



escola
britânica de
artes criativas
& tecnologia

Profissão: Engenheiro Front-End



Fundamentos do CSS Responsivo



GUIA DA AULA 6



Construa um layout com CSS Grid

- Display Grid
- Grid-template-columns
- Espaçamento



Acompanhe aqui
os temas que
serão tratados
na videoaula



Display grid

Com o valor “grid” para a propriedade CSS display podemos construir *layouts* de uma forma bem simples, até mesmo *layouts* responsivos.

O layout construído com o display Grid é dividido em **colunas** e **linhas**.

Para usar display Grid é necessário estilizar o *container* da página com a regra display: grid; e além disso definir o *layout* que será criado, fazemos isso com a propriedade: grid-template-columns.



Exemplo:

```
.container{  
  display: flex;  
  grid-template-columns: 1fr 1fr 1fr;  
}
```



Grid-template-columns

A unidade de medida que utilizamos no exemplo anterior, fr, significa **fração**, se tivéssemos apenas um valor fr (grid-template-columns: 1fr) teríamos um *layout* com apenas uma coluna.

Além do fr podemos utilizar porcentagem, pixels e auto. Exemplo:

10% 50% auto // auto seria igual à 40%

120px auto 50% // auto seria igual à 100% - (120px + 50%)



#1

#2

#3



Espaçamento

Podemos adicionar espaçamento ao *layout*, chamamos isso de *gap*. Para adicionar espaçamento às colunas utilizamos a propriedade **column-gap**, que recebe o valor em *pixels* ou percentual.

É importante levar em conta o espaçamento no momento em que se define o tamanho das colunas. Exemplo:

```

.container {
  display: grid;
  grid-template-columns: 25% 25% 25% 25%;
  column-gap: 1%;
}
  
```



Teríamos 4 colunas com o espaçamento de 1%, o que faria com que o layout fosse maior que o container, neste exemplo o tamanho ideal para as colunas seria de 24,25%.

100% - 3% (temos 3 divisões, entre as colunas) = 97

97/4=24,25 que é o tamanho ideal para cada coluna.

Para espaçar as linhas usamos a propriedade **row-gap**, neste caso não precisamos nos preocupar com o container, afinal não limitamos a altura do *layout*.



