

A diagram illustrating a CSS Grid layout. It features a grid of 6 columns and 2 rows. The top row contains three orange blocks (each 2 columns wide), a brown block (1 column wide), and three green blocks (each 2 columns wide). The bottom row contains two orange blocks (each 3 columns wide) and three green blocks (each 2 columns wide). The text "Layouts responsivos com CSS Grid" is overlaid on the grid.

# Layouts responsivos com CSS Grid

# Quem sou eu?

- Professor, Web Developer e Mozillian!
- 2006 - Comecei a trabalhar com desenvolvimento web;
- 2014 - Me juntei a Mozilla e comecei a lecionar;
- 2017 - Iniciei minhas atividades no Mozilla Tech Speakers;
- 2019 - Comecei o mestrado em Ensino e Processos Formativos;



# CSS Grid

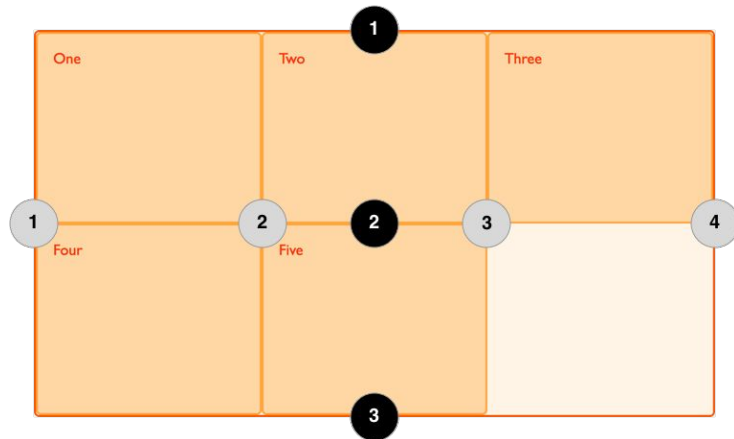
- **MDN** apresenta o CSS Grid como:

“CSS Grid Layout introduz um sistema bi-dimensional de grid (literalmente "grades") para CSS. Grids podem ser usados para o design de layouts de grandes seções de uma webpage, assim como de pequenos elementos de interface.”



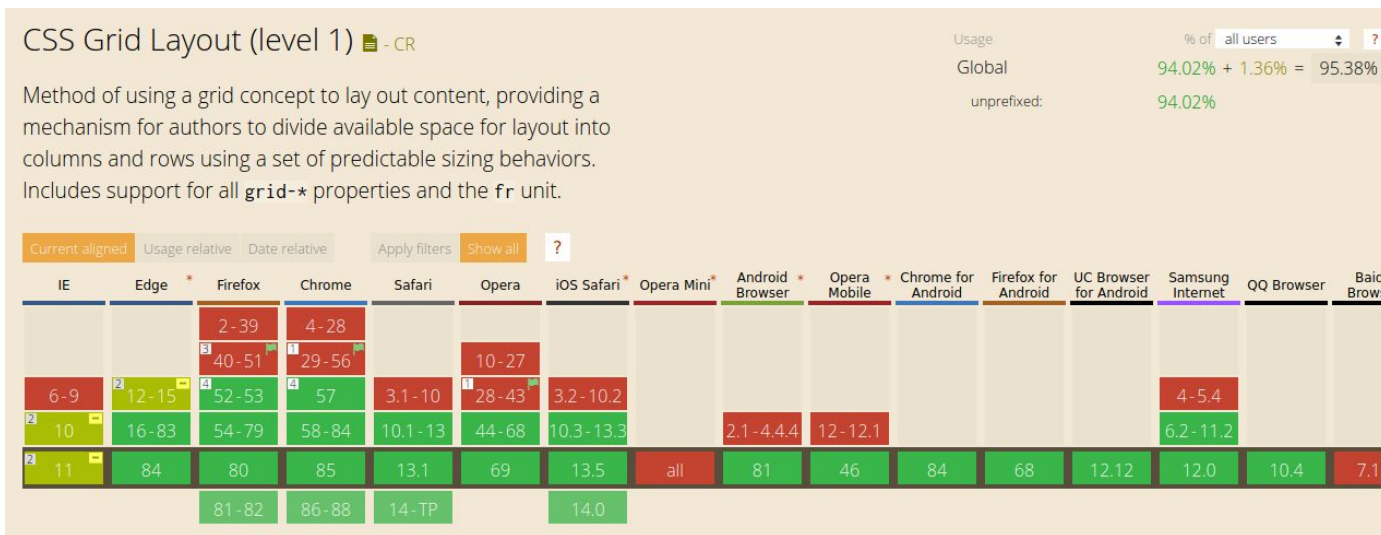
# Alguns recursos do CSS Grid

- **Dimensões fixas ou flexíveis**
- **Posicionamento de itens**
- **Alinhamento**
- **Controle sobre conteúdos sobrepostos**



# Grid + Flex = ❤️

- Grid entra para a criação do layout completo
- Flexbox entra para as áreas/células da Grid

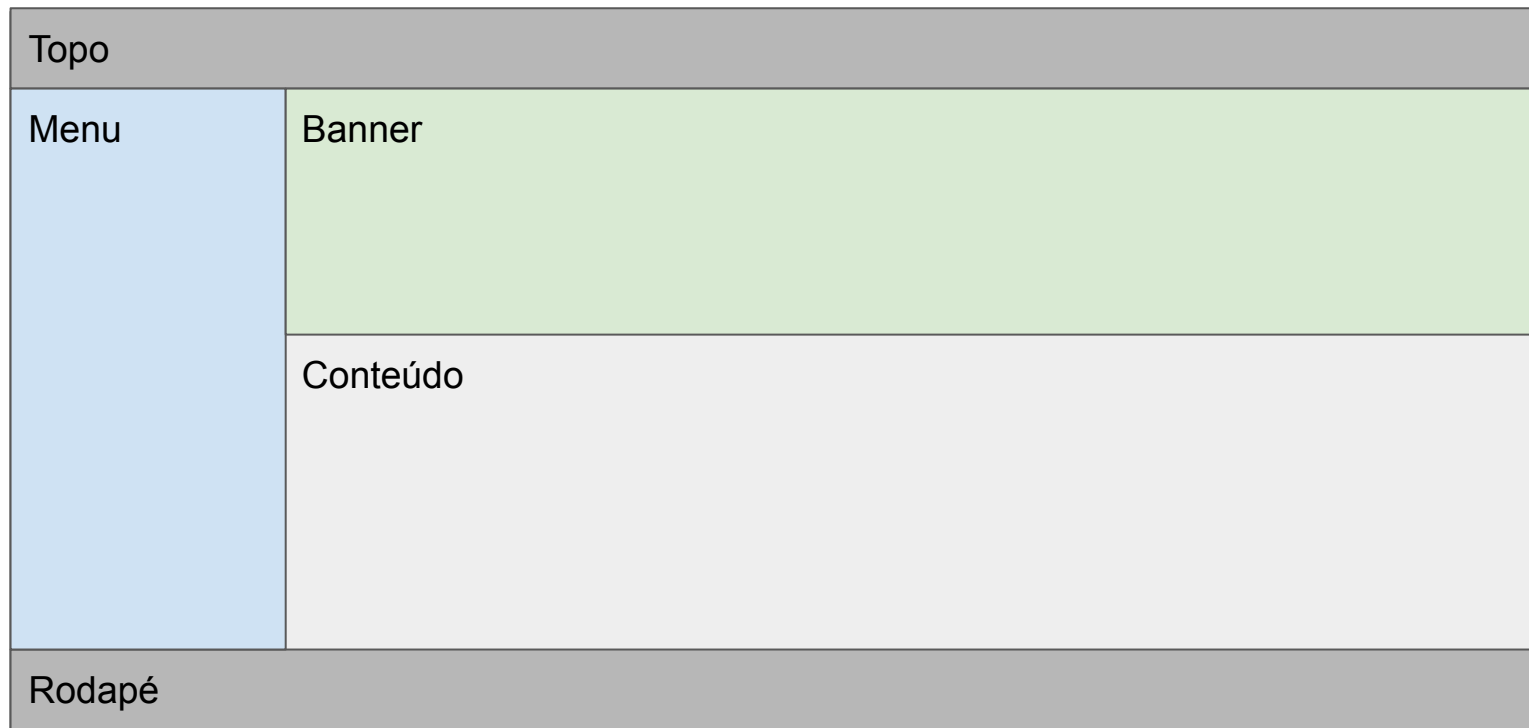


# Por onde eu começo?

```
<body>  
  <header class="topo">Topo</header>  
  <nav class="menu">Menu</nav>  
  <section class="banner">Banner</section>  
  <section class="conteudo">Conteúdo</section>  
  <footer class="rodape">Rodapé</footer>  
</body>
```



# Ideia de layout



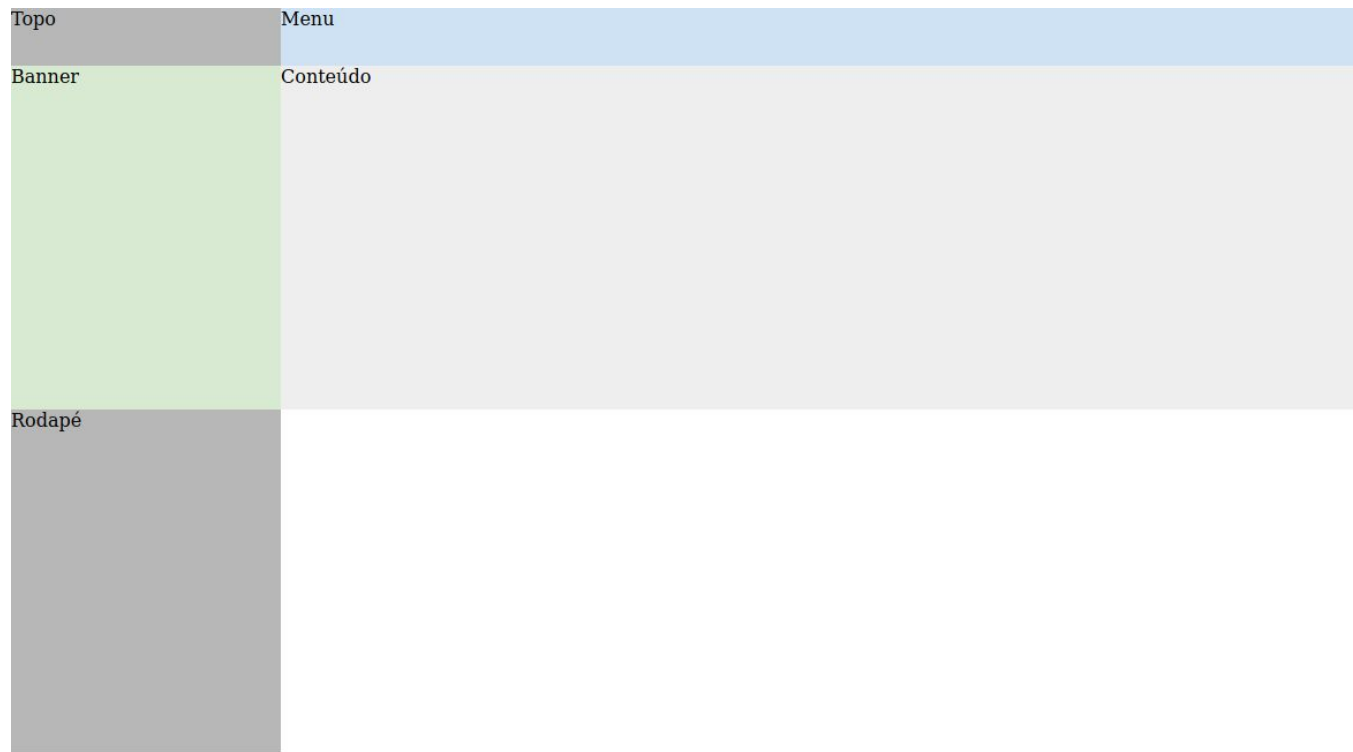
# Código CSS inicial

```
body{  
    display: grid;  
    grid-template-rows: 50px auto auto 50px;  
    grid-template-columns: 1fr 4fr;  
}
```





# Está meio quebrado né?



# Configurando as posições

```
.topo{  
    background-color: #b7b7b7;  
    grid-row: 1/2;  
    grid-column: 1/3;  
}
```

```
.menu{  
    background-color: #cfe2f3;  
    grid-row: 2/4;  
    grid-column: 1/2;  
}
```

```
.rodape{  
    background-color: #b7b7b7;  
    grid-row: 4/5;  
    grid-column: 1/3;  
}
```



# Não é Feitiçaria, é Tecnologia!



# Tem outras formas de fazer?

```
body{  
    display: grid;  
    grid-template-rows: 50px auto auto 50px;  
    grid-template-columns: 1fr 4fr;  
    grid-template-areas:  
        "topo topo"  
        "menu banner"  
        "menu conteudo"  
        "rodape rodape";  
}
```



# Tem outras formas de fazer?

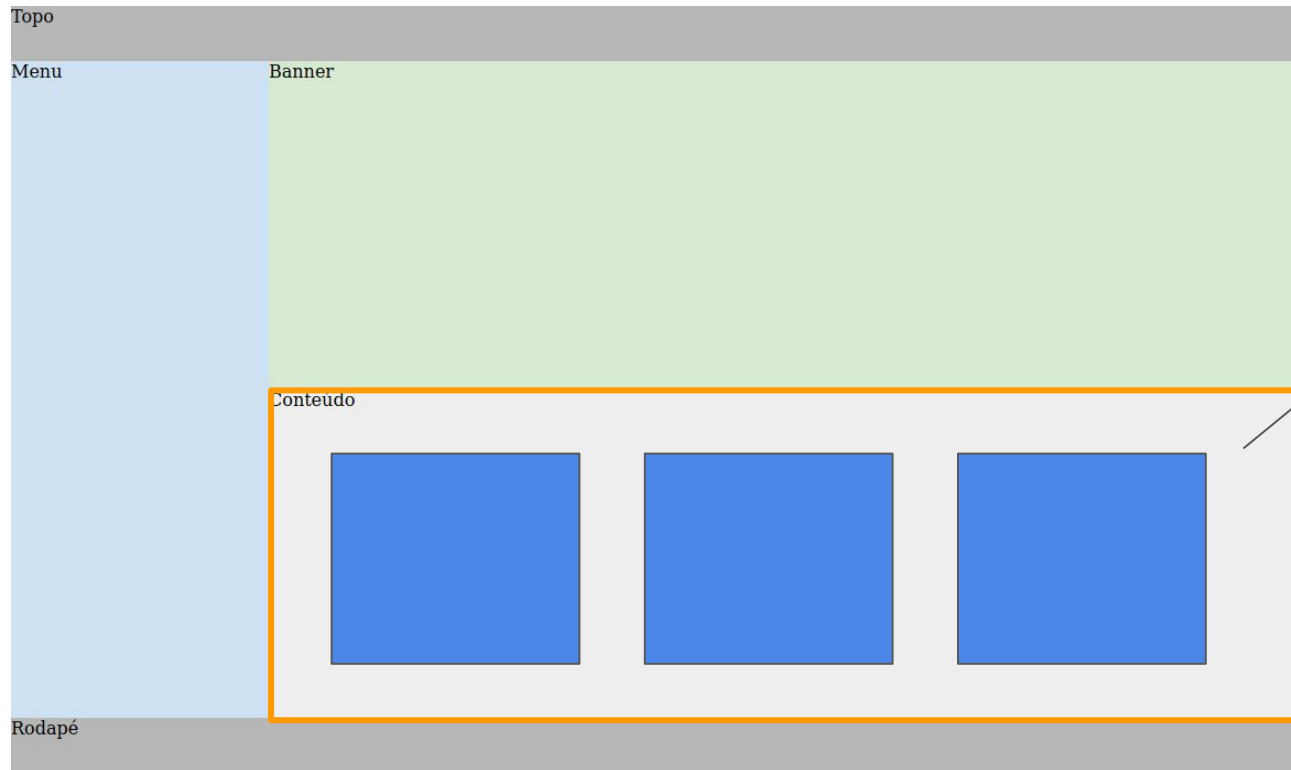
```
.topo{  
    background-color: #b7b7b7;  
    grid-area: topo;  
}  
  
.menu{  
    background-color: #cfe2f3;  
    grid-area: menu;  
}
```

```
.rodape{  
    background-color: #b7b7b7;  
    grid-area: rodape;  
}  
  
.banner{  
    background-color: #d9ead3;  
    grid-area: banner;  
}
```

# Não é Feitiçaria, é Tecnologia!



# Onde poderia usar o Flexbox?



Flexbox!

# E se o tamanho da tela mudar?

```
@media( .... ){  
  body{  
    grid-template-rows: 50px auto auto auto 50px;  
    grid-template-columns: 1fr;  
    grid-template-areas:  
      "topo"  
      "menu"  
      "banner"  
      "conteudo"  
      "rodape";  
  }  
}
```





# Links

- <https://felipez3r0.github.io/css-grid-puzzle/>
- <https://css-tricks.com/snippets/css/complete-guide-grid/>
- <https://css-tricks.com/grid-level-2-and-subgrid/>



# Fique conectado

- @felipez3r0 no [GitHub](#), [Twitter](#)
- facebook.com/felipez3r0
- twitch.tv/felipez3r0
- E-mail: [felipe@2ilabs.com.br](mailto:felipe@2ilabs.com.br)



# Dúvidas?



Muito obrigado!

