-1">Um</div>
  <div class="itens item-2">Dois</div>
  <div class="itens item-3">Três</div>
  <div class="itens item-4">Quatro</div>
  <div class="itens item-5">Cinco</div>
  <div class="itens item-6">Seis</div>
</div>
```

```
/* Código CSS de exemplo */
.item-1{
  grid-column-start: 1;
  grid-column-end: 3;
}

.item-1{
  grid-row-start: 1;
  grid-row-end: 3;
}
```

Esses comando também podem ser escritos da seguinte forma:

```
/* Código CSS de exemplo */
.item-1{
  grid-column: 1 / 3;
}

.item-1{
  grid-row: 1 / 3;
}
```

Ou, ainda, dessa forma:

```
/* Código CSS de exemplo */
.item-1{
  grid-column: 1 / span 2;
}

.item-1{
  grid-row: 1 / span 2;
}
```

A `grid-area` propriedade pode ser usada como uma propriedade abreviada para as propriedades `grid-row-start`, `grid-column-start`, `grid-row-end` e `grid-column-end`.

```
/* Código CSS de exemplo */
.item-1{
  grid-area: 1 / 2 / 4 / 3;
}
```

Outra maneira de organizar os itens em uma grade é usar a propriedade `grid-template-areas` e dar um nome aos itens.

```
/* Código CSS de exemplo */
.container{
  display: grid;
```

```
grid-template-columns: repeat(3, 200px);
grid-template-areas: "item-1 item-1 item-1" "item-2 item-2 item-3" "item-4 item-5
item-6" "item-7 item-7 item-7";
gap: 10px;
}

.items{
border-radius: 5px;
padding: 10px;
background-color: orange;
border: 2px solid black;
}

.item-1{
grid-area: item-1;
}

.item-2{
grid-area: item-2;
}

.item-3{
grid-area: item-3;
}

.item-4{
grid-area: item-4;
}

.item-5{
grid-area: item-5;
}

.item-6{
grid-area: item-6;
}
```

Outros exemplos

```
/* Código CSS de exemplo */
.container{
display: grid;
grid-template-columns: repeat(3, 200px);
grid-template-areas: "item-1 item-1 item-2" "item-3 item-4 item-4" "item-3 item-5
item-6";
gap: 10px;
}
```

```
/* Nesse exemplo, um "." foi utilizado na terceira linha para representar uma célula vazia */
.container{
  display: grid;
  grid-template-columns: repeat(3, 200px);
  grid-template-areas: "item-1 item-1 item-1" "item-2 item-3 item-4" "item-5 . item-6";
  gap: 10px;
}
```

Atividades

Utilizando Grids CSS, reproduza as imagens a seguir.

Imagem 1

Exemplos Grid

Outros conteúdos

Lorem ipsum dolor sit amet consectetur adipisicing elit. Assumenda laudantium reprehenderit laboriosam tenetur atque accusantium.

Conteúdo principal

Lorem ipsum dolor sit amet consectetur adipisicing elit. Nisi aperiam architecto culpa aut eum corrupti laboriosam sit, laborum suscipit rerum doloremque fugiat unde autem enim dolorum ipsum libero quo sequi, dolore quos. Repudiandae, mollitia dolore. Enim laboriosam hic sapiente, sint dignissimos quae. Eius adipisci cum sunt obcaecati numquam! Recusandae, facere.

Contato: eu@meusite.com

Imagem 2



Imagem 3

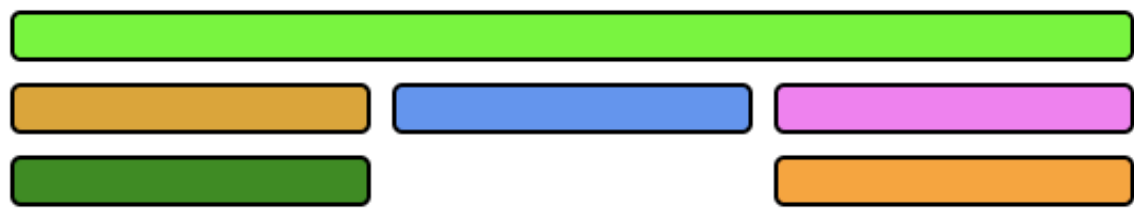


Imagem 4

