





Colunas



Podemos usar a propriedade ***column-count*** e especificar o número de colunas que quisermos.

Geralmente usamos somente em grandes quantidades de textos exibidas em telas grandes (para versão desktop de nosso site)



<> index.html

estilos.css 6 X

SITE TURMA PRONTO > css > # estilos.css > p

```
158 }
159
160 p{
161     font-size: clamp(1.2em, 1em+1vw, 1.4em);
162     line-height: 20px;
163     /* text-indent: 50px; */
164     text-align: justify;
165     font-family: "Montserrat", sans-serif;
166     font-optical-sizing: auto;
167     font-weight: 400;
168     font-style: normal;
169     column-count: 2;
170 }
171
172 #container-quem-somos{
173     display: flex;
174     flex-direction: row;
175     flex-wrap: wrap;
176     justify-content: center;
177     align-items: center;
178 }
```




Também podemos usar a propriedade ***column-width*** para determinar a largura da coluna e e ***column-gap*** para determinar a distância entre uma coluna e outra:

```
160  ✓  p{
161      font-size: clamp(1.2em, 1em+1vw, 1.4em);
162      line-height: 20px;
163      /* text-indent: 50px; */
164      text-align: justify;
165      font-family: "Montserrat", sans-serif;
166      font-optical-sizing: auto;
167      font-weight: 400;
168      font-style: normal;
169      column-count: 2;
170      column-width: 200px;
171      column-gap: 50px;
172
173  }
```



Gradientes



Define-se gradiente CSS como sendo uma transição de cores em um determinado espaço. Com uso de declarações CSS é possível ao autor definir se a transição será linear ou radial, em que quantidade de espaço será feita, bem como se duas ou mais cores participam do gradiente.



Diz-se que um gradiente é linear quando a transição das cores se faz segundo um eixo (linha reta) que possui um sentido (por exemplo: horizontal) e uma direção (por exemplo: da esquerda para a direita).



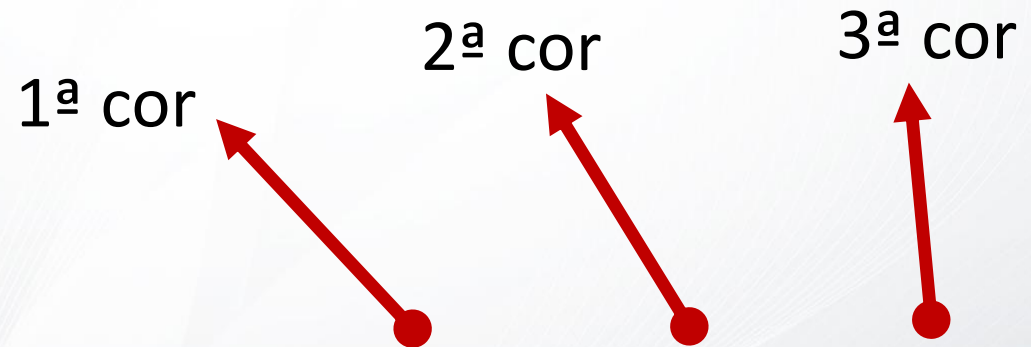
Sintaxe

1ª cor

2ª cor

```
✓ button{  
  width: 100px;  
  height: 30px;  
  margin-top: 10px ;  
  margin-right: 20px;  
  float: left;  
  background: linear-gradient(■ blue, ■ white);  
  color: ■ white;  
  box-shadow: 0 0 10px ■ rgb(211, 211, 211);  
  border: none;  
  border-radius: 10px;  
  cursor: pointer;  
  transition: all 0.3s;  
}
```

Sintaxe



```
background: linear-gradient(■blue, ■white, ■red);
```

Definindo direção

Para esquerda



```
.servicos{  
  background: linear-gradient(to left, blue, white, red);  
  padding-bottom: 50px;  
}
```


Para direita: **to right**

Para cima: **to top**

Para baixo: **to bottom**

Definindo direção


Ângulo de rotação (pode ser negativo para que seja rotacionado para o lado oposto)




A red arrow with a circular head points from the text "Ângulo de rotação" to the "45deg" value in the CSS code.

```
.servicos{  
  background: linear-gradient(45deg, blue, white, red);  
  padding-bottom: 50px;  
}
```


Quantidade de cor


Cada cor pode ter determinada quantidade com valor em %

A red arrow originates from a red dot on the "50%" value in the CSS code and points towards the text "com valor em %".

```
.servicos{  
  background: linear-gradient(45deg,  blue 50%,  white,  red);  
  padding-bottom: 50px;  
}
```

Quantidade de cor

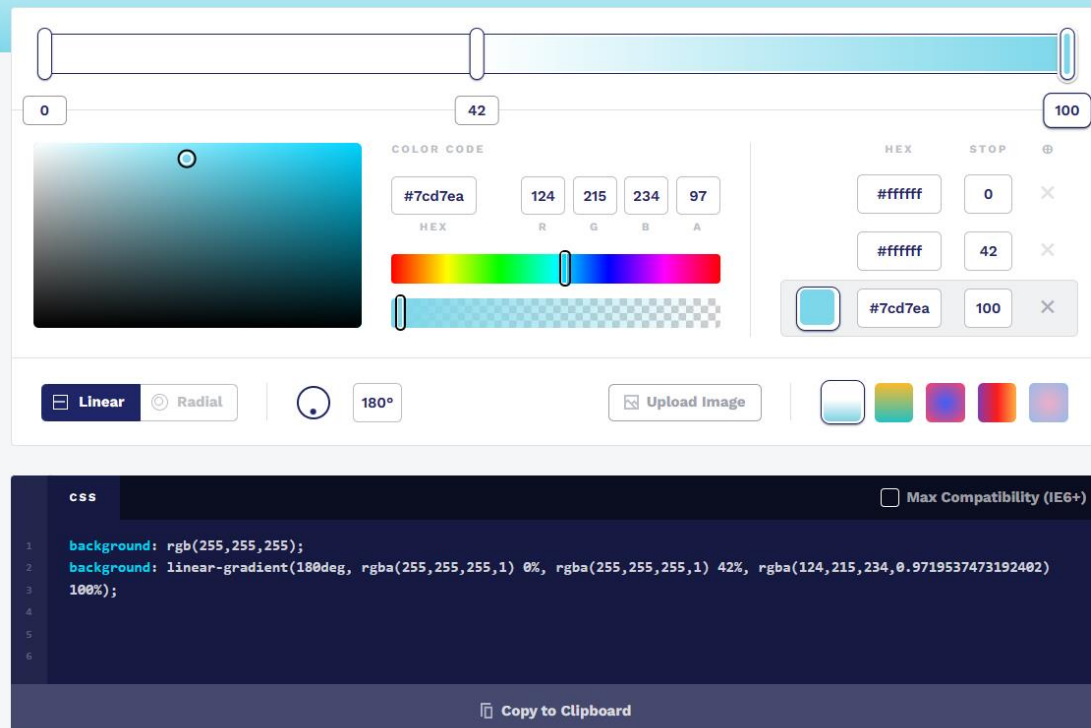
Podemos usar também a “cor” transparent

A red arrow with a circular head points from the word "transparent" in the code block to the word "transparent" in the text above.

```
.servicos{  
  background: linear-gradient(to bottom, blue , transparent);  
  padding-bottom: 50px;  
}
```

Dica: gerador de gradientes

<https://cssgradient.io/>



The image shows the CSS Gradient Generator interface. At the top, there is a horizontal gradient bar with a slider at 42%. Below the bar, the 'COLOR CODE' section displays the selected color as #7cd7ea, with its RGB values (124, 215, 234) and a small color picker. To the right, the 'HEX STOP' section shows three color stops: #ffffff at 0%, #ffffff at 42%, and #7cd7ea at 100%. Below this, the 'Linear' and 'Radial' tabs are visible, with 'Linear' selected. The '180°' angle is set. An 'Upload Image' button is present. At the bottom, the 'CSS' section shows the generated CSS code for the gradient, with a 'Copy to Clipboard' button.

```
css
1 background: rgb(255,255,255);
2 background: linear-gradient(180deg, rgba(255,255,255,1) 0%, rgba(255,255,255,1) 42%, rgba(124,215,234,0.9719537473192402) 100%);
3
4
5
6
```

Copy to Clipboard



Dicas para Responsividade de Elementos e Estruturas

Dicas

- ✓ Busque sempre que possível usar o display flex para o containers em sua estrutura.
- ✓ Use como visto na aula passada as unidades relativas novas para texto por exemplo.

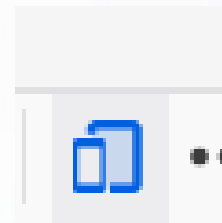


Dicas

- ✓ Use o Firefox para testar a compatibilidade de tela e tamanho de telas com o F12



Dicas



Resposivo 758 x 480 DPR: 1... Sem limitação UA: User agent personalizado

HOME QUEM SOMOS DESENVOLVEDORES CURSO GALERIA CONTATO

DEV3N222

Inspetor Console Debugger Rede Editor de estilos Desempenho Memória Armazenamento Acessibilidade Aplicação

Filtrar saída

Erros Warnings Logs Info

Modo de design responsivo (Ctrl+Shift+M)

downloadable font: kern: bad range shift (font-family: "batman" style:normal weight:400 stretch:100 src index:0) source: file:///D:/!%20AULAS/!%20DESENVOLVIMENTO%20WEB%20FRONT-END/!%20UC12/AULA%2019%20-%20CULUNAS,%20GRADIENTES,%20MEDIA%20QUERIES,%20MENU%20HAMBURGER/SITE%20TURMA%20PRONTO/fonts/batman.ttf

A ancoragem de rolagem foi desativada em um contêiner de rolagem devido a ajustes consecutivos em excesso (28) com distância total pequena demais (1.39345237186977 px em média, 39.0167 px no total).

Avisos de Content-Security-Policy

Acesso a cookies ou armazenamento particionado foi fornecido para "https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d3453.791615293096!2d-51.22146782473794!3d-30.042835674924394!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x951978567f17f28d%3A0x2c2c5272bacf4d3a!2sSenac%20Tech!5e0!3m2!1spt-BR!2sbr!4v1709065993336!5m2!1spt-BR!2sbr" porque é carregado em contexto de terceiros e o particionamento de estado dinâmico está ativado.

Search endpoint requested!

A ancoragem de rolagem foi desativada em um contêiner de rolagem devido a ajustes consecutivos em excesso (26) com distância total pequena demais (1.66217950674204 px em média, 43.2167 px no total).

Top

Dicas

- ✓ Vá percorrendo o site por seções (menu, banner, textos, galerias, formulários, tabelas, rodapés... e anote o tamanho em que surge o problema (pode ser somente o tamanho de largura).

Dicas

- ✓ Redimensione a janela e observe o tamanho de largura mudando



Responsivo ▾

758

×

480



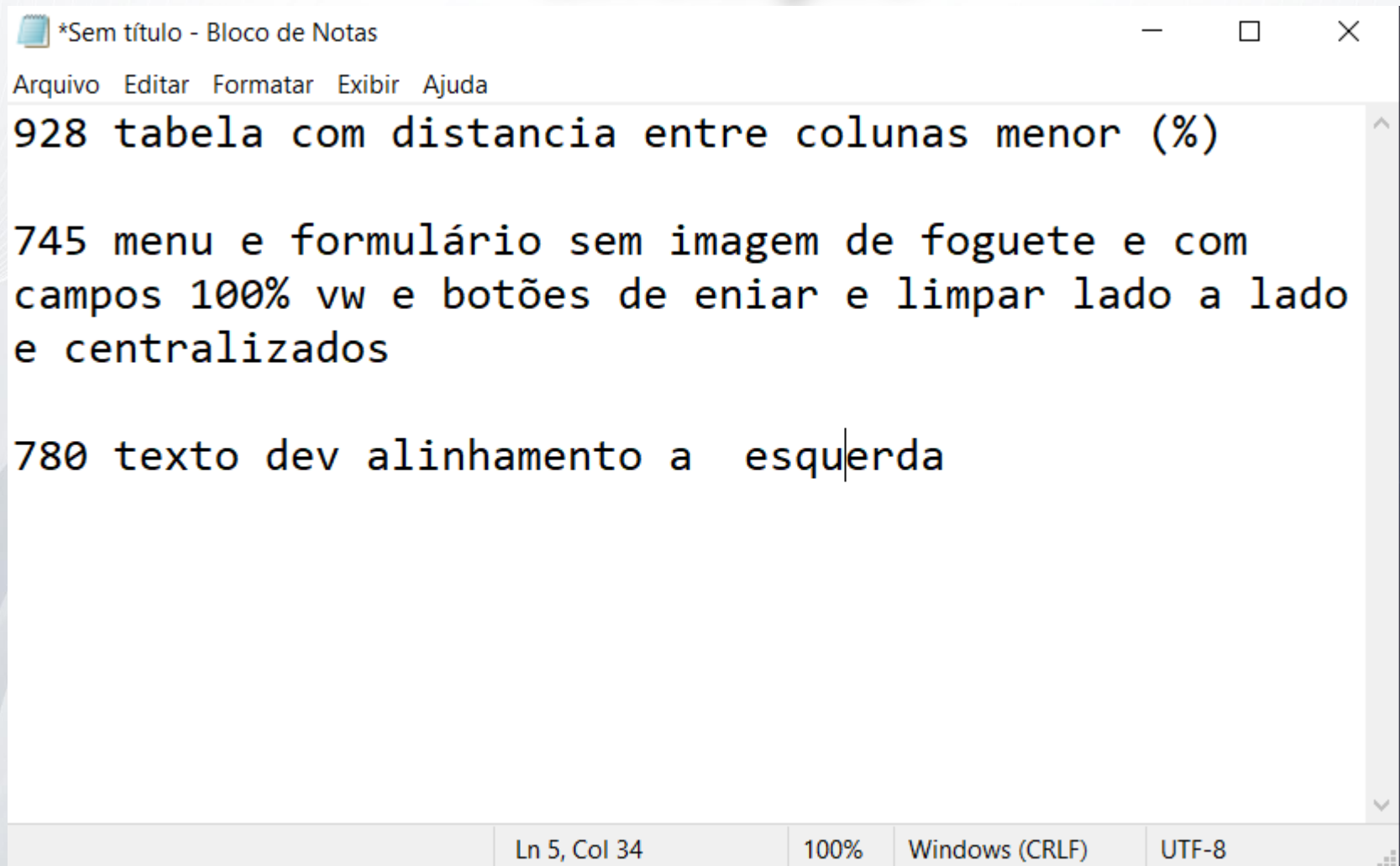
DPR: 1... ▾

Sem limitação ▾

UA: User agent personalizado



Exemplo

A screenshot of a Windows Notepad application window. The title bar reads "*Sem título - Bloco de Notas". The menu bar includes "Arquivo", "Editar", "Formatar", "Exibir", and "Ajuda". The text area contains three lines of code in a monospaced font: "928 tabela com distancia entre colunas menor (%)", "745 menu e formulário sem imagem de foguete e com campos 100% vw e botões de eniar e limpar lado a lado e centralizados", and "780 texto dev alinhamento a esquerda". The status bar at the bottom shows "Ln 5, Col 34", "100%", "Windows (CRLF)", and "UTF-8".

```
*Sem título - Bloco de Notas
Arquivo  Editar  Formatar  Exibir  Ajuda
928 tabela com distancia entre colunas menor (%)
745 menu e formulário sem imagem de foguete e com
campos 100% vw e botões de eniar e limpar lado a lado
e centralizados
780 texto dev alinhamento a esquerda
Ln 5, Col 34  100%  Windows (CRLF)  UTF-8
```




Media Queries



Media Queries

Aqui está a cereja do bolo da responsividade.

Podemos ter um CSS para cada tamanho de tela!

Ou seja, criar *breakpoints* e neles colocar regras diferentes para os elementos que quisermos (posição, cor, tamanhos...)



Media Queries

Isso é feito através das **media queries**, e a sintaxe lembra o que foi visto a pouco nas imagens com múltiplos *sources*.

Veja:

Sintaxe

```
@media (condição) {  
    /* Estilizações CSS */  
}
```

```
<style>
```

```
@media (max-width:938px){  
  table{  
    width:90%;  
    margin-left: 5%;  
    margin-right: 5%;  
  }  
  
  td{  
    width: 40%;  
  }  
}
```

```
</style>
```




Aqui iniciamos um conjunto de regras para o *breakpoint*

Aqui definimos o *breakpoint*

```
@media (max-width:938px){  
  table{  
    width:90%;  
    margin-left: 5%;  
    margin-right: 5%;  
  }  
  
  td{  
    width: 40%;  
  }  
}
```

Conjunto de regras para o *breakpoint*



```
<style>
```

```
@media (max-width:938px){
```

```
  table{
```

```
    margin-left: 5%;
```

```
    margin-right: 5%;
```

```
  }
```

```
  td{
```

```
    width: 40%;
```

```
  }
```

```
}
```

```
@media(max-width:780px){
```

```
  .container-devs p{
```

```
    text-align: left;
```

```
  }
```

```
  h3{
```

```
    margin-top: 0px;
```

```
    margin-bottom: 5px;
```

```
  }
```

```
}
```

```
</style>
```



```
    },
    @media(max-width:800px){
      .rocket{
        display:none;
      }
      form{
        width:90%;
        margin-left:10%;
        margin-right:5%;
      }
    }
  </style>
```

Observações

Nas condições das *medias queries*, usamos o **max-width** para colocar regras css **a partir desta condição**.

Podemos usar o **min-width** para criar regras css **até esta condição**.

Observações

Podemos assim criar faixas de valores para as condições:

```
@media (min-width: 769px) and (max-width: 1024px){  
    /* Estilizações CSS */  
}
```




Observações

Também é possível criar um css externo somente para um tamanho de tela:

```
<!-- CSS media query em um elemento de link -->  
<link rel="stylesheet" media="(max-width: 800px)" href="example.css" />
```



Observações

Existe o operador lógico ***and*** que podem ser para colocar condição adicionais:

CSS

```
@media (min-width: 700px) and (orientation: landscape) { ... }
```

Orientation: landscape (paisagem)

Orientation: portrait (retrato)



Observações

Existe o operador lógico ***and*** que podem ser para colocar condição adicionais:

CSS

```
@media (min-width: 700px) and (orientation: landscape) { ... }
```

Orientation: landscape (paisagem)

Orientation: portrait (retrato)



Observações

De acordo com o tipo de mídia:

```
@media only screen and (min-width: 480px) { }
```

Observações

Alguns tipo de mídia (principais):

all: este valor é usado para que o código CSS seja aplicado para todos os dispositivos.

braille: para dispositivos táteis.

handheld: para dispositivos de mão, celulares e outros dispositivos deste perfil. Normalmente com telas pequenas e banda limitada.



Observações

Alguns tipo de mídia (principais):

print: para impressão em papel.

projection: para apresentações, como PowerPoint. Este valor foi inventado pelo pessoal da Opera. MUITO útil.

screen: para dispositivos com telas coloridas e alta resolução.

Observações

speech: para sintetizadores de voz. O CSS 2 tem uma especificação de CSS chamada Aural (<https://www.w3.org/TR/CSS2/aural.html>) que podemos “formatar” a voz de leitores de tela e outros sintetizadores.

tv: para dispositivos como televisores, ou seja, com baixa resolução, com boa quantidade de cores e scroll limitado.



Ajustes de Menu



Fica uma dica estudo de frameworks CSS como **Bootstrap**. Ele produz menus responsivos de forma muito mais ágil e fácil, sem grandes adaptações de layout ou mesmo media queries.



Em nosso projeto experimental, podemos usar a media querie para alterar a visualização dos itens: em um determinado tamanho de tela:


```
@media(max-width:780px){  
  h3{  
    margin-bottom: 5px;  
  }  
  a{  
    font-size: 0.9em;  
  }  
  ul li{  
    display: flex;  
    justify-content: center;  
    margin: -10px;  
  }  
}
```



É interessante reajustar o submenu também para que ele apareça de forma deslocada e não corte/oculte nenhum subitem:



```
.submenu:hover .submenu-item {  
  top: -120px;  
  left: 20px;  
}
```