

```
<!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 - ATRIBUINDO UM GRID LAYOUT A UMA TAG</title>
<style>
```

```
/*      GRID:
```

O valor grid é um valor que pode ser aplicado ao display de uma tag container para que possamos operar sobre as tags dentro dela gerando formatação de grid.

Esse valor apenas habilita a função de grid sobre uma tag container, para que os elementos possam ser operados em grid é preciso usar as demais propriedades grid.

```
*/

#template {
  color: white;
  display: grid; /*Aqui temos o exemplo do uso de um valor grid no display...*/
  grid-template-areas: /*veja que só o valor grid não trabalha sozinho é necessário usar um grid-template-areas*/
    "cabecalho"          /*para que o container possa reconhecer as áreas grid...*/
    "conteudo"
    "rodape";
  grid-template-columns: 60vw; /*Bem como temos que usar os "grid-template.." columns e rows para delimitar a */
  grid-template-rows: 10vh auto 10vh; /*quantidade de linhas e colunas que um grid container pode ter.*/
}

.cabecalho {
  grid-area: cabecalho; /*uma grid-area reconhecida pelo grid-template-areas*/
  background-color: red;
}
```

```
.conteudo {
  grid-area: conteudo; /*uma grid-area reconhecida pelo grid-template-areas*/
  background-color: green;
}

.rodape {
  grid-area: rodape; /*uma grid-area reconhecida pelo grid-template-areas*/
  background-color: blue;
}

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

  <header class="cabecalho">
    Aqui é o cabeçalho
  </header>
  <main class="conteudo">
    Aqui é o conteúdo
  </main>
  <footer class="rodape">
    Aqui é o rodapé
  </footer>

</body>
</html>
```

RESULTADO NO NAVEGADOR LOGO ABAIXO...

