



CSS

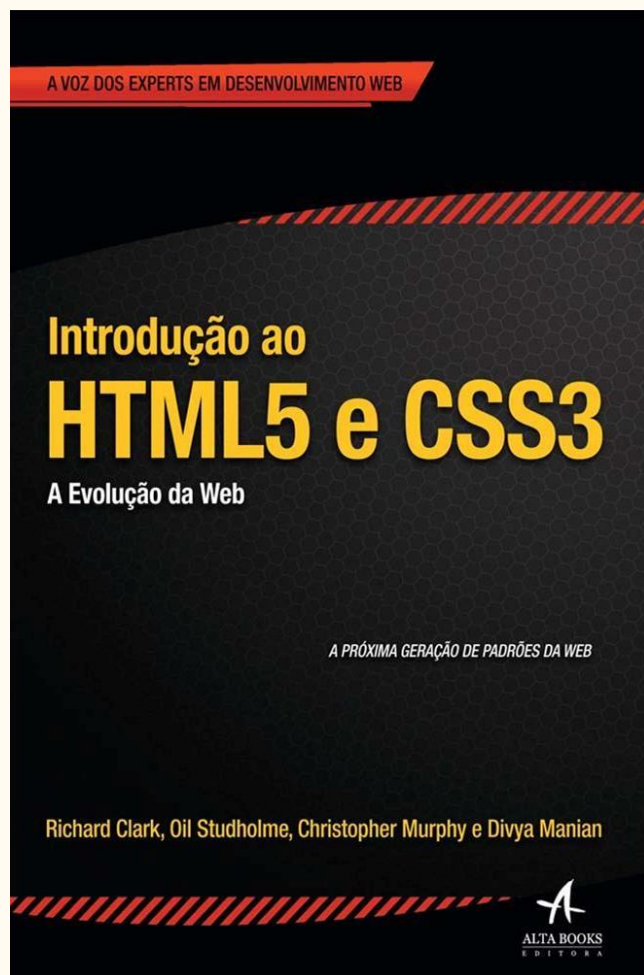
—

Prof. Victor Farias

v 1.1

Referências

w3schools.com



Introdução

CSS

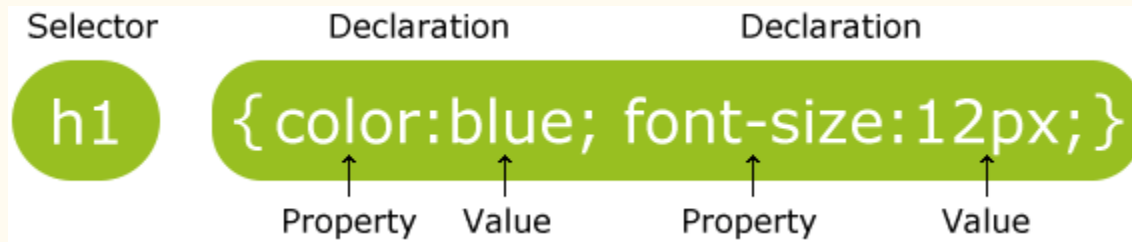
- CSS significa **C**ascading **S**tyle **S**heets
- Descreve como os elementos são renderizados na tela
- Folhas de estilo são salvas em arquivos CSS
- CSS define o layout da página
- HTML define apenas o conteúdo

CSS



Sintaxe

Uma folha de estilo CSS consiste em **conjunto de regras**:



- Seletores apontam para elementos HTML
- Cada declaração corresponde a uma propriedade CSS

Seletores CSS

- Seletor elemento
 - Seleciona pelo nome do elemento (todos)

```
p {  
  text-align: center;  
  color: red;  
}
```

- Seletor por id
 - Seleciona pelo atributo id dos elementos

```
#para1 {  
  text-align: center;  
  color: red;  
}
```

Seletores CSS

- Seletor por classe
 - Seleciona todos elementos de uma determinada classe

```
.center {  
    text-align: center;  
    color: red;  
}
```

- Agrupamento de seletores
 - Pode-se agrupar elementos que tem mesmas definições de estilo

```
h1, h2, p {  
    text-align: center;  
    color: red;  
}
```


Seletores CSS

- Seletores de elementos internos
 - Seleciona elemento que estão dentro de outros elementos
 - Exemplo: Selecionar todo elementos <p> que estão dentro de elementos <div>

```
div p {  
    text-align: center;  
    color: red;  
}
```

- Mais seletores:
 - http://www.w3schools.com/cssref/css_selectors.asp

CSS na Página

- Existem 3 modos de incluir CSS em uma página HTML
 - *Inline*
 - Folha de estilo interna
 - Folha de estilo externa
- O que ocorre se mais de um estilo for especificado para um mesmo elemento?
 - Vale o último estilo lido pelo navegador

Inline

- Estilo inline se aplicam a apenas um elemento
- Usa atributo *style* para definir propriedades CSS

```
<h1 style="color:blue;margin-left:30px;">
```

```
  This is a heading.
```

```
</h1>
```

- A grande desvantagem do *inline* é que se perde a separação entre conteúdo e apresentação -> mistura HTML com CSS

Folha de Estilo Interna

- Define estilo único de uma página
- Tag <style> dentro de <head>

```
<head>  
<style>  
body {  
    background-color: linen;  
}  
  
h1 {  
    color: maroon;  
    margin-left: 40px;  
}  
</style>  
</head>
```

Folha de Estilo Externa

- Pode mudar o estilo completo da página em apenas um arquivo
- Tag <link> para incluir folha de estilo CSS externa

index.html

```
<head>  
<link rel="stylesheet"  
type="text/css" href="mystyle.css">  
</head>
```

mystyle.css

```
body {  
    background-color: lightblue;  
}  
  
h1 {  
    color: navy;  
    margin-left: 20px;  
}
```

Unidades CSS

- Em CSS, existem dois tipos de unidades
 - Absolutas - comprimentos são fixos
 - cm - centímetros
 - mm - millimeters
 - in - polegadas (1in = 96px = 2.54cm)
 - px - pixels (1px = 1/96th de 1in)
 - pt - points (1pt = 1/72 of 1in)
 - pc - picas (1pc = 12 pt)

Unidades CSS

- Em CSS, existem dois tipos de unidades
 - Relativos - comprimentos relativos a outros comprimentos
 - em - Relative to the font-size of the element (2em means 2 times the size of the current font)
 - ex Relative to the x-height of the current font (rarely used)
 - ch Relative to width of the "0"
 - rem Relative to font-size of the root element
 - vw Relative to 1% of the width of the viewport
 - vh Relative to 1% of the height of the viewport
 - vmin Relative to 1% of viewport's* smaller dimension
 - vmax Relative to 1% of viewport's* larger dimension
 - %

Propriedades CSS



Background

- Propriedade **background-color**
 - Define cor de fundo de um elemento

```
body {  
    background-color: lightblue;  
}
```

- Propriedade **background-image**
 - Especifica imagem como background
 - Por padrão, a imagem é repetida até preencher o elemento todo

```
body {  
    background-image: url("paper.gif");  
}
```

Background

- Propriedade **background-repeat**

- Repete imagem horizontalmente ou verticalmente
- Valores: repeat-x, repeat-y, no-repeat

```
body {  
    background-image: url("gradient_bg.png");  
    background-repeat: repeat-x;  
}
```

- Propriedade **background-position**

- Posiciona imagem de background quando não se repete (**no-repeat**)

```
body {  
    background-image: url("img_tree.png");  
    background-repeat: no-repeat;  
    background-position: right top;  
}
```

Background

- **background-attachment: fixed**

- Fixa background no lugar ao rolar a página

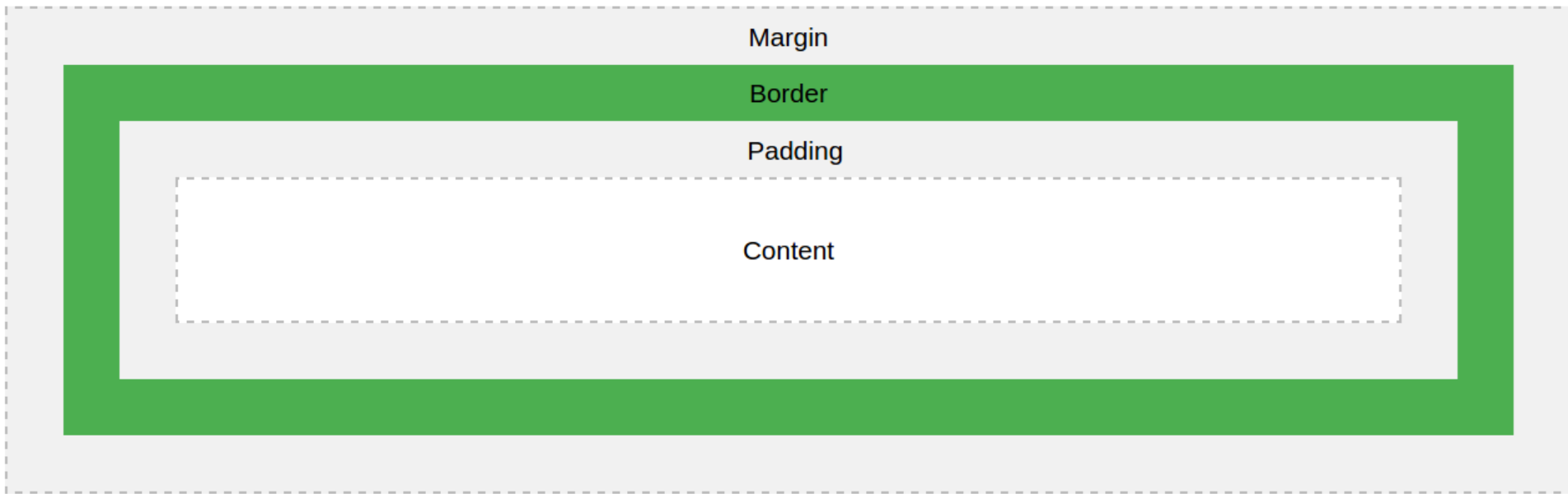
```
body {  
  background-image: url("img_tree.png");  
  background-repeat: no-repeat;  
  background-position: right top;  
  background-attachment: fixed;  
}
```

- **Propriedade background**

- Atalho para propriedades background
- Ordem: background-color, background-image, background-repeat, background-attachment, background-position

```
body {  
  background: #ffffff url("img_tree.png") no-repeat right top;  
}
```

Modelo de Caixa CSS



Content: O conteúdo da caixa

Padding: Espaço ao redor do conteúdo. Padding transparente

Border: É desenhada em torno do padding e do conteúdo

Margin: Margem para elementos ao lado. A margem é transparente.

Modelo de Caixa CSS

- Ao definir Width e Height dos elementos, estamos apenas definindo largura e altura da área do conteúdo
 - Para calcular a área completa do elemento, devemos considerar também padding, border e margin

```
div {  
    width: 320px;  
    padding: 10px;  
    border: 5px solid gray;  
    margin: 15px;  
}
```

Largura total = width + (padding esquerdo e direito) + (margin esquerdo e direito)
+ (border esquerdo e direito) = $320 + 2 \cdot 10 + 2 \cdot 5 + 2 \cdot 15 = 380$

Bordas

- Propriedade **border-style**

- Especifica qual o tipo de borda
- Valores
 - dotted - linha pontilhada
 - dashed - linha tracejada
 - solid - linha sólida
 - double - linha dupla
 - groove - 3D grooved border
 - ridge - 3D ridged border
 - inset - 3D inset border
 - outset - 3D outset border
 - none - sem borda
 - hidden - borda escondida
- Pode colocar 1 a 4 valores (top border, right border, bottom border, and the left border)

```
p {  
    border-style: dotted solid dotted solid;  
}
```

Bordas

- Propriedade **border-width**

- Largura da borda
- 1 a 4 valores em ordem: top border, right border, bottom border, and the left border

```
p.three {  
    border-style: solid;  
    border-width: 2px 10px 4px 20px;  
}
```

- Propriedade **border-color**

- Cor da borda
- 1 a 4 valores em ordem: top border, right border, bottom border, and the left border

```
p.three {  
    border-style: solid;  
    border-color: red green blue yellow;  
}
```

Bordas

- Propriedade **border**

- Atalho para as propriedades border-width, border-style (requerido), border-color

```
p {
```

```
    border: 5px solid red;
```

```
}
```

```
p {
```

```
    border-left: 6px solid red;
```

```
    background-color: lightgrey;
```

```
}
```


Margens

- Propriedades **margin-top**, **margin-right**, **margin-bottom** e **margin-left**
 - Valores:
 - **auto** - o navegador calcula a margem para centralizar o elemento
 - tamanho em px, pt, cm, etc.
 - % da largura do elemento que o contém
 - **inherit** - a margem é herdada do elemento pai
 - Dica: valores negativos são permitidos

```
p {  
    margin-top: 100px;  
    margin-bottom: 100px;  
    margin-right: 150px;  
    margin-left: 80px;  
}
```

Margens

- Propriedade **margin**

- Atalho para margin-top, margin-right, margin-bottom e margin-left (nessa ordem)

```
p {  
    margin: 100px 150px 100px 80px;  
}
```

Padding

- Padding gera espaço ao redor do conteúdo
- Propriedade **padding-top**, **padding-right**, **padding-bottom** e **padding-left**
- Valores:
 - tamanho em px, pt, cm, etc.
 - % da largura do elemento que o contém
 - **inherit** - a margem é herdada do elemento pai

```
p {  
    padding-top: 50px;  
    padding-right: 30px;  
    padding-bottom: 50px;  
    padding-left: 80px;  
}
```

Padding

- Propriedade **padding**
- Atalho para padding-top, padding-right, padding-bottom e padding-left

```
p {  
    padding: 50px 30px 50px 80px;  
}
```

Height e Width

- Propriedades **height** e **width** definem a altura e a largura do elemento
- Valores:
 - tamanho em px, pt, cm, etc.
 - % da largura do elemento que o contém

```
div {  
    height: 200px;  
    width: 50%;  
    background-color: powderblue;  
}
```

Height e Width

- Propriedade **max-width**
 - Define largura máxima
 - Caso janela seja menor que largura máxima, o elemento diminui de tamanho
- Valores:
 - tamanho em px, pt, cm, etc.
 - % da largura do elemento que o contém

```
div {  
    max-width: 500px;  
    height: 100px;  
    background-color: powderblue;  
}
```

Texto

- Propriedade **color**

- Muda cor do texto

```
h1 {  
    color: green;  
}
```

- Propriedade **text-align**

- Define alinhamento horizontal do texto
- Valores: center, left, right, justify

```
h1 {  
    text-align: center;  
}
```

Texto

- Propriedade **text-decoration**
 - Define decorações no texto
 - Valores: none, overline, line-through e underline

```
h1 {  
    text-decoration: overline;  
}
```

- Propriedade **text-transform**
 - Especifica case das letras
 - Valores: uppercase, lowercase e capitalize

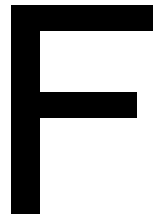
```
p.lowercase {  
    text-transform: lowercase;  
}
```


Fontes

- Existem dois tipos de famílias de fonte em CSS
 - Família genéricas: grupo de famílias de fonte que parecem similar
 - Família da fonte: uma família específica de fonte

Generic family	Font family	Description
Serif	Times New Roman Georgia	Serif fonts have small lines at the ends on some characters
Sans-serif	Arial Verdana	"Sans" means without - these fonts do not have the lines at the ends of characters
Monospace	Courier New Lucida Console	All monospace characters have the same width

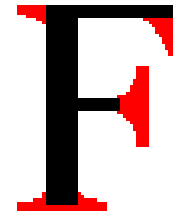
Fontes



Sans-serif



Serif



Serif
(red serifs)

Fontes

- Propriedade **font-family**

- Define vários nomes de fontes
- Se o navegador não suportar a primeira fonte, ele tenta a segundo e assim em diante
- Melhor usar a fonte desejada no começo e depois colocar famílias de fontes genéricas para o navegador escolher a que tiver suporte

```
p {  
    font-family: "Times New Roman", Times, serif;  
}
```

Fontes

- Propriedade **font-size**
 - Define tamanho do texto
 - Tamanho padrão da fonte é **16px**

```
h1 {  
    font-size: 40px;  
}  
  
h2 {  
    font-size: 1.875em; /* 30px/16=1.875em */  
}
```

Links

- Links podem ser estilizados com propriedades CSS
 - color, font-family, background

```
a {  
  color: hotpink;  
}
```

Links

- Links podem ser estilizados segundo seus estados
 - a:link - um link no visitado
 - a:visited - um link visitado
 - a:hover - um link quando o mouse passa por cima
 - a:active - um link no momento que é clicado

```
a:link {  
    color: red;  
}
```

```
a:visited {  
    color: green;  
}
```

```
a:hover {  
    color: hotpink;  
}
```

```
a:active {  
    color: blue;  
}
```

OBS: a:hover tem que vir depois de a:link e a:visited. a:active deve vir depois de a:hover

Propriedades width, max-width e margin: auto

- Elementos bloco usam toda a largura disponível
 - width previne que o bloco use toda a largura
 - quando largura da página é menor que width, a página quebra
 - max-width fornece uma solução melhor
 - quando a largura da página é menor que max-width, o elemento diminui de largura
 - margin: auto distribui a sobra de espaço horizontalmente igualmente entre a margem esquerda e direita para centralizar elemento

```
div.ex1 {  
    width: 500px;  
    margin: auto;  
    border: 3px solid #73AD21;  
}
```

```
div.ex2 {  
    max-width: 500px;  
    margin: auto;  
    border: 3px solid #73AD21;  
}
```

Propriedade Position

- Propriedade **position** define o método de posicionamento do elemento
- Temos quatro valores:
 - static
 - relative
 - fixed
 - absolute

position: static;

- Valor padrão de **position** é static
- Posiciona os elementos com o fluxo normal da página

```
div.static {  
    position: static;  
    border: 3px solid #73AD21;  
}
```

position: relative;

- O posicionamento é relativo a sua posição normal
- Propriedades top, right, bottom, and left vão ajustar a posição relativo com a posição normal

```
div.relative {  
    position: relative;  
    left: 30px;  
    border: 3px solid #73AD21;  
}
```

position: fixed;

- O elemento é posicionado segundo a *viewport*
- Não deixa espaço onde ele estaria posicionado normalmente

```
div.fixed {  
    position: fixed;  
    bottom: 0;  
    right: 0;  
    width: 300px;  
    border: 3px solid #73AD21;  
}
```

position: absolute;

- O elemento é posicionado de acordo com o ancestral mais próximo
- Se o elemento não tem ancestrais posicionados, ele usa o body como base

```
div.relative {  
    position: relative;  
    width: 400px;  
    height: 200px;  
    border: 3px solid #73AD21;  
}
```

```
div.absolute {  
    position: absolute;  
    top: 80px;  
    right: 0;  
    width: 200px;  
    height: 100px;  
    border: 3px solid #73AD21;  
}
```

Propriedade overflow

- A propriedade overflow especifica se o conteúdo vai ser cortado ou vai adicionar um *scrollbar* quando conteúdo é muito grande
- Valores:
 - visible - Padrão. O conteúdo renderiza fora da caixa
 - hidden - O conteúdo é cortado e o excedente fica invisível
 - scroll - O scroll é adicionado dentro da caixa mesmo que não seja necessário
 - auto - Só adiciona scroll quando necessário
- Propriedades overflow-x e overflow-y especificam com o scroll devem se comportar em cada eixo

```
div {  
  
    overflow-x: hidden; /* Hide horizontal scrollbar */  
  
    overflow-y: scroll; /* Add vertical scrollbar */  
  
}
```

Propriedade float

- Propriedade float especifica para onde elemento vai flutuar

```
img {  
    float: right;  
    margin: 0 0 10px 10px;  
}
```

Propriedade Display

- Especifica como o elemento é renderizado
- Valores
 - none - Esconde elemento e não deixa espaço -> interessante para fazer elementos aparecerem e desaparecem usando JS
 - block - elemento quebra linha e ocupa largura completa
 - inline - elemento não quebra linha e só ocupa o espaço necessário
 - inline-block - funcionam como elemento inline tendo *width* e *height* - usado para fazer caixas que se reorganizam quando o navegador é redimensionado

```
.floating-box {  
    display: inline-block;  
    width: 150px;  
    height: 75px;  
    margin: 10px;  
    border: 3px solid #73AD21;  
}
```

display:none vs visibility:hidden

- Ambos fazem o elemento desaparecer
- **display:none**
 - elemento não ocupa mais lugar no layout
- **visibility:hidden**
 - elemento ainda ocupa lugar no layout

Pseudo-classes

- Pseudo-classes definem um estado especial do elemento
 - Exemplo: links mostrados em slides anteriores

```
selector:pseudo-class {  
    property:value;  
}
```

CSS

```
p {  
    display: none;  
    background-color: yellow;  
    padding: 20px;  
}
```

```
div:hover p {  
    display: block;  
}
```

HTML

```
<div>Hover over me to show the p element  
  <p>Tada! Here I am!</p>  
</div>
```

Pseudo-elementos

- Estiliza partes do elemento
 - Primeira letra ou linha de uma elemento
 - Inserir conteúdo antes ou depois do conteúdo de um elemento

- Sintaxe:

```
selector::pseudo-element {  
    property:value;  
}
```

- Pseudo-elementos

- ::first-line
- ::first-letter
- ::before

Opacidade

- Propriedade opacity

```
img {  
    opacity: 0.5;  
    filter: alpha(opacity=50);  
}
```

```
img:hover {  
    opacity: 1.0;  
    filter: alpha(opacity=100);  
}
```

- Usando rgba

```
div {  
    background: rgba(76, 175, 80, 0.3)  
}
```

Barra de Navegação

- Uma barra de navegação é essencialmente uma lista de links

```
<ul>
  <li><a href="default.asp">Home</a></li>
  <li><a href="news.asp">News</a></li>
  <li><a href="contact.asp">Contact</a></li>
  <li><a href="about.asp">About</a></li>
</ul>
```

- Remove-se os bullets

```
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
}
```

Barra de Navegação Vertical

- Transformar elementos <a> em bloco para se tornar todo clicável dar width

```
li a {  
  display: block;  
  width: 60px;  
}
```

- Após isso, é possível estilizar como quiser

Barra de Navegação Vertical - Exemplo

```
body {  
    margin: 0;  
}  
  
ul {  
    list-style-type: none;  
    margin: 0;  
    padding: 0;  
    width: 25%;  
    background-color: #f1f1f1;  
    position: fixed;  
    height: 100%;  
    overflow: auto;  
}  
  
li a {  
    display: block;  
    color: #000;  
    padding: 8px 16px;  
    text-decoration: none;  
}  
  
li a.active {  
    background-color: #4CAF50;  
    color: white;  
}  
  
li a:hover:not(.active) {  
    background-color: #555;  
    color: white;  
}  
  
.container {  
    margin-left: 25%;  
    padding: 1px 16px;  
    height: 1000px;  
}
```

Barra de Navegação Horizontal - Exemplo

```
body {margin:0;}
ul {
    list-style-type: none;
    margin: 0;
    padding: 0;
    overflow: hidden;
    background-color: #333;
    position: fixed;
    top: 0;
    width: 100%;
}

li {
    float: left;
}

li:last-child {
    float:right;
}

li a {
    display: block;
    color: white;
    text-align: center;
    padding: 14px 16px;
    text-decoration: none;
}

li a:hover:not(.active) {
    background-color: #111;
}

.active {
    background-color: #4CAF50;
}

.container{
    padding:20px;
    margin-top:30px;
    background-color:#1abc9c;
    height:1500px;
}
```

Dropdown

```
.dropdown {  
    position: relative;  
    display: inline-block;  
}
```

```
.dropdown-content {  
    display: none;  
    position: absolute;  
    background-color: #f9f9f9;  
    min-width: 160px;  
    padding: 12px 16px;  
}
```

```
.dropdown:hover .dropdown-content {  
    display: block;  
}
```

```
<div class="dropdown">  
  <span>Mouse over me</span>  
  <div class="dropdown-content">  
    <p>Hello World!</p>  
  </div>  
</div>
```


Perguntas?

Prof. Victor Farias