

```

<!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">
<link rel="stylesheet" href="../Examples/normalize.css">
<link rel="stylesheet" href="../Examples/reset.css">
<title>GRID-TEMPLATE-COLUMNS - FORMATANDO AS COLUNAS DE UM GRID</title>
<style>

```

```

/*      GRID-TEMPLATE-COLUMNS:

```

A propriedade "grid-template-columns" é sempre usada em uma tag que seja grid container, essa propriedade é que define qual será a largura de cada linha do grid, e essas colunas são identificadas pelos nomes atribuídos as "grid-areas" e elas devem ser por assim dizer "desenhadas" como valor das propriedades "grid-template-areas" e "grid-template-columns".

Essa propriedade só aceita valores de medida para atribuir a largura das colunas, como pixels, vw, vh e outras, temos que atribuir valores diferentes para cada "grid-area" que tivermos, mas se colocarmos um valor único e as colunas já tiverem sido desenhadas na "grid-template-areas" esse valor único será atribuído a todas as colunas...

```

*/

```

```

#template { /*Aqui temos a formatação do template de fora, esse não nos interessa no momento*/
  color: white;
  display: grid;
  grid-template-areas:
    "cabecalho"
    "conteudo"
    "rodape";
  grid-template-columns: 60vw;
  grid-template-rows: 10vh 20vh;
}

```

```

.cabecalho {
  grid-area: cabecalho;
}

```

```
background-color: red;
}

.conteudo { /*Esse é o grid container que nos interessa*/
  grid-area: conteudo; /*Veja que toda a formatação será feita dentro do conteudo que ficará verde na página*/
  background-color: green;
  display: grid; /*Para ser um container ele tem que ter um display grid...*/
  grid-template-areas: "coluna1 coluna2 coluna3"; /*Veja que tivemos que por assim dizer "desenhar" as 3 colunas*/
  grid-template-columns: auto; /*veja que, como as colunas já forem "desenhadas" no "grid-template-areas" nós*/
  grid-template-rows: 50%; /*só precisamos colocar um único valor "auto" no "grid-template-columns" para*/
  grid-gap: 1%; /*para que o valor fosse atribuído sobre todas as colunas. E temos um "grid-gap"*/
} /*de 1% só para ver onde as colunas se separam*/

.rodape {
  grid-area: rodape; /*Veja que o nome atribuído a essa "grid-area" é o mesmo que encontramos no */
  background-color: blue; /*"grid-template-areas"*/
}

.coluna1 {
  grid-area: coluna1;
  background-color: pink;
  color: black;
}

.coluna2 {
  grid-area: coluna2;
  background-color: pink;
  color: black;
}

.coluna3 {
  grid-area: coluna3;
  background-color: pink;
  color: black;
}
```

```
</style>
</head>
<body id="template">

  <header class="cabecalho">
    Aqui é o cabeçalho
  </header>
  <main class="conteudo">
    Aqui é o conteúdo
    <div class="coluna1">Aqui temos a coluna 01</div>
    <div class="coluna2">Aqui temos a coluna 02</div>
    <div class="coluna3">Aqui temos a coluna 03</div>
  </main>
  <footer class="rodape">
    Aqui é o rodapé
  </footer>

</body>
</html>
```

RESULTADO NO NAVEGADOR LOGO ABAIXO...

WhatsApp x PROGRAMAÇÃO 2022 - Planilha x GRID-TEMPLATE-COLUMNS - FO x CSS Grid – Um Guia Interativo (P x +

127.0.0.1:5500/Propriedades/127-grid-template-columns__Definindo_as_colunas_de_um_grid.html

Icons: [various application icons]

Aqui é o cabeçalho		
Aqui temos a coluna 01	Aqui temos a coluna 02	Aqui temos a coluna 03
Aqui é o conteúdo		
Aqui é o rodapé		

Windows taskbar: Search bar (Digite aqui para pesquisar), Taskbar icons (Clock, File Explorer, Calculator, etc.), System tray (16°C, 06:41, 02/06/2022)