code4sp
coding for social promotion

Co-funded by the
Erasmus+ Programme
of the European Union

1.:

Pacote de Material de Formação Code4SP

WP3:

Materiais de Formação Code4SP

Preparado por:

social
hackers
academy

Informação do Projeto

Acrónimo do projeto: Code4SP

Título do projeto: Coding for Social Promotion

Referência do projeto: 621417-EPP-1-2020-1-PT-EPPKA3-IPI-SOC-IN

Website do projeto: www.code4sp.eu

Parceiro autor: Social Hacker Academy

Versão do documento: 1

Data de preparação: 25/02/2022

Histórico do Documento			
Data	Versão	Autor	Descrição
25/02/2022	1	SHA	Esboço

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Tabela de Conteúdos

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Informação do Tópico

Tópico:

3. CSS

Pré-requisitos:

Literacia informática básica, software básico instalado, conhecimentos básicos de gestão de ficheiros, e noções básicas de HTML (Introdução ao HTML).

Carga horária:

10 horas.

Descrição:

O CSS é uma linguagem de folhas de estilo utilizada para descrever a apresentação de um documento escrito em HTML ou XML (incluindo dialetos XML, tais como o SVG, MathML ou XHTML). O CSS descreve como os elementos devem ser apresentados no ecrã, em papel, ou noutros meios.

Resultados de aprendizagem:

Os alunos obterão conhecimentos sobre como definir CSS, utilizar sintaxe CSS básica, configurar páginas web com CSS, utilizar CSS para estilizar texto, fonte, e propriedades, utilizar CSS para estilizar fundos de páginas, listas de estilo em CSS, utilizar classes CSS e IDs, utilizar bordas e propriedades CSS de altura e largura, utilizar pseudo elementos CSS, posicionar elementos com CSS e validar CSS e HTML.

Material necessário:

- Computador ou portátil
- Ligação à internet

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

- Editor de texto (online ou offline): Sublime Text/Brackets/W3Schools online editor

Cenário de Aula:

O tempo total para este tópico é de 10 horas, e caberá ao *coach*/formador decidir quanto tempo dedicar ao ensino de cada subtópico. Para aproveitar ao máximo todo o tempo disponível, propomos a utilização dos materiais de formação produzidos pelo projeto (apresentações PPT), que foram concebidos tendo em mente uma utilização eficaz do tempo. Estas apresentações são compostas pelos seguintes elementos:

- Desenvolvimento do subtópico e principais ideias a reter;
- Atividades/Exercícios propostos.

Dito isto, se o *coach*/formador seguir a sequência lógica dos PPTs, poderá certamente completar a sessão dentro do tempo limite estipulado. Estas apresentações podem também ser disponibilizadas aos formandos para estudo individual.

Subtópicos:

- 3.1. CSS - Introdução
- 3.2. CSS - Modelos de Design Reativo
- 3.3. CSS GRID
- 3.4. CSS - Avançado
- 3.5. CSS SASS

Recursos adicionais:

- Tutorial de CSS: w3schools
- Curso online de HTML e CSS: CodeAcademy

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

3.1. CSS - Introdução

O que é o CSS?

Cascading Style Sheets (Folhas de Estilo em Cascata), habitualmente designado por CSS, é uma linguagem de design simples, destinada a simplificar o processo de apresentar uma página web.

O CSS trata do aspeto e da vibração que uma página web liberta. Ao utilizar o CSS é possível controlar a cor do texto, o estilo das fontes, o espaçamento entre parágrafos, como as colunas são dimensionadas e dispostas, que imagens ou cores de fundo são usadas, bem como uma variedade de outros efeitos.

CSS é fácil de aprender e compreender, mas proporciona um controlo poderoso sobre a apresentação de um documento HTML. Mais comummente, o CSS é combinado com as linguagens de marcação HTML ou XHTML

Vantagens do CSS

- **O CSS poupa tempo** - Poderás digitar CSS uma vez e depois reutilizar a mesma folha em múltiplas páginas HTML. Podes definir um estilo para cada elemento HTML e aplicá-lo a quantas páginas web entenderes.
- **As páginas carregam mais rápido** - Se estiveres a usar o CSS, não precisarás de escrever atributos de *tag* HTML todas as vezes. Basta escrever uma regra CSS de uma *tag* e aplicá-la a todas as ocorrências dessa *tag*. Portanto, menos código significa tempos de *download* mais rápidos.
- **Manutenção fácil** - Para fazer uma alteração global, basta alterar o estilo e todos os elementos em todas as páginas da web serão atualizados automaticamente.
- **Estilos superiores ao HTML** - O CSS tem uma gama muito mais ampla de atributos em relação ao HTML, por isso pode dar uma aparência muito melhor à tua página HTML, em comparação com os atributos HTML.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

- **Compatibilidade de múltiplos dispositivos** - As folhas de estilo permitem a optimização do conteúdo para mais de um tipo de dispositivo. Utilizando o mesmo documento HTML, diferentes versões de um website podem ser apresentadas para dispositivos portáteis, tais como PDAs e telemóveis ou para impressão.
- **Padrões globais da web** – Agora os atributos HTML estão a tornar-se obsoletos, recomendando-se o uso de CSS. Portanto, é uma boa ideia começar a usar o CSS em todas as páginas HTML para torná-las compatíveis com futuros navegadores.

Quem cria e mantém o CSS?

O CSS é criado e mantido através de um grupo de pessoas dentro do W3C chamado Grupo de Trabalho do CSS. O Grupo de Trabalho do CSS cria documentos, designados por *especificações*. Quando uma especificação é discutida e ratificada oficialmente pelos membros do W3C, torna-se uma recomendação.
Estas especificações ratificadas são chamadas recomendações porque o W3C não tem qualquer controlo sobre a implementação efetiva da língua. Empresas e organizações independentes criam esse *software*.

NOTA: O World Wide Web Consortium ou W3C é um grupo que faz recomendações sobre como funciona a Internet e como esta deve evoluir.

Versões do CSS

O CSS foi originalmente lançado em 1996 e consiste em propriedades para adicionar propriedades da fonte, tais como tipo de letra e cor de ênfase do texto, fundos, e outros elementos. O CSS2 foi lançado em 1998 com estilos adicionados para outros tipos de suportes, de modo a poder ser utilizado para a conceção do *layout* da página. O CSS3 foi lançado em 1999 e nele foram adicionadas propriedades de estilo de apresentação que lhe permitem construir uma apresentação a partir de documentos.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Sintaxe do CSS

Um CSS inclui regras de estilo que são interpretadas pelo navegador e depois aplicadas aos elementos correspondentes no seu documento. Uma regra de estilo é feita de três partes:

- **Selecionador**: Um selecionador é uma etiqueta HTML na qual será aplicado um estilo. Esta poderá ser qualquer tag, como por exemplo <h1>, <table>, etc.
- **Propriedade:** Uma propriedade é um tipo de atributo da etiqueta HTML. Em termos simples, todos os atributos HTML são convertidos em propriedades CSS, que podem ser cor, contorno, etc.
- **Valor**: Os valores são atribuídos às propriedades. Por exemplo, a propriedade de cor pode ter o valor ou vermelho ou #F1F1F1F1 etc.

Selecionadores do CSS

Os selecionadores CSS são usados para *encontrar* (ou *selecionar*) os elementos HTML nos quais desejas aplicar um estilo (ou *estilizar*). Podemos dividir os selecionadores de CSS em cinco categorias:

- **Selecionadores simples** – selecionam elementos com base no nome, id, classe;
- **Selecionadores combinados** – selecionam elementos com base num relacionamento específico entre eles;
- **Selecionadores de pseudo-classe** – selecionam elementos com base num determinado estado;
- **Selecionadores de pseudo-elementos** – selecionam e estilizam uma parte de um elemento;
- **Selecionadores de atributo** – selecionam elementos com base num atributo ou valor de atributo.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

O selecionador de elementos CSS

O selecionador de elementos seleciona elementos HTML com base no nome do elemento.

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

***Figura 1** – O selecionador de elementos CSS (**Fonte**: https://www.w3schools.com/css/css_selectors.asp)*

O selecionador de ID do CSS

O selector de id utiliza o atributo id de um elemento HTML para seleccionar um elemento específico. O id de um elemento é único dentro de uma página, por isso o selector de id é usado para selecionar um elemento único. Para selecionar um elemento com um id específico, digita-se um caractere *hash* (#), seguido do *ID* do elemento.

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

***Figura 2** – O Selector de ID CSS (**Fonte:** https://www.w3schools.com/css/css_selectors.asp)*

O selecionador de classe do CSS

O selector de classes seleciona elementos HTML com um atributo de classe específico. Para selecionar elementos com uma classe específica, digita-se um caractere ponto ('.'), seguido do nome da classe.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

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

Figura 3 *– O selecionador de classe do CSS (**Fonte:** https://www.w3schools.com/css/css_selectors.asp)*

O selecionador universal do CSS

O selecionador universal (*) seleciona todos os elementos HTML da página.

```
* {
  text-align: center;
  color: blue;
}
```

Figura 5 *– O selecionador universal do CSS (**Fonte:** https://www.w3schools.com/css/css_selectors.asp)*

Selecionadores de agrupamento

Se desejável, poderá aplicar-se um estilo a vários selecionadores. Basta separar os selecionadores com uma vírgula, como indicado no exemplo seguinte:

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

Figura 6 *– Selecionadores de agrupamento (**Fonte:** https://www.w3schools.com/css/css_selectors.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Comentários do CSS

- Os comentários são usados para explicar o código e podem ajudar quando se edita o código-fonte *a posteriori*;
- Os comentários são ignorados pelos navegadores;
- Um comentário do CSS começa com /* e termina com */.

```
/* This is a single-line comment */
p {
  color: red;
}
```

Figura 7 *– Comentários do CSS (**Fonte:** https://www.w3schools.com/css/css_comments.asp)*

Cores do CSS

As cores no CSS podem ser especificadas através dos seguintes métodos:

- Cores hexadecimais;
- Cores hexadecimais com transparência;
- Cores RGB;
- Cores RGBA;
- Cores HSL;
- Cores HSLA;
- Nomes de cores pré-definidos/cruzados;
- Com a palavra-chave de cor atual.

Cores hexadecimais

É especificada uma cor hexadecimal com: #RRGGBB, onde os números inteiros hexadecimais RR (vermelho), GG (verde) e BB (azul) especificam os componentes da cor. Todos os valores devem estar compreendidos entre 00 e FF.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Por exemplo, o valor #0000ff é apresentado como azul, porque o componente azul é definido para o seu valor mais alto (ff) e os outros são definidos para 00.

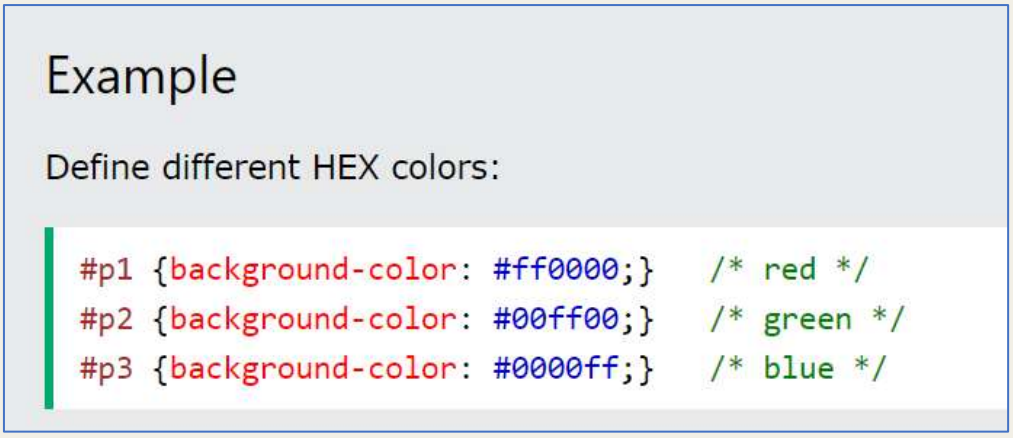


Figura 8 *– Cores hexadecimais (**Fonte:** https://www.w3schools.com/css/css_colors.asp)*

Cores hexadecimais com transparência

É especificada uma cor hexadecimal com: #RRGGBB. Para acrescentar transparência, acrescentar dois dígitos adicionais entre 00 e FF.

Figura 9 *– Cores hexadecimais com transparência (**Fonte:** https://www.w3schools.com/css/css_colors.asp)*

Cores RGB

Um valor de cor RGB é especificado com a função rgb()#, que tem a seguinte sintaxe:

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```
rgb(vermelho, verde, azul)
```

Cada parâmetro (vermelho, verde e azul) define a intensidade da cor e pode ser um número inteiro entre 0 e 255 ou um valor percentual (de 0% a 100%).

Por exemplo, o valor rgb (0,0,255) é apresentado como azul, porque o parâmetro azul é fixado no seu valor mais alto (255) e os outros são fixados em 0.

Além disso, os seguintes valores definem cores iguais: rgb (0,0,255) e rgb (0%,0%,100%).

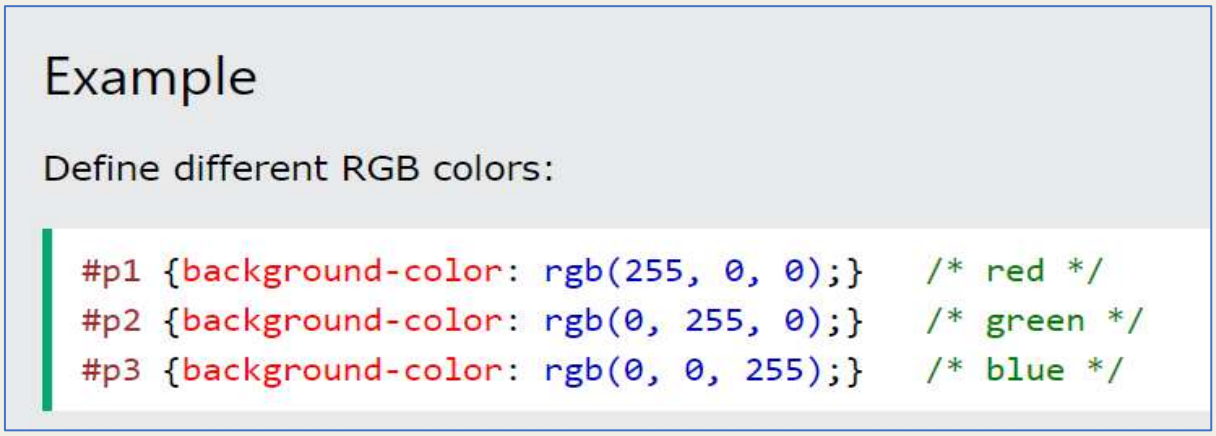


Figura 10 *– Cores RGB (**Fonte:** https://www.w3schools.com/css/css_colors.asp)*

Os valores RGBA das cores

Os valores RGBA das cores são uma extensão dos valores de cor RGB com um canal *alfa* - que especifica a opacidade do objeto.

Os RGBA de uma cor são especificados com a função rgba()#, que tem a seguinte sintaxe:

```
rgba (vermelho, verde, azul, alfa)
```

O parâmetro alfa é um número entre 0,0 (totalmente transparente) e 1,0 (totalmente opaco).

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

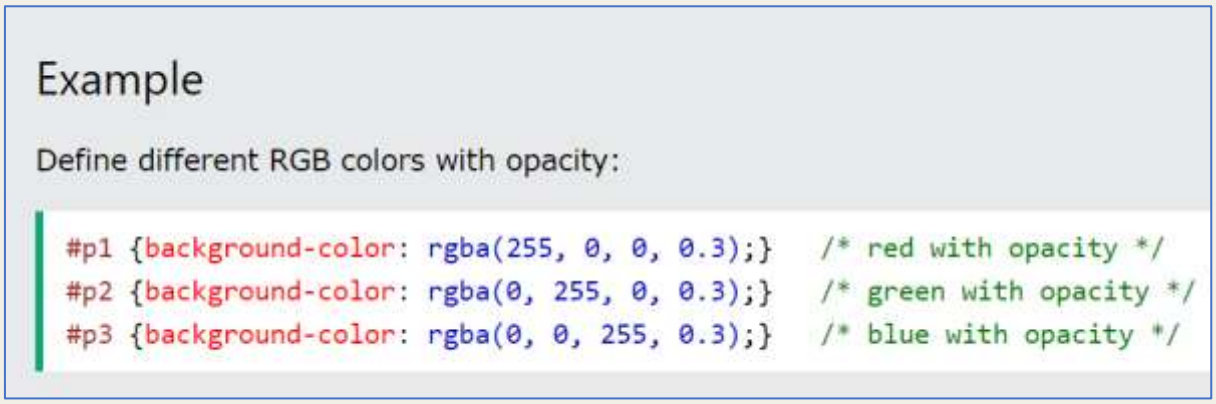


Figura 11 *– RGBA das cores com opacidade (**Fonte:** https://www.w3schools.com/css/css_colors.asp)*

Valores HSL das cores

A sigla HSL (do inglês *hue, saturation and lightness*) remete-nos para tonalidade, saturação e leveza - e transmite uma representação cilíndrico-coordenada de cores.

Um valor de cor HSL é especificado com a função #hsl()##, que possui a seguinte sintaxe:

```
hsl(hue, saturation, lightness)
```

A tonalidade é um grau na roda de cor (de 0 a 360) - 0 (ou 360) é vermelho, 120 é verde, 240 é azul. A saturação é um valor percentual; 0% significa uma tonalidade de cinzento e 100% é a cor completa. A leveza é também uma percentagem; 0% é o preto, 100% é o branco.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Example

Define different HSL colors:

```
#p1 {background-color: hsl(120, 100%, 50%);}   /* green */
#p2 {background-color: hsl(120, 100%, 75%);}   /* light green */
#p3 {background-color: hsl(120, 100%, 25%);}   /* dark green */
#p4 {background-color: hsl(120, 60%, 70%);}    /* pastel green */
```

***Figura 12** – HSL das cores (**Fonte:** https://www.w3schools.com/css/css_colors.asp)*

HSLA das cores

Os valores de cor HSLA são uma extensão dos valores de cor HSL com um canal alfa - que especifica a opacidade do objeto.

Um valor de cor HSLA é especificado com a #função hsla()##, #que tem a seguinte sintaxe:

```
hsla(hue, saturation, lightness, alpha)
```

O parâmetro alfa é um número entre 0,0 (totalmente transparente) e 1,0 (totalmente opaco).

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Example

Define different HSL colors with opacity:

```
#p1 {background-color: hsla(120, 100%, 50%, 0.3);}   /* green with opacity */
#p2 {background-color: hsla(120, 100%, 75%, 0.3);}   /* light green with opacity */
#p3 {background-color: hsla(120, 100%, 25%, 0.3);}   /* dark green with opacity */
#p4 {background-color: hsla(120, 60%, 70%, 0.3);}    /* pastel green with opacity */
```

Figura 13 *– HSL das cores com opacidade (**Fonte:** https://www.w3schools.com/css/css_colors.asp)*

Nomes de cores pré-definidos/cruzados

140 nomes de cores são predefinidos na especificação de cores HTML e CSS.

Por exemplo: azul, vermelho, coral, castanho, etc.

Example

Define different color names:

```
#p1 {background-color: blue;}
#p2 {background-color: red;}
#p3 {background-color: coral;}
#p4 {background-color: brown;}
```

Figura 14 *– Nomes de cores cruzados (**Fonte:** https://www.w3schools.com/css/css_colors.asp)*

A palavra-chave currentcolor

A palavra-chave *currentcolor* refere-se ao valor da propriedade de cor de um elemento.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Example

The border color of the following element will be blue, because the text color of the element is blue:

```
#myDIV {
  color: blue; /* Blue text color */
  border: 10px solid currentcolor; /* Blue border color */
}
```

Figura 15 *– A palavra-chave currentcolor (**Fonte:** https://www.w3schools.com/colors/colors_currentcolor.asp)*

Fundos do CSS

As propriedades de plano de fundo do CSS são usadas para adicionar efeitos de plano de fundo para elementos. Algumas propriedades de fundo são:

- background-color
- background-image
- background-repeat
- background-attachment
- background-position
- background (shorthand property)

Figura 16 *– Fundo do CSS (**Fonte:** https://www.w3schools.com/css/css_background.asp)*

A propriedade *background-color*

A propriedade background-color especifica a cor de fundo de um elemento.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

***Figura 17** – Definição da cor de um fundo (**Fonte:**https://www.w3schools.com/css/css_background.asp)*

A propriedade *background-image*

A propriedade *background-image* especifica uma imagem a ser usada como plano de fundo de um elemento.

Por defeito, a imagem é repetida de modo a cobrir todo o elemento.

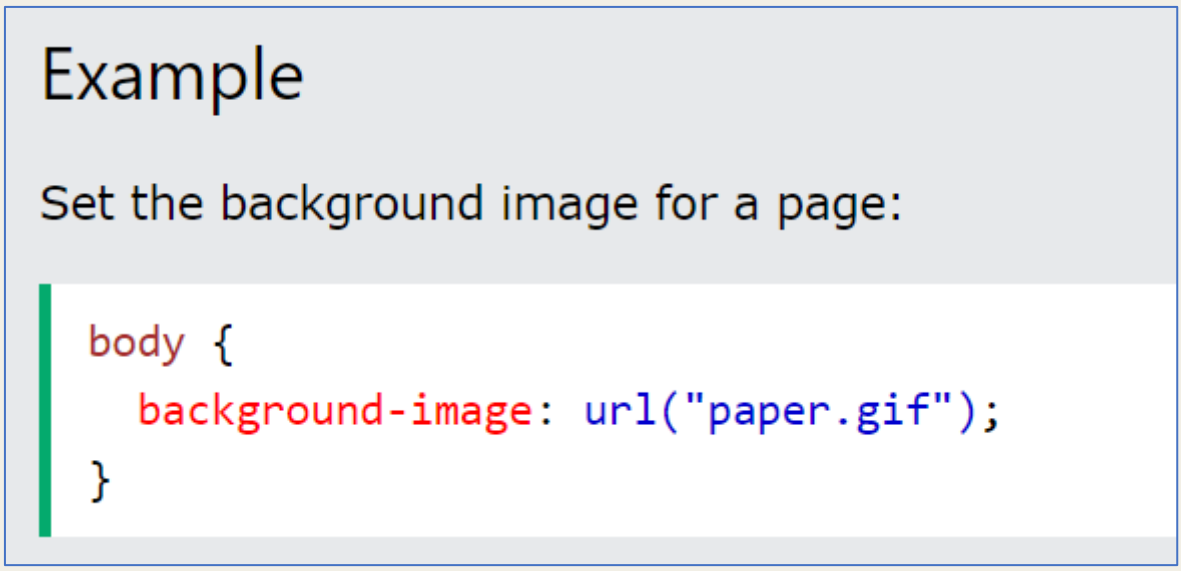


***Figura 18** – A propriedade background-image (**Fonte:** https://www.w3schools.com/css/css_background.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

A propriedade *background-repeat*

Por defeito, a propriedade *background-repeat* repete uma imagem na horizontal e na vertical. Algumas imagens devem ser repetidas apenas horizontalmente ou verticalmente, caso contrário, terão um aspeto estranho.

A propriedade *background-attachment*

A propriedade *background-attachment* especifica se a imagem de fundo deve rolar ao longo da página (efeito "*scroll*") ou ser fixa.

Example

Specify that the background image should be fixed:

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

Figura 19 *– A propriedade background-attachment (**Fonte:** https://www.w3schools.com/css/css_background_attachment.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Example

Specify that the background image should scroll with the rest of the page:

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

***Figura 20** – A propriedade background-attachment, ex. 2(**Fonte:** https://www.w3schools.com/css/css_background_attachment.asp)*

A propriedade *shorthand*

Para encurtar o código, é também possível especificar todas as propriedades de fundo numa única propriedade. A isto chama-se uma propriedade *shorthand* ("*abreviada*").

Em vez de digitar o seguinte:

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

***Figura 21** – A propriedade shorthand*
*(**Fonte:**https://www.w3schools.com/css/css_background_shorthand.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Pode ser usada a propriedade *shorthand*:

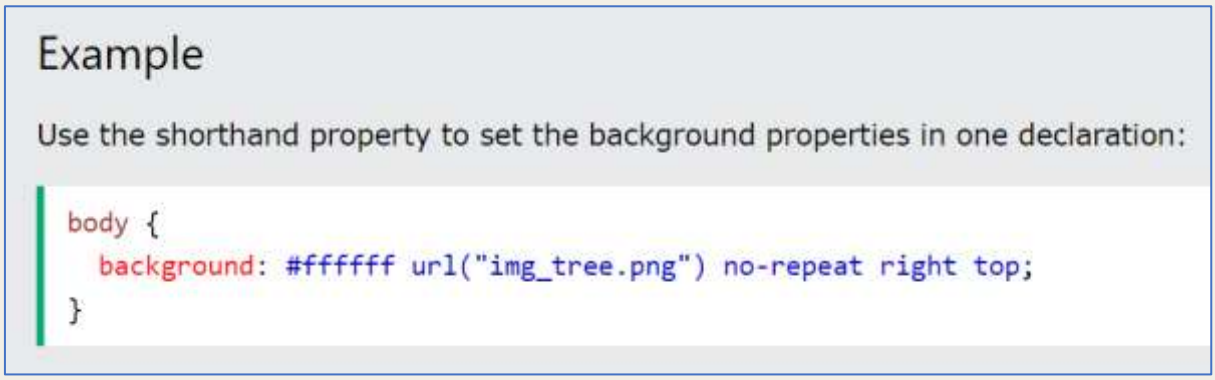


***Figura 22** – A propriedade shorthand*
*(**Fonte:**https://www.w3schools.com/css/css_background_shorthand.asp)*

Limites de margem no CSS

As propriedades de margens no CSS permitem especificar o estilo, largura e cor dos limites de margem de um elemento.

***Figura 23** – Limites de margem no CSS (**Fonte:** https://www.w3schools.com/css/css_border.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

As margens no CSS

As margens são utilizadas para criar espaço em torno de elementos, fora de quaisquer fronteiras definidas.

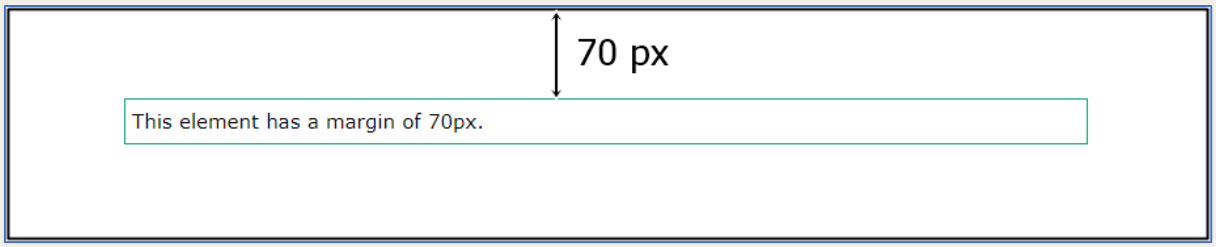


Figura 24 *– Margens no CSS (**Fonte:**https://www.w3schools.com/css/css_margin.asp)*

Com o CSS, há total controlo sobre as margens. Existem propriedades para definir a margem para cada lado de um elemento (superior, direita, inferior, e esquerda).

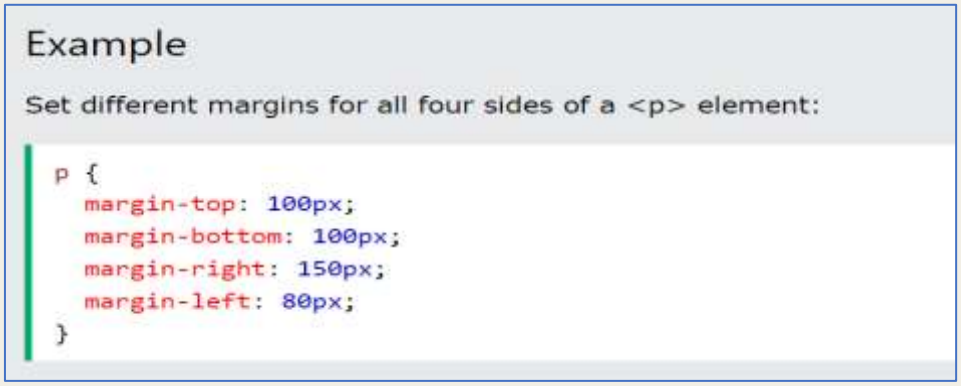


Figura 25 *– Margens no CSS (**Fonte:**https://www.w3schools.com/css/css_margin.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Preenchimento no CSS

O preenchimento é usado para criar espaço ao redor do conteúdo de um elemento, dentro de qualquer limite de margem definido.

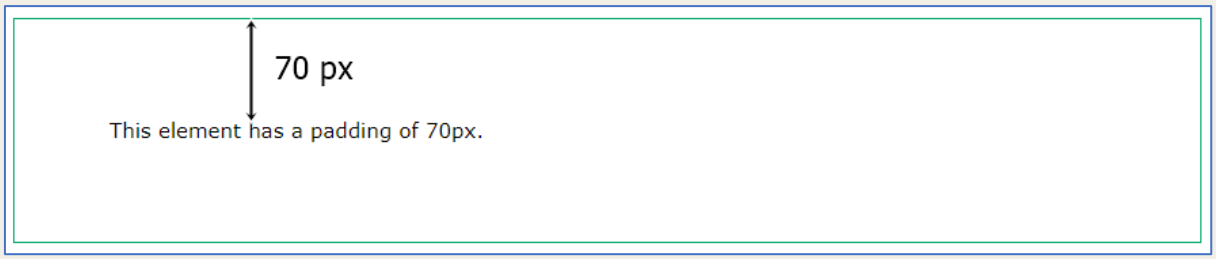


Figura 26 *– Preenchimento no CSS (**Fonte:**https://www.w3schools.com/css/css_padding.asp)*

Com o CSS, há total controlo sobre as margens. Existem propriedades para definir a margem para cada lado de um elemento (superior, direita, inferior, e esquerda).

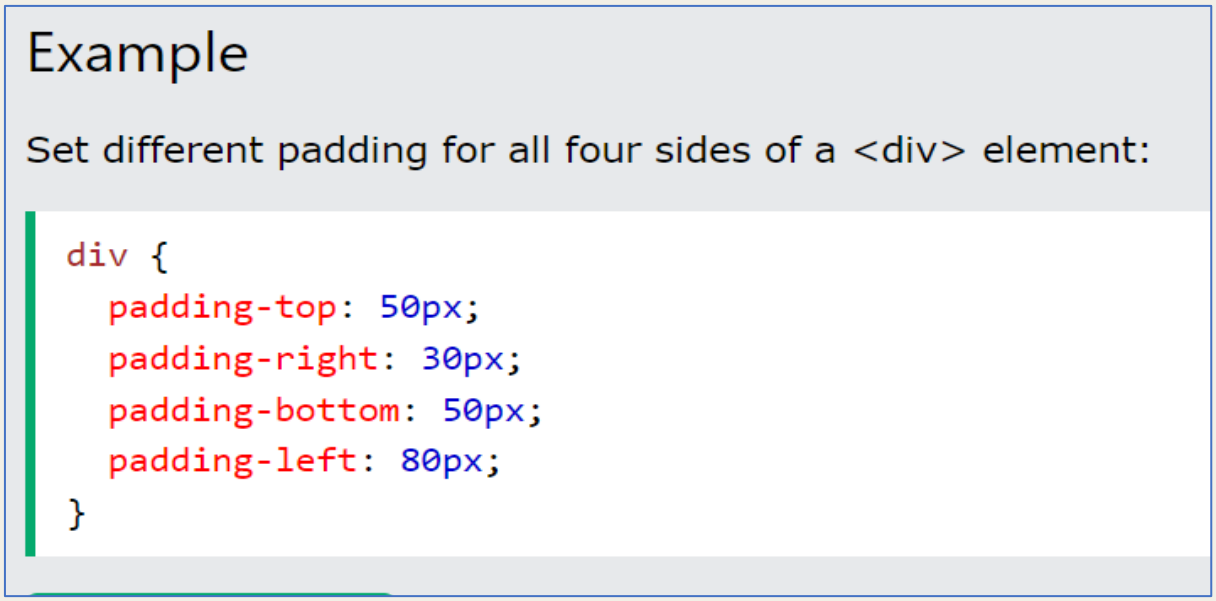


Figura 27 *– Preenchimento no CSS (**Fonte:**https://www.w3schools.com/css/css_padding.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Altura e largura do CSS

As propriedades de altura e largura são utilizadas para definir a altura e largura de um elemento.
As propriedades de altura e largura não incluem preenchimento, limites de margem ou margens. Ele define a altura/largura da área dentro do preenchimento, limites de margem e margem do elemento.

As propriedades de altura e largura podem ter os seguintes valores:

- *auto* – o padrão. O navegador calcula a altura e a largura;
- comprimento – define a altura/largura em px, cm, etc.;
- % - define a altura/largura em percentagem do bloco que contém;
- inicial – define a altura/largura para o seu valor por defeito;
- *inherit* - altura/largura será *herdada* do seu valor-pai.

Example

Set the height and width of a element:

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

Figura 28 *– Altura e largura (**Fonte:** https://www.w3schools.com/css/css_dimension.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

O modelo de caixa CSS

Em CSS, o termo "modelo de caixa" é usado quando se fala de design e *layout*.
O modelo de caixa CSS é essencialmente uma caixa que envolve cada elemento HTML. Consiste em: margens, bordas, acolchoamento, e o conteúdo real. A imagem abaixo ilustra o modelo de caixa:

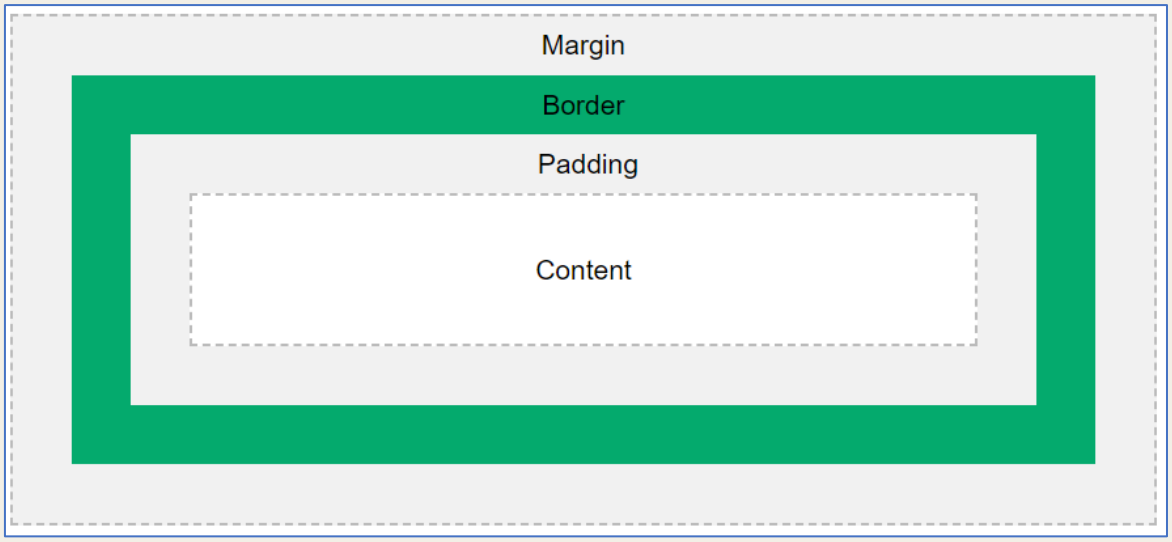


***Figura 29** – Modelo de caixa do CSS (**Fonte:** https://www.w3schools.com/css/css_boxmodel.asp)*

Explicação das diferentes partes:

- Conteúdo – o conteúdo da caixa, onde o texto e as imagens aparecem;
- Preenchimento – limpa uma área ao redor do conteúdo. O forro é transparente;
- Limite da margem/borda – borda que contorna o preenchimento e o conteúdo;
- Margem - área transparente, fora da fronteira.
  O modelo de caixa permite-nos adicionar uma fronteira em torno de elementos, e definir o espaço entre elementos.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Contorno do CSS

Um contorno é uma linha que é traçada em torno de elementos, fora das fronteiras para fazer o elemento "*sobressair*".

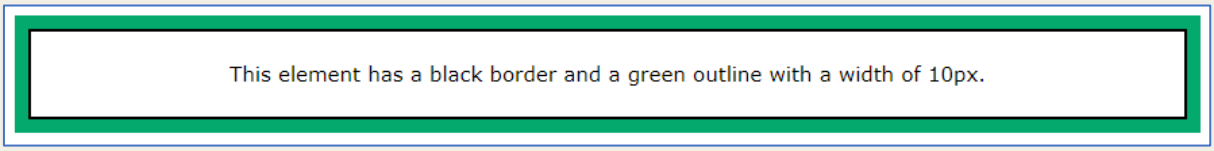


Figura 30 *– Contorno do CSS (**Fonte:** https://www.w3schools.com/css/css_outline.asp)*

Texto em CSS

O CSS tem muitas propriedades de formatação de texto.

Figura 31 *– Texto em CSS (**Fonte:** https://www.w3schools.com/css/css_text.asp)*

- Cor do texto
- Alinhamento do texto
- Decoração do texto
- Transformação do texto
- Espaçamento do texto
- Sombra do texto

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Tipos de letra no CSS

- O tipo de letra certo pode ajudar a criar uma identidade forte para um website.
- Usar uma fonte que seja fácil de ler é importante. A fonte agrega valor ao texto. Também é importante escolher a cor e o tamanho do texto corretos para a fonte.

Generic Font Family	Examples of Font Names
Serif	Times New Roman Georgia Garamond
Sans-serif	Arial Verdana Helvetica
Monospace	Courier New Lucida Console Monaco
Cursive	Brush Script MT Lucida Handwriting
Fantasy	Copperplate Papyrus

***Figura 32** – CSS Fonts (**Fonte:** https://www.w3schools.com/css/css_font.asp)*

Ícones do CSS

- A forma mais simples de adicionar um ícone à página HTML, é com uma biblioteca de ícones, tal como o tipo de letra “Awesome”.
- Adicionar o nome da classe do ícone especificado a qualquer elemento HTML em linha (como <i> ou).
- Todos os ícones nas bibliotecas de ícones abaixo são vetores escaláveis que podem ser personalizados com CSS.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

***Figura 33** – Ícones do CSS (**Fonte:** https://www.w3schools.com/css/css_icons.asp)*

Hiperligações no CSS

As hiperligações (ou "links") podem ser estilizadas com qualquer propriedade CSS (por exemplo, cor, font-family, fundo, etc,).

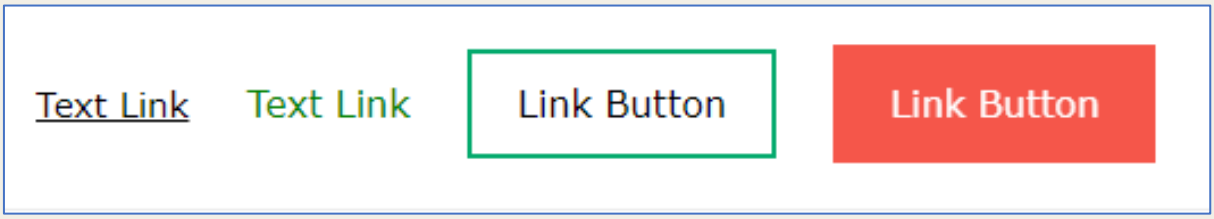


***Figura 34** – Links ou hiperligações no CSS (**Fonte:** https://www.w3schools.com/css/css_link.asp)*

Além disso, as hiperligações podem ser estilizadas de forma diferente dependendo do estado em que se encontram.

Os quatro estados dos *links*/hiperligações são:

- a:link - um link normal, não visitado;
- a:visited - um link que o utilizador visitou;
- a:hover - um link que surge quando o utilizador passa o cursor por cima dele;
- a:active - um link no momento em que é clicado.

Listas do CSS

As propriedades delista CSS permitem:

- Definir marcadores de itens de lista diferentes para listas ordenadas;
- Definir marcadores de itens de lista diferentes para listas não ordenadas;
- Definir uma imagem como marcador de item da lista;

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

- Adicionar cores de fundo a listas e itens de lista.

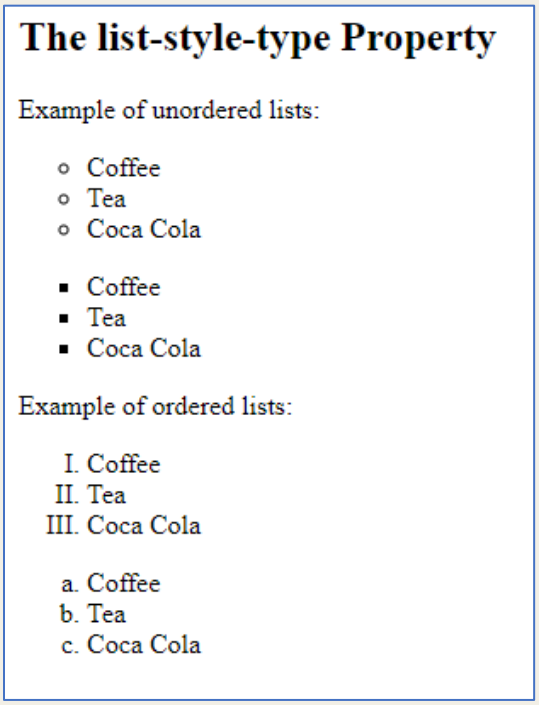


***Figura 35** – Listas do CSS (**Fonte:** https://www.w3schools.com/css/css_list.asp)*

Propriedades de disposição no CSS

- A propriedade de exibição especifica se/como é exibido um elemento.
- Cada elemento HTML tem um valor de exibição padrão dependendo do tipo de elemento que é. O valor de exibição padrão para a maioria dos elementos é *block* ou *inline*.

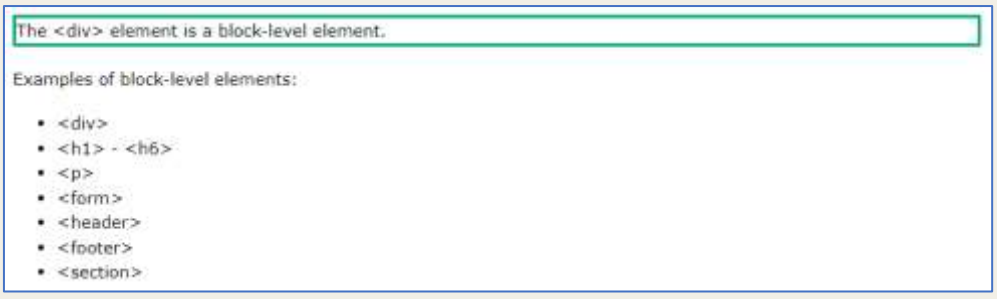


***Figura 36** – Disposição no CSS (**Fonte:** https://www.w3schools.com/css/css_display_visibility.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Tabelas no CSS

O aspeto de uma tabela HTML pode ser fortemente melhorado com o CSS:

Company	Contact	Country
Alfreds Futterkiste	Maria Anders	Germany
Berglunds snabbköp	Christina Berglund	Sweden
Centro comercial Moctezuma	Francisco Chang	Mexico
Ernst Handel	Roland Mendel	Austria
Island Trading	Helen Bennett	UK
Königlich Essen	Philip Cramer	Germany
Laughing Bacchus Winecellars	Yoshi Tannamuri	Canada
Magazzini Alimentari Riuniti	Giovanni Rovelli	Italy

Figura 37 *– Tabelas no CSS (**Fonte:** https://www.w3schools.com/css/css_table.asp)*

A propriedade *max-width*

- O problema com a função , mencionado anteriormente, ocorre quando a janela do navegador é menor do que a largura do elemento. O navegador adiciona então uma barra de deslocamento horizontal à página. O navegador adiciona então uma barra de deslocamento horizontal à página.
- Usar *max-width* em vez disso, nessa situação, melhorará o manuseio de janelas pequenas pelo navegador. Isto é importante ao tornar um site utilizável em dispositivos de tamanho reduzido.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Figura 38 *– A propriedade maxwidth (**Fonte:** https://www.w3schools.com/css/css_max-width.asp)*

O posicionamento no CSS

A propriedade de posição especifica o tipo de método de posicionamento utilizado para um elemento.

Existem cinco valores de posição diferentes:

- estático
- relativo
- fixo
- absoluto
- "*sticky*"

A propriedade *z-index*

A propriedade CSS Z-index especifica a ordem de empilhamento de um elemento (que elemento deve ser colocado à frente, ou atrás dos outros). Um elemento pode ter uma ordem de empilhamento positiva ou negativa, como se pode verificar abaixo:

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

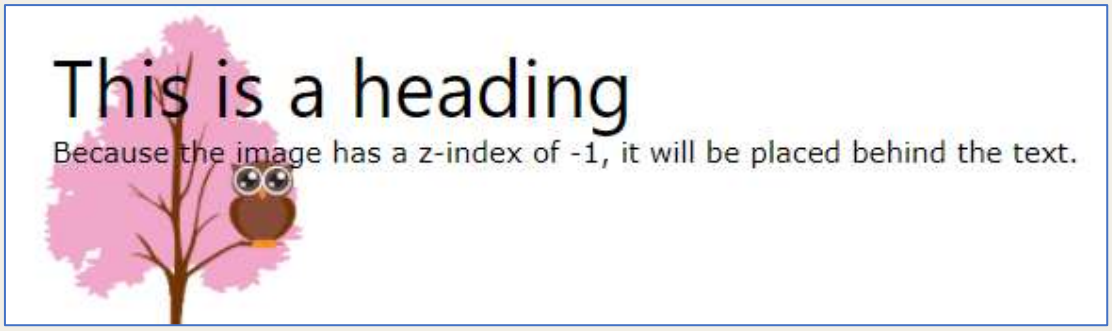


***Figura 39** – A propriedade Z-index (**Fonte:** https://www.w3schools.com/css/css_z-index.asp)*

Transbordamento (ou *overflow*) no CSS

A propriedade de transbordo ("overflow") especifica se se deve cortar o conteúdo ou adicionar barras de rolagem ("scrollbars") quando o conteúdo de um elemento é demasiado grande para caber na área determinada.

A propriedade de *overflow* tem os seguintes valores:

- visível

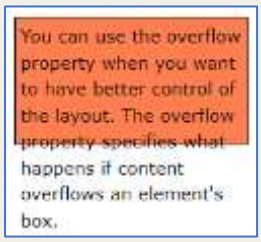


***Figura 40** – Valor visível na propriedade overflow (**Fonte:** https://www.w3schools.com/css/css_overflow.asp)*

- escondido

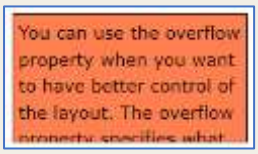


***Figura 41** – Valor escondido na propriedade overflow (**Fonte:** https://www.w3schools.com/css/css_overflow.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

- "scroll"

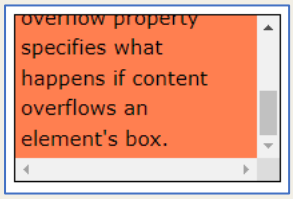


***Figura 42** – Valor "scroll" na propriedade overflow (**Fonte:** https://www.w3schools.com/css/css_overflow.asp)*

- auto

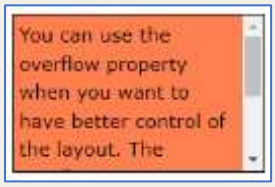


***Figura 43** – Valor "auto" na propriedade overflow (**Fonte:** https://www.w3schools.com/css/css_overflow.asp)*

A propriedade *float*

A propriedade float é utilizada para posicionamento e formatação do conteúdo.

Exemplo: **float: right**

O exemplo seguinte especifica que uma imagem deve flutuar para a **direita**.

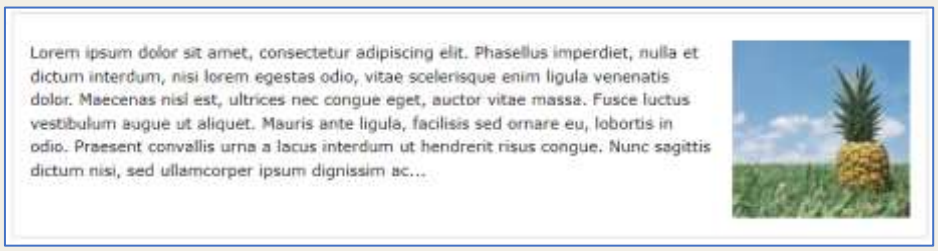


***Figura 44** – Flutuar para a direita no CSS (**Fonte:** https://www.w3schools.com/css/css_float.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

O exemplo seguinte especifica que uma imagem deve flutuar para a **esquerda** num texto:

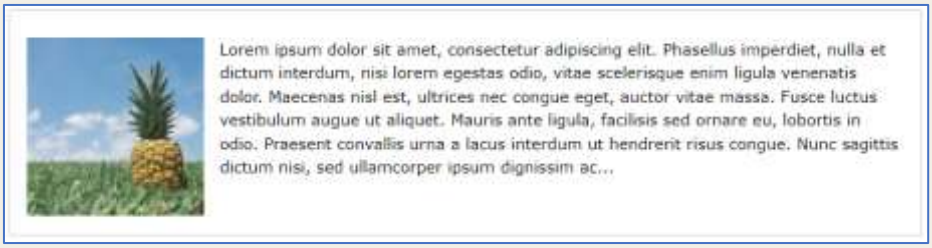


Figura 45 *– Uso da propriedade float-left (**Fonte:** https://www.w3schools.com/css/css_float.asp)*

Propriedade *inline-block*

- *Display: inline-block* permite definir uma largura e altura no elemento;
- Com *display: inline-block*, as margens/preenchimentos superiores e inferiores são respeitadas;
- *Display: inline-block* não adiciona uma quebra de linha após o elemento, então o elemento pode ficar ao lado de outros elementos.

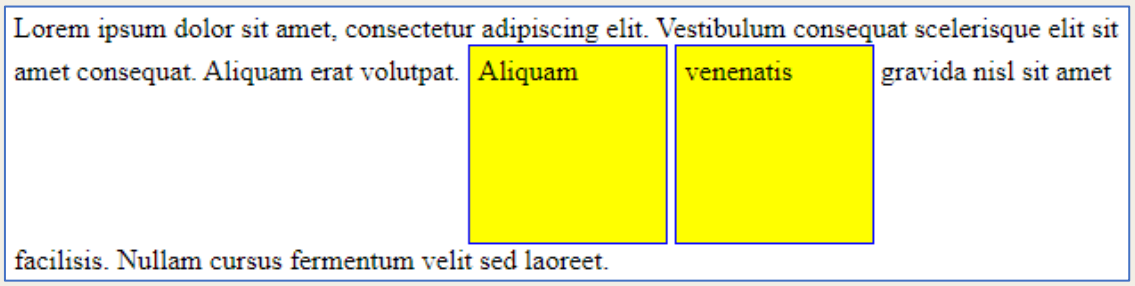


Figura 46 *– A propriedade Inline-block (**Fonte:** https://www.w3schools.com/css/css_inline-block.asp)*

A propriedade *align*

Para centralizar horizontalmente um elemento de bloco (como), usar *margin: auto.* A regulação da largura do elemento impedirá que este se estique até ao limite marginal do seu recipiente.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

O elemento ocupará então a largura especificada, e o espaço restante será dividido igualmente entre as duas margens:

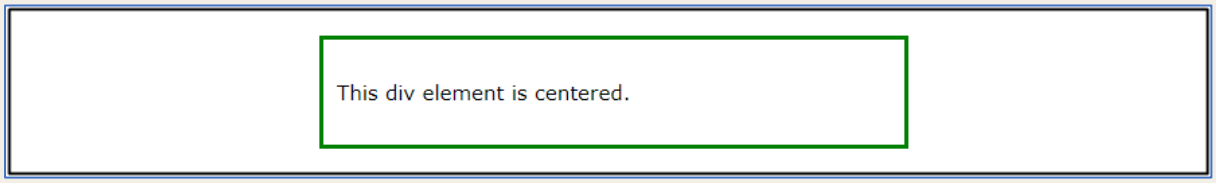


***Figura 47** – Alinhar elementos em CSS (**Fonte:** https://www.w3schools.com/css/css_align.asp)*

Combinadores do CSS

Um seletor CSS pode conter mais de um seletor simples. Entre os seletores simples, podemos incluir um combinador. Existem quatro combinadores diferentes no CSS:

- seletor descendente (espaço) -> ex. div p {background-color: yellow}
- seletor filho (>) -> ex. div > p {background-color: yellow}
- seletor de irmãos adjacente (+) -> ex. div + p {background-color: yellow}
- seletor geral de irmãos (~) -> ex. div ~ p {background-color: yellow}

Pseudo-classes de CSS

Uma pseudo-classe é utilizada para definir um estado especial de um elemento. Por exemplo, pode ser usado para:

- Estilizar um elemento quando um utilizador passa o cursor nele;
- Estilizar hiperligações visitadas e não visitadas de forma diferente;
- Estilizar um elemento quando ele estiver em foco.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```css
/* unvisited link */
a:link {
  color: #FF0000;
}

/* visited link */
a:visited {
  color: #00FF00;
}

/* mouse over link */
a:hover {
  color: #FF00FF;
}

/* selected link */
a:active {
  color: #0000FF;
}
```

Figura 48 *– Pseudo-Classes no CSS (**Fonte:** https://www.w3schools.com/css/css_pseudo_classes.asp)*

Pseudo-elementos no CSS

Um pseudo-elemento CSS é usado para estilizar partes especificadas de um elemento.

Por exemplo, pode ser usado para:

- Estilizar a primeira letra, ou linha, de um elemento;
- Inserir conteúdo antes, ou depois, do conteúdo de um elemento.

```css
p::first-line {
  color: #ff0000;
  font-variant: small-caps;
}
```

Figura 49 *– Pseudo element (**Fonte:** https://www.w3schools.com/css/css_pseudo_elements.asp)*

O pseudo-elemento de primeira linha é utilizado para acrescentar um estilo especial à primeira linha de um texto.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Opacidade no CSS

A propriedade de opacidade especifica a opacidade/transparência de um elemento. A propriedade de opacidade pode ter um valor de 0,0 - 1,0. Quanto mais baixo for o valor, mais transparente será o seu elemento.

Figura 50 *– Opacidade no CSS (**Fonte:** https://www.w3schools.com/css/css_image_transparency.asp)*

A barra de navegação do CSS

Com o CSS pode transformar menus HTML aborrecidos em barras de navegação atrativas.

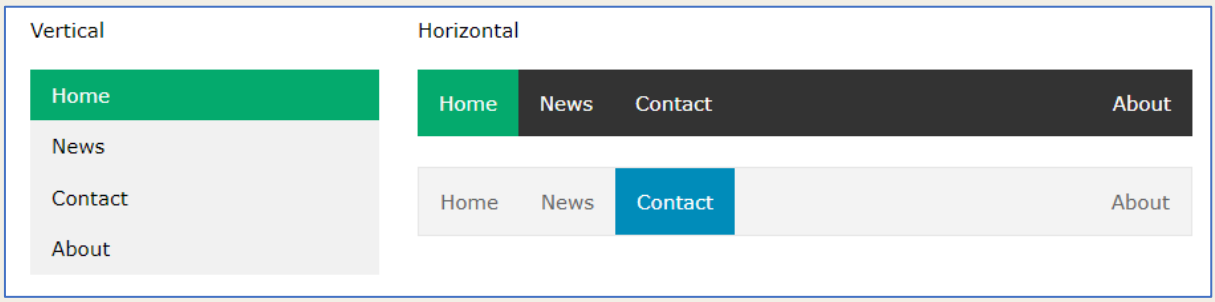


Figura 51 *– A barra de navegação do CSS (**Fonte:** https://www.w3schools.com/css/css_navbar.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Menus em cascata do CSS

Com o CSS, é possível transformar menus HTML aborrecidos em barras de navegação atrativas.

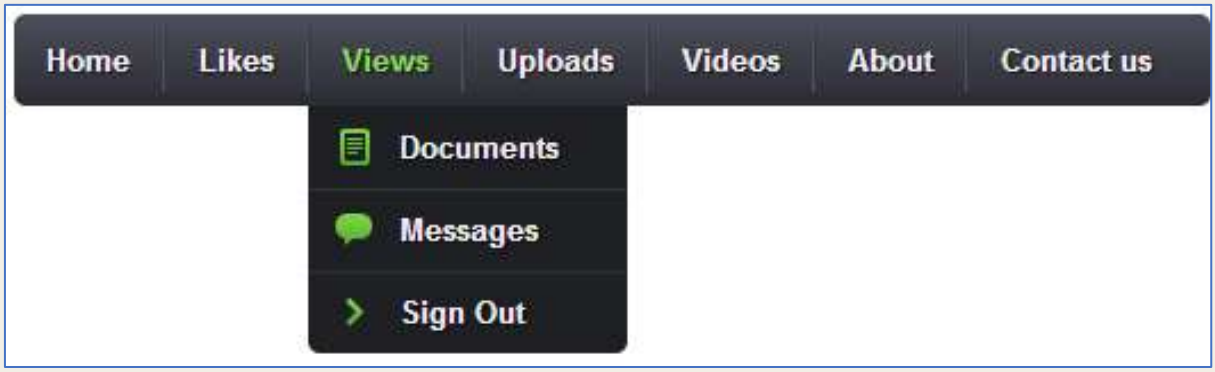


Figura 52 *– Menu em cascata (**Fonte:** https://www.w3schools.com/css/css_dropdowns.asp)*

Galeria de Imagens do CSS

O CSS pode ser criado para criar uma galeria de imagens.

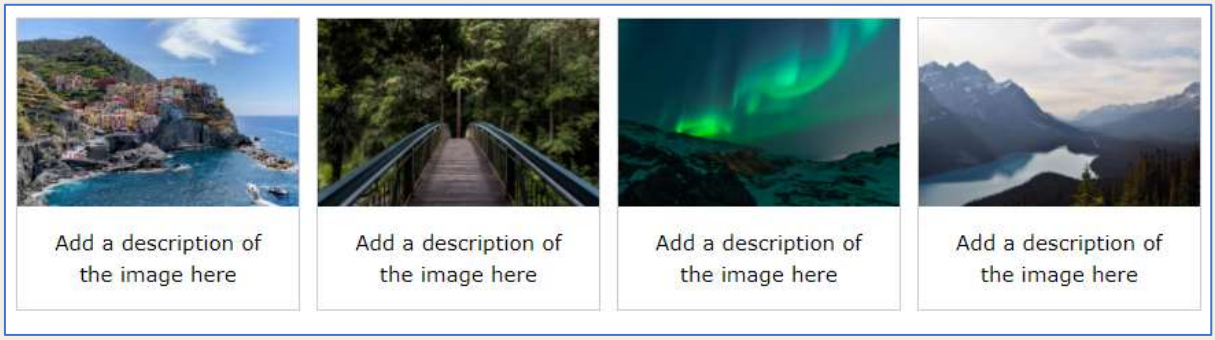


Figura 53 *– Galeria de Imagens (**Fonte:** https://www.w3schools.com/css/css_image_gallery.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Sprites de imagem do CSS

- Um sprite de imagem é uma coleção de imagens colocadas numa única imagem.
- Uma página web com muitas imagens pode demorar muito tempo a carregar e gera múltiplos pedidos de servidor.
- A utilização de sprites de imagem reduzirá o número de pedidos de servidor e poupará a largura de banda.

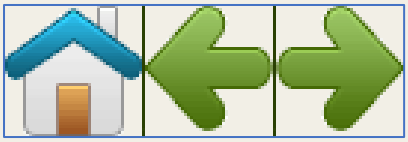


Figura 54 *– Sprites de imagem (**Fonte:** https://www.w3schools.com/css/css_image_sprites.asp)*

Selecionadores de atributos do CSS

O seletor [attribute] é usado para selecionar elementos com um atributo especificado.

- O exemplo seguinte seleciona todos os elementos com um atributo alvo:

```
a[target] {
  background-color: yellow;
}
```

Figura 55 *– Selecionador de atributos (**Fonte:** https://www.w3schools.com/css/css_attribute_selectors.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Formulários do CSS

O aspeto de um formulário HTML pode ser fortemente incrementado com o CSS:

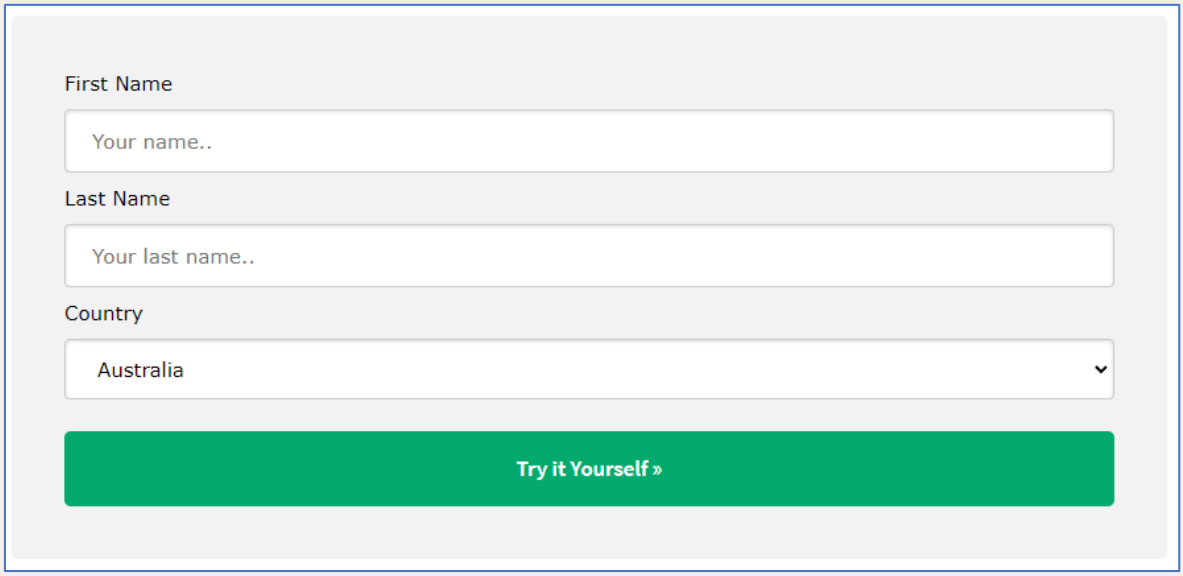


***Figura 56** – Formulários do CSS (**Fonte:** https://www.w3schools.com/css/css_form.asp)*

Contadores do CSS

Os contadores do CSS são "variáveis" mantidas pelo CSS cujos valores podem ser incrementados por regras do próprio CSS (para rastrear quantas vezes são utilizados). Os contadores permitem ajustar a aparência do conteúdo com base na sua colocação no documento.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

***Figura 57** – Contador do CSS (**Fonte:** https://www.w3schools.com/css/css_counters.asp)*

Layout do website do CSS

Um site geralmente é dividido em cabeçalhos, menus, conteúdo e rodapé:

***Figura 58** – Layout de um website (**Fonte:** https://www.w3schools.com/css/css_website_layout.asp)*

Unidades do CSS

O CSS tem várias unidades diferentes para expressar um comprimento. Muitas propriedades do CSS tomam valores de "comprimento", tais como largura, margem, preenchimento, tamanho da fonte, etc.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Unit	Description
cm	centimeters
mm	millimeters
in	inches (1in = 96px = 2.54cm)
px *	pixels (1px = 1/96th of 1in)
pt	points (1pt = 1/72 of 1in)
pc	picas (1pc = 12 pt)

Figura 59 *– Units (**Fonte:** https://www.w3schools.com/css/css_units.asp)*

Especificidade no CSS

Se houver duas ou mais regras CSS que apontem para o mesmo elemento, o selecionador com o maior valor de especificidade irá prevalecer, e a sua declaração de estilo será aplicada a esse elemento HTML. Devemos olhar para a especificidade como uma pontuação que determina qual a declaração de estilo que, em última análise, é aplicado a um elemento.

```
<html>
<head>
  <style>
    p {color: red;}
  </style>
</head>
<body>

<p>Hello World!</p>

</body>
</html>
```

Figura 60 *– Especificidade no CSS (**Fonte:** https://www.w3schools.com/css/css_specificity.asp)*

Neste exemplo, utilizámos o elemento "p" como selector e especificámos uma cor vermelha para este elemento. O texto será vermelho.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

A regra “!important” no CSS

A regra "!important" no CSS é usada para acrescentar mais importância a uma propriedade/valor do que o normal. Na verdade, se a regra "!important" for usada, esta substituirá **todas** as regras de estilo anteriores para essa propriedade específica nesse elemento.

```
#myid {
  background-color: blue;
}

.myclass {
  background-color: gray;
}

p {
  background-color: red !important;
}
```

Figura 61 *– A regra !important (**Fonte:** https://www.w3schools.com/css/css_important.asp)*

A função math no CSS

As funções matemáticas do CSS permitem que expressões matemáticas sejam usadas como valores de propriedade. Algumas funções matemáticas são: funções calc(), max() e min().

Exemplo: o uso de calc() para calcular a largura de um elemento.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```
#div1 {
  position: absolute;
  left: 50px;
  width: calc(100% - 100px);
  border: 1px solid black;
  background-color: yellow;
  padding: 5px;
}
```

Figura 62 *– Função matemática no CSS (**Fonte:** https://www.w3schools.com/css/css_math_functions.asp)*

3.4. CSS – nível avançado

Cantos arredondados do CSS#

A propriedade border-radius define o raio dos cantos de um elemento. Esta propriedade permite adicionar cantos arredondados aos elementos.

Figura 63 *– Cantos arredondados no CSS, ex.1 (**Fonte:** https://www.w3schools.com/css/css3_borders.asp)*

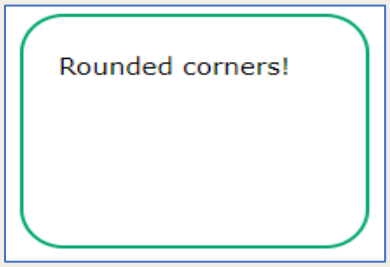


Figura 64 *– Cantos arredondados no CSS, ex.2 (**Fonte:** https://www.w3schools.com/css/css3_borders.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

A propriedade *border-image*

A propriedade *border-image* permite especificar uma imagem a ser utilizada no lugar do limite de margem convencional em torno de um elemento.

Esta propriedade tem três pontos a ter em conta:

- A imagem a ser usada como limite de margem;
- O local onde se faz o corte da imagem;
- Definir se as secções intermédias devem ser repetidas ou esticadas.

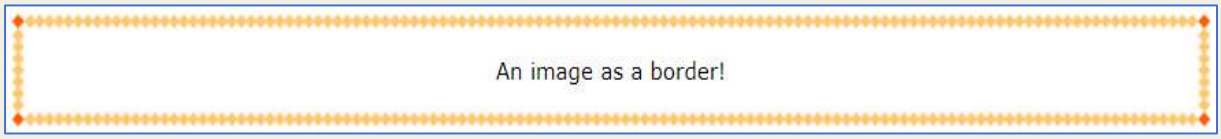


Figura 65 *– A propriedade border-image (**Fonte:** https://www.w3schools.com/css/css3_border_images.asp)*

Fundos do CSS

CSS permite adicionar várias imagens de fundo para um elemento, através da propriedade background-image.

O exemplo seguinte tem duas imagens de fundo, a primeira imagem é uma flor (alinhada para baixo e para a direita) e a segunda imagem é um fundo de papel (alinhada para o canto superior esquerdo).

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```
#example1 {
  background-image: url(img_flwr.gif), url(paper.gif);
  background-position: right bottom, left top;
  background-repeat: no-repeat, repeat;
}
```

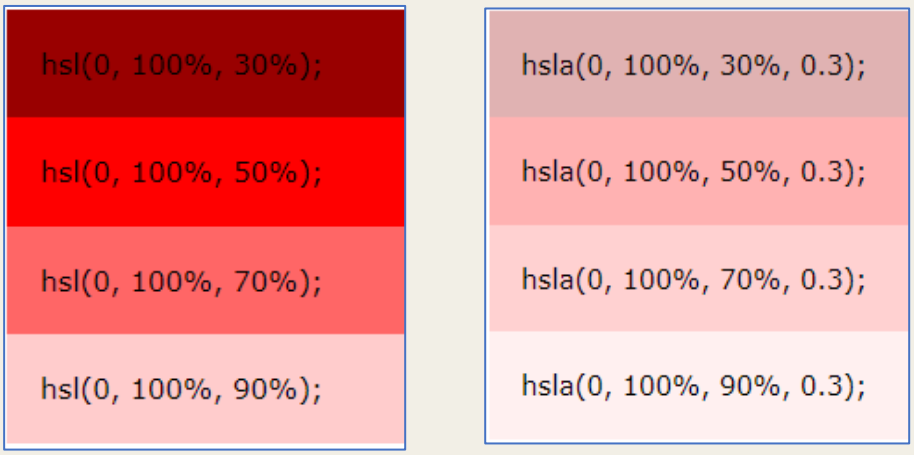
Figura 66 *– A propriedade background-image no CSS (**Fonte:** https://www.w3schools.com/css/css3_backgrounds.asp)*

Figura 67 *– Imagem de fundo (**Fonte:** https://www.w3schools.com/css/css3_backgrounds.asp)*

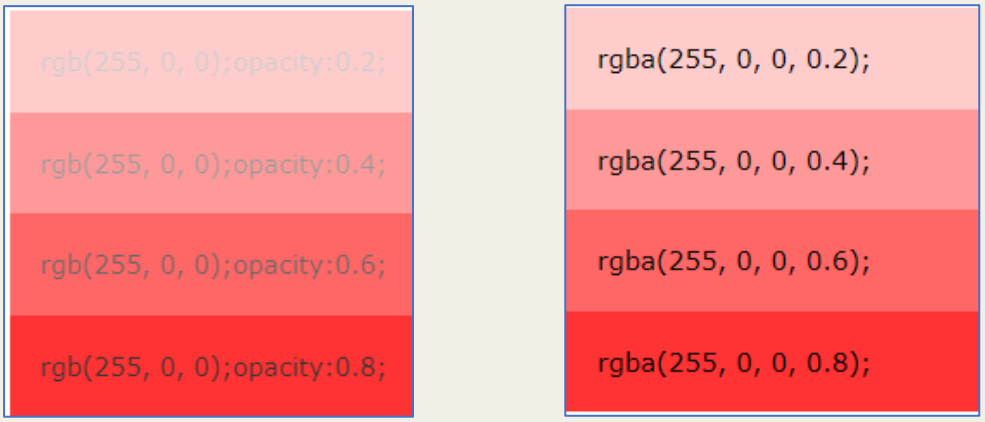
Cores no CSS

O CSS suporta mais de 140 nomes de cores, valores HEX, valores RGB, valores RGBA, valores HSL, valores HSLA e opacidade.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.



Figuras 68 e 69 *– Parâmetros HSL e HSLA do CSS (**Fonte:** https://www.w3schools.com/css/css3_colors.asp)*



Figuras 70 e 71 *– Parâmetros RGB e RGBA do CSS (**Fonte:** https://www.w3schools.com/css/css3_colors.asp)*

Palavras-chave relativas a cores no CSS

Esta secção propõe-se a explicar de que tratam as palavras-chave *transparent, currentcolor* e inherit.

A palavra-chave *transparent* é utilizada para tornar uma cor transparente. Esta é frequentemente utilizada para fazer uma cor de fundo transparente para um elemento.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

A palavra-chave *currentcolor* é como uma variável que detém o valor atual da propriedade da cor de um elemento. Esta palavra-chave pode ser útil se quiser que uma cor específica seja consistente num elemento ou numa página.

A palavra-chave inherit especifica que uma propriedade deve herdar o seu valor do seu "elemento-pai". A palavra-chave *inherit* pode ser usada para qualquer propriedade CSS e em qualquer elemento HTML.

Gradiência no CSS

Os gradientes CSS permitem mostrar transições suaves entre duas ou mais cores especificadas.

CSS define três tipos de gradientes:

- Gradientes Lineares (abaixo/acima/à esquerda/à direita/na diagonal);
- Gradientes radiais (definidos pelo seu centro);
- Gradientes Cónicos (girados em torno de um ponto central).

Figura 72 *– Gradiência nos fundos do CSS (**Fonte:** https://www.w3schools.com/css/css3_gradients.asp)*

Sombras no CSS

Com o CSS, é possível adicionar sombras ao texto e aos elementos.
Falamos das seguintes propriedades:

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

- Text-shadow
- Box-shadow

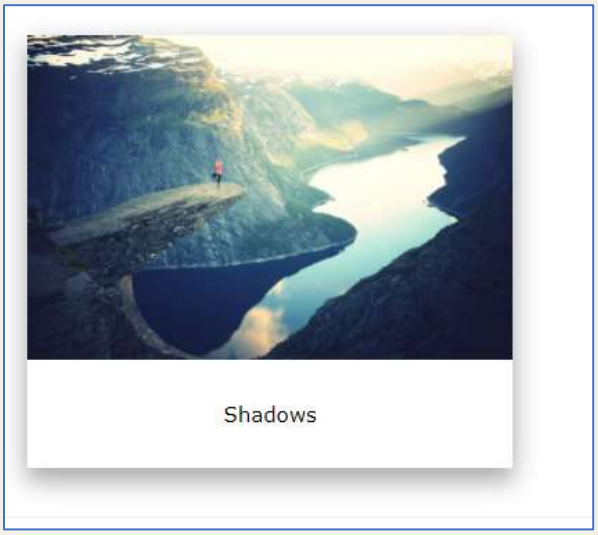


***Figura 73** – Exemplo de sombras numa caixa (**Fonte:** https://www.w3schools.com/css/css3_shadows.asp)*

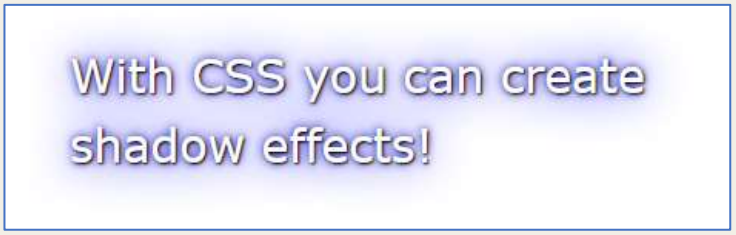


***Figura 74** – Exemplo de sombras em texto (**Fonte:** https://www.w3schools.com/css/css3_shadows.asp)*

Efeitos de texto no CSS

Neste capítulo, aprenderás sobre as funcionalidades seguintes: text-overflow, word-wrap, word-break, writing-mode.

- A propriedade text-overflow especifica como o conteúdo transbordado que não é exibido deve ser sinalizado ao utilizador.
- A propriedade CSS word-wrap permite que palavras longas possam ser quebradas e colocadas na linha seguinte.
- A propriedade CSS word-break especifica as regras de quebra de linha.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

- A propriedade writing-mode especifica se as linhas de texto são dispostas horizontalmente ou verticalmente.

Tipos de letra da web do CSS

Os tipos de letra da web permitem aos *web designers* utilizar tipos de letra que não estão instalados no computador do utilizador. Quando tiver encontrado/comprado a fonte que deseja utilizar, basta incluir o ficheiro da fonte no seu servidor web, e este será automaticamente descarregado para o utilizador quando necessário. As fontes a utilizar são definidas com a regra font-face.

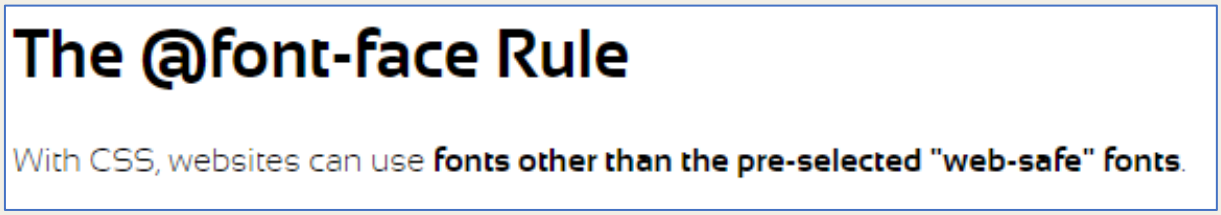


Figura 75 *– A regra font-face (**Fonte:** https://www.w3schools.com/css/css3_fonts.asp)*

Transformações em 2D do CSS

As transformações CSS permitem mover, girar, dimensionar e inclinar elementos.

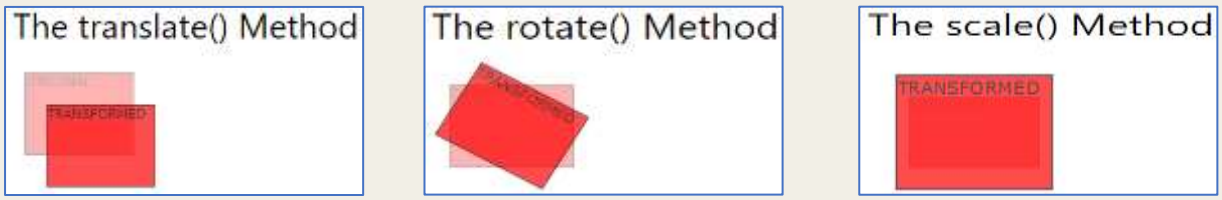


Figura 76 *– Transformações em 2D do CSS e os seus 3 métodos – translate(), rotate() e scale() (**Fonte:** https://www.w3schools.com/css/css3_2dtransforms.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Transformações em 3D do CSS

Com a propriedade de transformação CSS, é possível utilizar os seguintes métodos de transformação 3D:

- RotateX()
- RotateY()
- RotateZ()

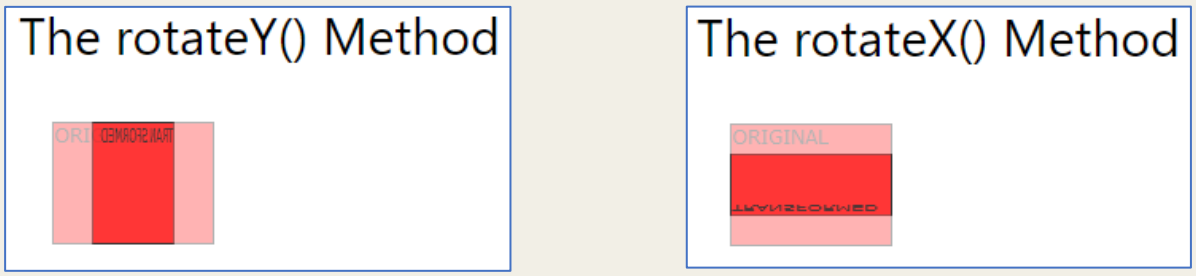


***Figura 78** – Métodos de transformação 3D no CSS (**Fonte:** https://www.w3schools.com/css/css3_3dtransforms.asp)*

Transições no CSS

As transições CSS permitem que se altere os valores das propriedades sem problemas, durante um determinado período.

Para criar um efeito de transição, é necessário especificar duas coisas:

- a propriedade CSS a que se pretende acrescentar um efeito;
- a duração do efeito.

O exemplo seguinte mostra um elemento vermelho de 100px * 100px. O elemento também especificou um efeito de transição para a propriedade da largura, com uma duração de 2 segundos:

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```
div {
  width: 100px;
  height: 100px;
  background: red;
  transition: width 2s;
}
```

Figura 79 *– Transições do CSS (**Fonte:** https://www.w3schools.com/css/css3_transitions.asp)*

Animações do CSS

Uma animação permite que um elemento mude gradualmente de um estilo para outro. É possível alterar quantas propriedades CSS forem desejadas, as vezes que forem necessárias. Para utilizar animação CSS, é necessário primeiro especificar alguns quadros-chave para a animação. Os quadros-chave contêm os estilos que o elemento terá em determinados momentos.

Estas são as principais propriedades da animação:

- `@keyframes`
- `animation-name`
- `animation-duration`
- `animation-delay`
- `animation-iteration-count`
- `animation-direction`
- `animation-timing-function`
- `animation-fill-mode`
- `animation`

Figura 80 *– Animações do CSS (**Fonte:** https://www.w3schools.com/css/css3_animations.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Tooltips do CSS

- Uma *tooltip* é frequentemente utilizada para especificar informação extra sobre algo quando o utilizador move o ponteiro do rato sobre um elemento:

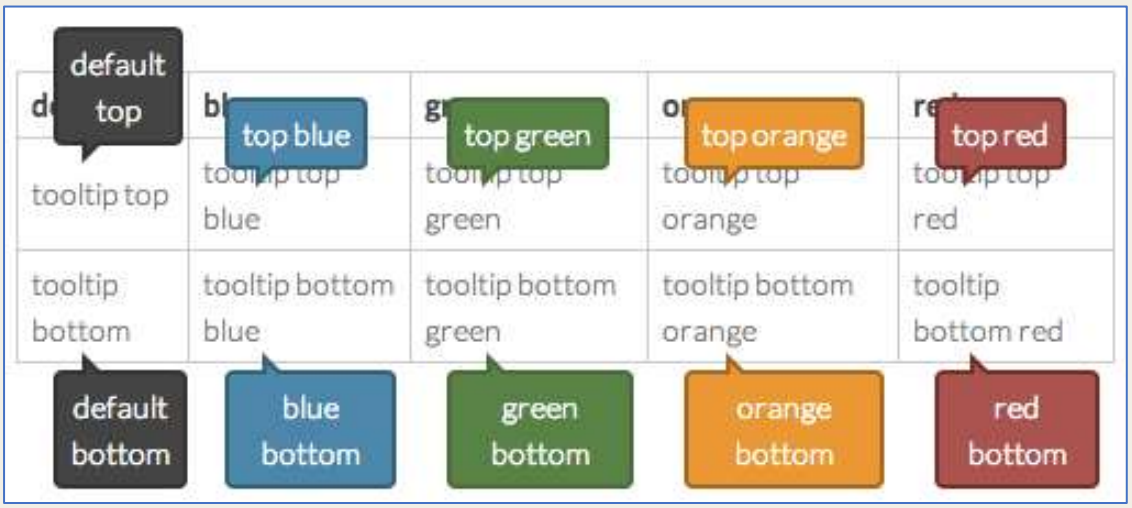


Figura 81 *– Tooltip do CSS (**Fonte:** https://www.w3schools.com/css/css3_animations.asp)*

Estilizar imagens no CSS

A utilização do CSS para estilizar imagens permite especificar uniformemente como as imagens devem aparecer no website com apenas alguns conjuntos de regras, tais como adicionar uma margem, mudar a forma e o tamanho da imagem, etc.

Figura 82 *– Imagens estilizadas a partir do CSS (**Fonte:** https://www.w3schools.com/css/css3_image_reflection.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

A propriedade box-reflect do CS

A propriedade box-reflect é utilizada para criar um reflexo de imagem.O valor da propriedade box-reflect pode ser: abaixo, acima, esquerda ou direita.

Figura 83 – *Image reflection (**Fonte:** https://www.w3schools.com/css/css3_image_reflection.asp)*

A propriedade object-fit do CSS

A propriedade do CSS é utilizada para especificar como uma <img> ou <video> deve ser redimensionado para caber no seu recipiente. Essa propriedade informa o conteúdo para preencher o recipiente de várias maneiras; como "preservar essa proporção" ou "esticar e ocupar o máximo de espaço possível".

Veja-se a seguinte imagem de Paris. Esta imagem tem 400 pixels de largura e 300 pixels de altura:

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

***Figura 84** – A propriedade object-fit (**Fonte:** https://www.w3schools.com/css/css3_object-fit.asp)*

A propriedade object-position do CSS

Digamos que a parte da imagem que é mostrada não está posicionada como desejamos. Para posicionar a imagem, utilizaremos a propriedade object-position. Aqui utilizaremos a propriedade object-position para posicionar a imagem de modo a que o grande edifício antigo esteja no centro:

***Figura 85** – A propriedade object-position (**Fonte:** https://www.w3schools.com/css/css3_object-position.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Propriedade de máscara no CSS

Com a máscara CSS cria-se uma camada de máscara para colocar sobre um elemento para esconder porções do elemento, seja de forma parcial ou totalmente. A propriedade mask-image do CSS especifica uma imagem de camada de máscara.

A imagem da camada de máscara pode ser uma imagem PNG, uma imagem SVG, um gradiente CSS ou um elemento SVG.

Figura 86 *– Propriedade de máscara no CSS (**Fonte:** https://www.w3schools.com/css/css3_masking.asp)*

Botões do CSS

Existem muitas propriedades para estilizar um botão no CSS. Abaixo segue um exemplo:

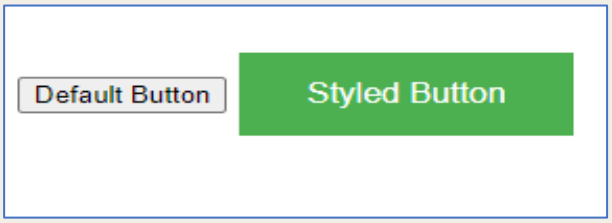


Figura 87 *– O botão "convencional" (**Fonte:** https://www.w3schools.com/css/css3_buttons.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```css
.button {
  background-color: #4CAF50;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  font-size: 16px;
  margin: 4px 2px;
}
```

***Figura 88** – Código para conceber um botão "convencional" (**Fonte:** https://www.w3schools.com/css/css3_buttons.asp)*

Paginação no CSS

Caso tenhas um website com muitas páginas, poderás desejar adicionar algum tipo de paginação a cada página. Estes são alguns exemplos:

***Figura 90** – Paginação no CSS – ex.1 (**Fonte:** https://www.w3schools.com/css/css3_pagination.asp)*

***Figura 91** – Paginação no CSS – ex.2 (**Fonte:** https://www.w3schools.com/css/css3_pagination.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Múltiplas colunas no CSS

O *layout* multi-column do CSS permite a fácil definição de múltiplas colunas de texto, tal como se vê nos jornais:

Figura 92 *– Múltiplas Colunas no CSS (**Fonte:** https://www.w3schools.com/css/css3_multiple_columns.asp)*

Propriedades de interface do utilizador no CSS

Neste capítulo, aprenderás sobre as seguintes propriedades da interface de utilizador CSS:

- A propriedade resize especifica se (e como) um elemento deve ser redimensionável pelo utilizador.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

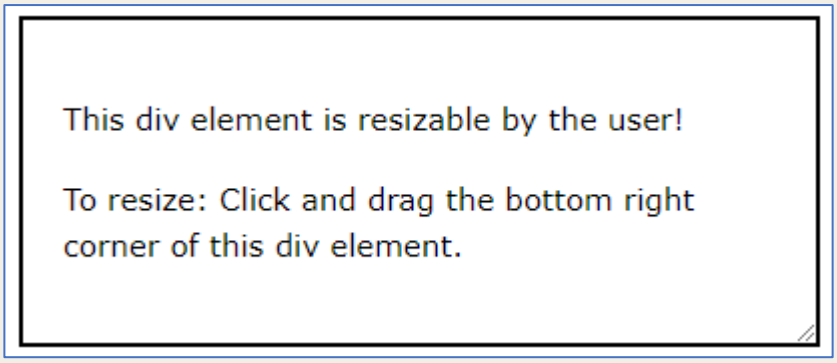


Figura 93 *– Como aplicar a propriedade resize (**Fonte:** https://www.w3schools.com/css/css3_user_interface.asp)*

- A propriedade outline-offset adiciona espaço entre o contorno e o canto ou limite de um elemento.

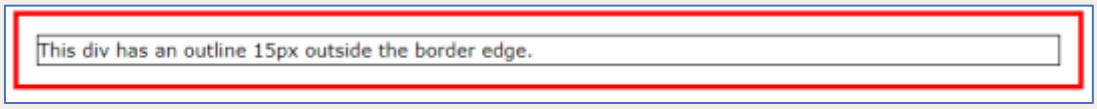


Figura 94 *– A propriedade outline-offset (**Fonte:** https://www.w3schools.com/css/css3_user_interface.asp)*

Variáveis do CSS

A função var() é utilizada para inserir o valor de uma variável CSS.

As variáveis CSS têm acesso ao DOM, o que significa que pode criar variáveis com alcance local ou global, alterar as variáveis com JavaScript, e alterar as variáveis com base em consultas de media.

Uma boa maneira de aplicar variáveis CSS dá-se ao tratar das cores de um design. Em vez de copiar e colar as mesmas cores uma e outra vez, poderemos colocá-las em variáveis.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

A propriedade de box-sizing no CSS

A propriedade box-sizing em CSS define como o utilizador deve calcular a largura e altura total de um elemento, ou seja, se preenchimento e bordas devem ser incluídos ou não.

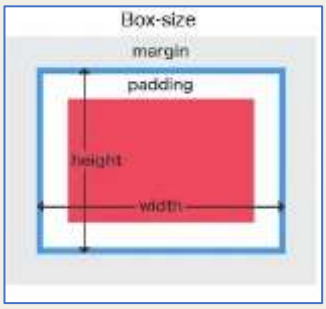


Figura 95 *– A propriedade de box-sizing (**Fonte:** https://www.w3schools.com/css/css3_box-sizing.asp)*

Consultas de aparelhos multimédia no CSS

As consultas de aparelhos multimédia no CSS3 alargaram o espetro dos tipos de multimédia da variante CSS2: agora, em vez de procurarem um tipo de dispositivo, olham para a capacidade do dispositivo.

As consultas por parte de aparelhos multimédia podem ser usadas para verificar várias coisas, tais como:

- largura e altura da janela de visualização;
- largura e altura do dispositivo;
- orientação (o tablet/telefone está no modo paisagem ou retrato?);
- resolução.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

A utilização de consultas de media é uma técnica popular para fornecer uma folha de estilo adaptada aos computadores de secretária, portáteis, tablets e *smartphones* (tais como iPhone e Android).

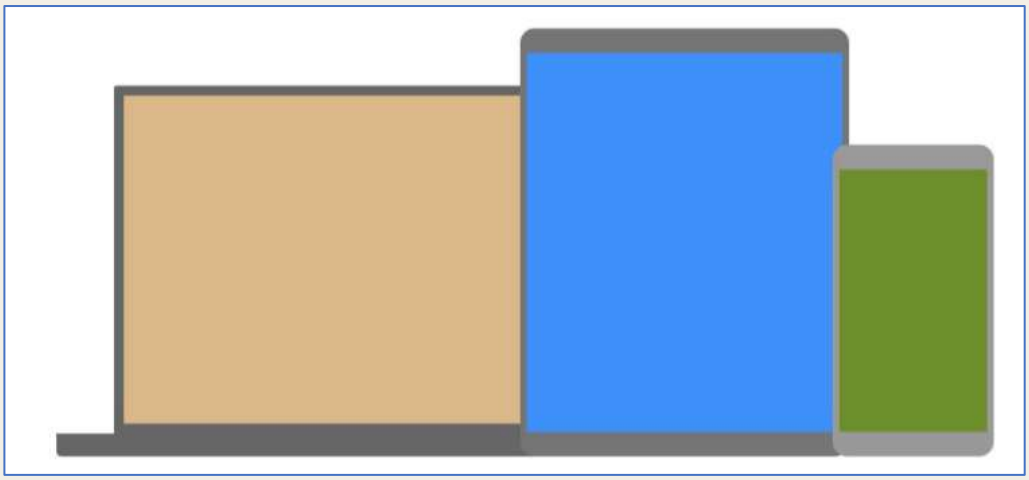


***Figura 96** – Exemplo de consultas de aparelhos multimédia no CSS (**Fonte:** https://www.w3schools.com/css/css3_mediaqueries_ex.asp)*

O módulo de layout Flexbox

O Módulo de Layout de Flexbox (“Caixa Flexível”) torna mais fácil projetar uma estrutura de layout reativa e flexível sem usar flutuação ou posicionamento.

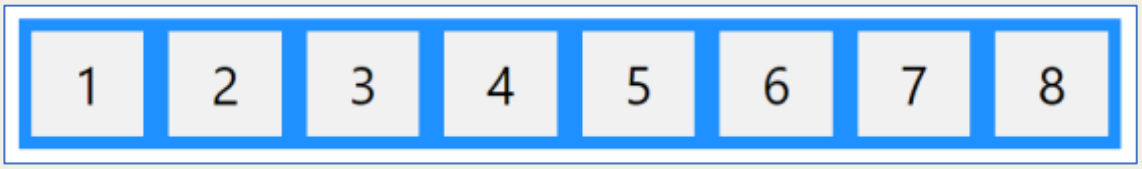


***Figura 97** – O módulo de layout Flexbox (**Fonte:** https://www.w3schools.com/css/css3_flexbox.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

3.2. Web Design Reativo do CSS

Introdução ao WDR

O design reactivo da web faz com que a tua página web tenha bom aspecto em todos os dispositivos.

Web design reativo usa apenas HTML e CSS.

As páginas da web podem ser visualizadas usando muitos dispositivos diferentes: desktops, tablets e telefones. A tua página da web deve ter uma boa aparência e ser fácil de usar, independentemente do dispositivo.

As páginas Web não devem deixar de fora a informação para caber em dispositivos mais pequenos, mas sim adaptar o seu conteúdo para caber em qualquer dispositivo:

Figura 98 *– Exemplo do WDR (**Fonte:** https://www.w3schools.com/css/css_rwd_intro.asp)*

Ao ato de usar CSS e HTML para redimensionar, esconder, encolher, ampliar, ou mover o conteúdo para o fazer parecer bem em qualquer ecrã chamamos "*web design reativo*".

Janela de visualização

A janela de visualização ("*viewport*") é a área visível do utilizador de uma página web.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

O *viewport* varia com o dispositivo e será mais pequeno num telemóvel do que num ecrã de computador. Antes dos *tablets* e *smartphones*, as páginas web eram concebidas apenas para ecrãs de computador, e era comum que as páginas web tivessem um design estático e um tamanho fixo.

Então, quando começamos a navegar na internet usando tablets e telefones celulares, as páginas da web de tamanho fixo eram muito grandes para caber na janela de visualização. Para corrigir isso, os navegadores desses dispositivos reduziram toda a página da Web para caber no ecrã.

Esta não foi, de todo, uma solução perfeita! Tratou-se de uma solução rápida.

Configuração do *viewport*

O HTML5 introduziu um método para deixar os *web designers* assumirem o controlo do *viewport*, através da tag <meta>.

Deverá incluir-se o seguinte elemento de viewport <meta> em todas as suas páginas web:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Figura 99 *– Definição do viewport (**Fonte:** https://www.w3schools.com/css/css_rwd_viewport.asp)*

Isto dá ao navegador instruções sobre como controlar as dimensões e escalas da página. A fração *width=device-width* define a largura da página para seguir a largura do ecrã do dispositivo (que variará dependendo do dispositivo).

A parte *initial-scale=1.0* define o nível de *zoom* inicial quando a página é carregada pela primeira vez pelo navegador.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Aqui está um exemplo de uma página web sem a meta tag *viewport*, e a mesma página web com a meta tag *viewport*:

Figura 100 *– Páginas com e sem viewport (**Fonte:** https://www.w3schools.com/css/css_rwd_viewport.asp)*

Visualização em Grelha

Muitas páginas web são baseadas numa visualização em grelha, o que significa que a página é dividida em colunas:

Figura 101 *– Visualização em grelha (**Fonte:** https://www.w3schools.com/css/css_rwd_grid.asp)*

Muitas páginas *web* são baseadas numa visualização em grelha, o que significa que a página é dividida em colunas:

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Figura 102 *– Divisão da página em colunas (**Fonte:** https://www.w3schools.com/css/css_rwd_grid.asp)*

Construir uma visualização em grelha reativa

Primeiro, é necessário certificar-se de que todos os elementos HTML têm a propriedade *box-sizing* definida como *border-box*. Isso garante que o preenchimento e o limite de margem sejam incluídos na largura e altura totais dos elementos.

O seguinte código deve ser adicionado:

```
* {
  box-sizing: border-box;
}
```

Figura 103 *– Construir uma visualização em grelha reativa (**Fonte:** https://www.w3schools.com/css/css_rwd_grid.asp)*

O exemplo seguinte mostra uma página web simples e reativa, com duas colunas:

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

25%	75%

Figura 104 *– Visualização em grelha reativa (**Fonte:** https://www.w3schools.com/css/css_rwd_grid.asp)*

```
.menu {
  width: 25%;
  float: left;
}
.main {
  width: 75%;
  float: left;
}
```

Figura 105 *– Codificação para grelhas reativas (**Fonte:** https://www.w3schools.com/css/css_rwd_grid.asp)*

Consultas de aparelhos multimédia

Como já foi visto, as consultas de aparelhos multimédia em WDR são uma técnica introduzida no CSS3. Utiliza-se a regra *@media* para incluir um bloco de propriedades CSS apenas se uma determinada condição for verdadeira.

Se a janela do navegador for 600px ou menor, a cor de fundo será azul-claro:

```
@media only screen and (max-width: 600px) {
  body {
    background-color: lightblue;
  }
}
```

Figura 106 *– Consultas de aparelhos multimédia em WDR (**Fonte:** https://www.w3schools.com/css/css_rwd_mediaqueries.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Adicionar um ponto de interrupção

Para este propósito, as consultas de aparelhos multimédia podem ser úteis. Podemos adicionar um ponto de interrupção onde certas partes do *design* se comportarão de maneira diferente em cada lado do ponto de interrupção.

Figura 107 *– Consultas de aparelhos multimédia em WRD (**Fonte:** https://www.w3schools.com/css/css_rwd_mediaqueries.asp)*

Deve usar-se uma consulta de aparelhos multimédia para adicionar um ponto de interrupção em 768px. Exemplo: Quando o ecrã (janela do navegador) fica menor que 768px, cada coluna deve ter uma largura de 100%.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```
/* For desktop: */
.col-1 {width: 8.33%;}
.col-2 {width: 16.66%;}
.col-3 {width: 25%;}
.col-4 {width: 33.33%;}
.col-5 {width: 41.66%;}
.col-6 {width: 50%;}
.col-7 {width: 58.33%;}
.col-8 {width: 66.66%;}
.col-9 {width: 75%;}
.col-10 {width: 83.33%;}
.col-11 {width: 91.66%;}
.col-12 {width: 100%;}

@media only screen and (max-width: 768px) {
  /* For mobile phones: */
  [class*="col-"] {
    width: 100%;
  }
}
```

Figura 108 *– Consulta de aparelhos multimédia em WDR (**Fonte:** https://www.w3schools.com/css/css_rwd_mediaqueries.asp)*

Imagens em WRD

Se a propriedade largura for definida para uma percentagem e a propriedade altura for definida para "auto", a imagem será reativa e escalada para cima e para baixo:

```
img {
  width: 100%;
  height: auto;
}
```

Figura 109 *– Código de imagens em WRD (**Fonte:** https://www.w3schools.com/css/css_rwd_images.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Se a propriedade max-width estiver definida como 100%, a imagem será reduzida se for necessário, mas nunca será maior do que seu tamanho original:

```
img {
  max-width: 100%;
  height: auto;
}
```

Figura 110 *– A propriedade max-width (**Fonte:** https://www.w3schools.com/css/css_rwd_images.asp)*

Note-se que no exemplo acima, a imagem pode ser ampliada para ser maior do que o seu tamanho original. Uma solução melhor, em muitos casos, será a utilização da propriedade max-width.

Emprego da propriedade max-width:

A propriedade max-width é definida como 100%, a imagem será reduzida se for necessário, mas nunca será aumentada para ser maior que seu tamanho original:

```
img {
  max-width: 100%;
  height: auto;
}
```

Figura 111 *– A propriedade max-width (**Fonte:** https://www.w3schools.com/css/css_rwd_images.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Vídeos em DRW

Se a propriedade width estiver definida como 100%, o reprodutor de vídeo responderá, aumentando ou diminuindo em conformidade:

```
video {
  width: 100%;
  height: auto;
}
```

***Figura 112** – Vídeo em DRW – a propriedade width (**Fonte:** https://www.w3schools.com/css/css_rwd_videos.asp)*

Se a propriedade max-width estiver definida como 100%, o player de vídeo será reduzido se necessário, mas nunca será maior do que o tamanho original:

```
video {
  max-width: 100%;
  height: auto;
}
```

***Figura 113** – Vídeo em DRW – a propriedade max-width (**Fonte:** https://www.w3schools.com/css/css_rwd_videos.asp)*

Observe-se que, no exemplo acima, o reprodutor de vídeo pode ser dimensionado para ser maior do que seu tamanho original. Uma solução melhor, em muitos casos, será usar a propriedade max-width.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Emprego da propriedade max-width:

Se a propriedade max-width estiver definida como 100%, o player de vídeo será reduzido se necessário, mas nunca será maior do que o tamanho original:

```
video {
  max-width: 100%;
  height: auto;
}
```

Figura 114 *– Propriedade max-width em vídeo (**Fonte:** https://www.w3schools.com/css/css_rwd_videos.asp)*

Frameworks em WRD

Existem vários frameworks no CSS gratuitos, que possibilitam um design reativo. Uma ótima maneira de criar um design responsivo é usar uma folha de estilo reativa, como a W3.CSS, que facilita o desenvolvimento de sites, que ficam bem em qualquer tamanho.

Outra estrutura popular é o Bootstrap, que utiliza HTML e CSS para fazer páginas web reativas.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

3.3. Grelhas no CSS

Introdução

O CSS Grid Layout Module oferece um sistema de layout baseado em grelha, com linhas e colunas, facilitando a conceção de páginas web sem ter de usar flutuadores e posicionamento.

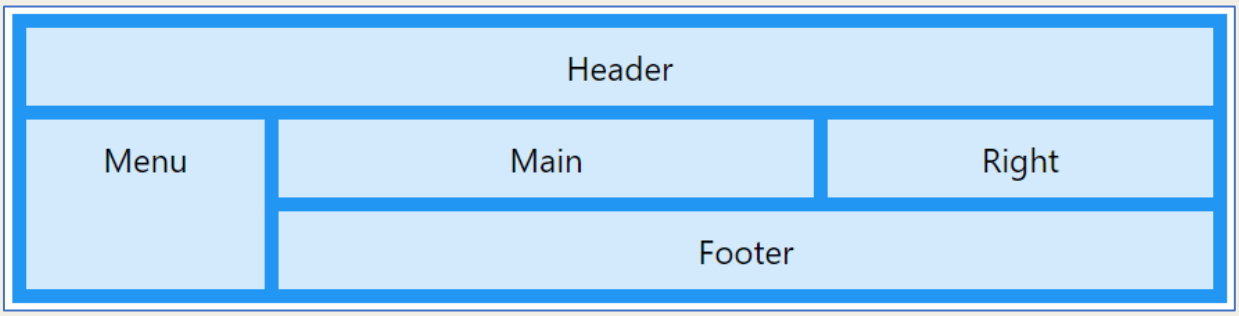


Figura 115 *– Grelhas no CSS (**Fonte:** https://www.w3schools.com/css/css_grid.asp)*

Grid Containers

To make an HTML element behave as a grid container, you have to set the display property to grid or inline-grid. Para que um elemento HTML se comporte como um *grid container*, é necessário definir a propriedade de exibição como *grid* ou *inline-grid*.

Grid containers consistem em itens de grelha colocados dentro de colunas e filas.

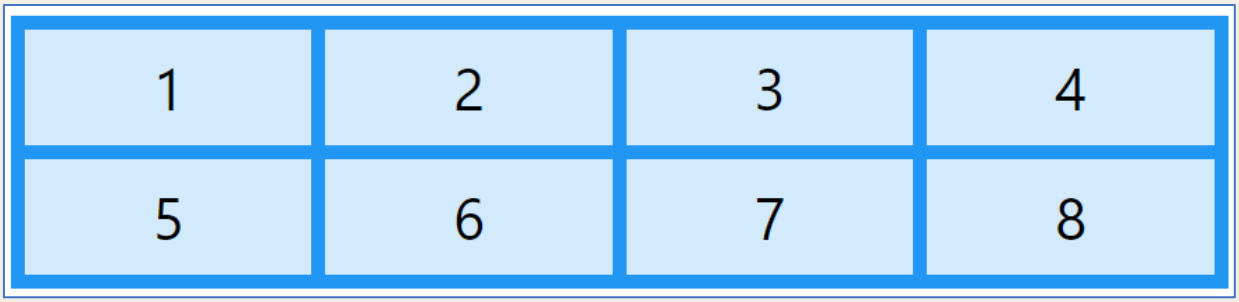


Figura 116 *– Grid Containers (**Fonte:** https://www.w3schools.com/css/css_grid_container.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Elementos da *grid*

Uma grid *container* contém elementos de *grid*. Por defeito, um *container* possui um item de grelha para cada coluna, em cada linha, mas pode estilizar os itens da grelha de modo a que abranjam várias colunas e/ou filas. A propriedade grid-column define em qual(is) coluna(s) colocar um elemento. Aqui, define-se onde o elemento irá começar, e onde irá terminar.

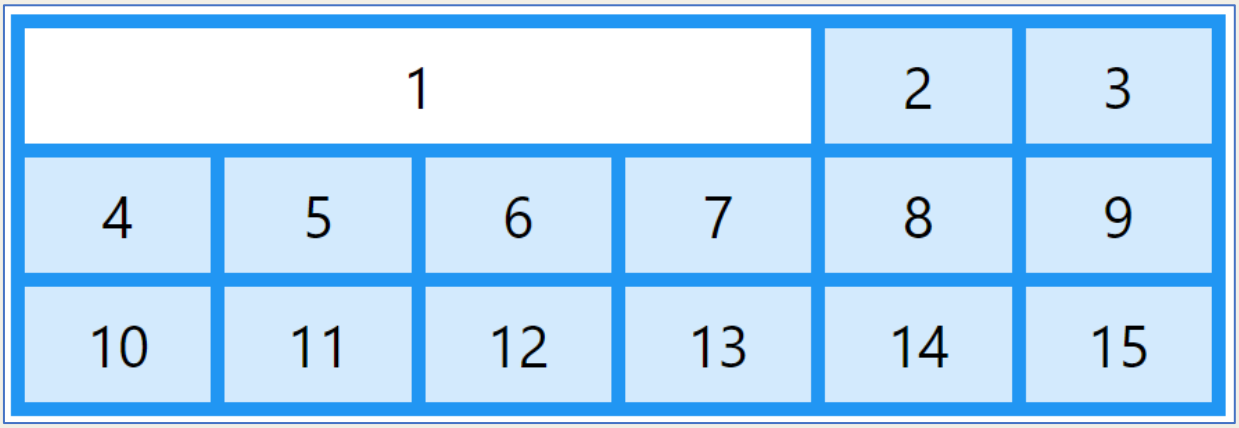


Figura 117 *– Elementos da grid (**Fonte:** https://www.w3schools.com/css/css_grid_item.asp)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

3.5. CSS SASS

Introdução

O que significa SASS?

SASS significa "Syntactically Awesome Stylesheet" (*Folha de Estilos Sintaticamente Fantásticos*), tratando-se de um pré-processador de CSS, compatível com todas as suas versões. É conhecido por reduzir repetições, poupando imenso tempo. Foi concebido por Hampton Catlin e desenvolvido por Natalie Weizenbaum, em 2006. O seu download e utilização são gratuitos.

Porquê usar SASS?

As folhas de estilo estão a ficar maiores, mais complexas, e mais difíceis de manter. É aqui que um pré-processador de CSS pode ajudar.

O SASS permite-te usar recursos que não existem no CSS, como variáveis, *nested rules*, *mixins*, importações, heranças, funções internas, entre outros.

Um exemplo simples do porquê de SASS ser útil

Digamos que temos um website com três cores principais:

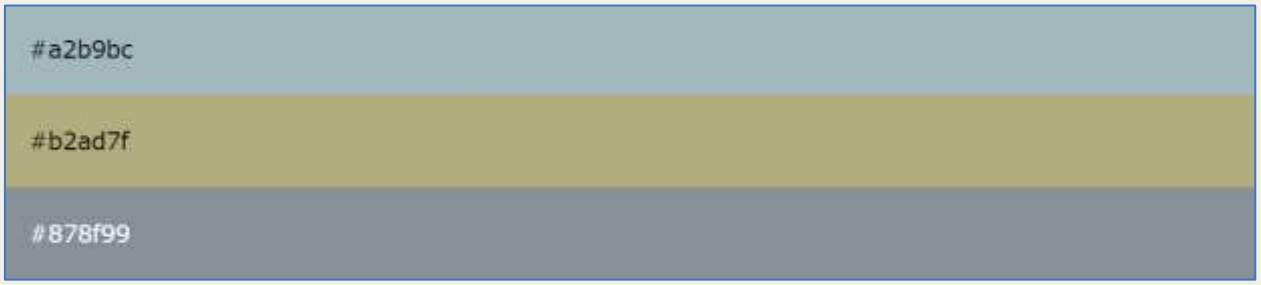


Figura 118– *Cores em SASS (**Fonte:** https://www.w3schools.com/sass/sass_intro.php)*

Então, quantas vezes é necessário digitar esses valores HEX? Exato, muitas vezes.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

E as variações das mesmas cores?

Em vez de digitar muitas vezes os valores acima, poderá usar-se “SASS” e digitar o seguinte:

```
/* define variables for the primary colors */
$primary_1: #a2b9bc;
$primary_2: #b2ad7f;
$primary_3: #878f99;

/* use the variables */
.main-header {
  background-color: $primary_1;
}

.menu-left {
  background-color: $primary_2;
}

.menu-right {
  background-color: $primary_3;
}
```

***Figura 119** – Uma das variáveis “SASS” (**Fonte:** https://www.w3schools.com/sass/sass_variables.php)*

Assim, quando se usa SASS e a cor primária muda, só é necessário mudá-la num único local.

Como funciona o SASS?

Um navegador não entende o código SASS. Por conseguinte, necessitarás de um pré-processador SASS para converter o código SASS em CSS padrão.

Este processo chama-se “*transpiling”*. Portanto, é necessário dar a um *transpiler* (algum tipo de programa) algum código SASS e depois obter algum código CSS de volta.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Tipos de ficheiros do SASS

Os ficheiros SASS têm a extensão de ficheiro “. scss".

Comentários do SASS

O SASS suporta comentários CSS padrão /* comment */ e, além disso, suporta comentários inline // comment:

```
/* define primary colors */
$primary_1: #a2b9bc;
$primary_2: #b2ad7f;

/* use the variables */
.main-header {
  background-color: $primary_1; // here you can put an inline comment
}
```

Figura 120 *– Comentários do SASS (**Fonte:** https://www.w3schools.com/sass/sass_intro.php)*

Variáveis do SASS

As variáveis são uma forma de armazenar informação que se poderá reutilizar mais tarde.

Através do SASS, é possível armazenar informação em variáveis, como por exemplo:

- *strings*
- números
- cores
- booleanos

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

- listas
- *nulls*

O SASS utiliza o símbolo $, seguido de um nome, para declarar variáveis:

```
$variablename: value;
```

Figura 121 *– Variável do SASS (**Fonte:** https://www.w3schools.com/sass/sass_variables.php)*

O exemplo seguinte declara quatro variáveis, denominadas "myFont", "myColor", "myFontSize" e "myWidth". Depois das variáveis serem declaradas, poder-se-á utilizá-las onde for necessário:

```
$myFont: Helvetica, sans-serif;
$myColor: red;
$myFontSize: 18px;
$myWidth: 680px;

body {
  font-family: $myFont;
  font-size: $myFontSize;
  color: $myColor;
}

#container {
  width: $myWidth;
}
```

Figura 122 *– SASS variable (**Fonte:** https://www.w3schools.com/sass/sass_variables.php)*

Assim, quando o ficheiro Sass é transposto, toma as variáveis (myFont, myColor, etc.) e produz CSS normais com os valores das variáveis colocadas no CSS da seguinte forma:

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```
body {
  font-family: Helvetica, sans-serif;
  font-size: 18px;
  color: red;
}

#container {
  width: 680px;
}
```

Figura 123 *– Variáveis SASS afetas ao tipo de letra (**Fonte:** https://www.w3schools.com/sass/sass_variables.php)*

O âmbito das variáveis SASS

As variáveis de baixo nível só estão disponíveis ao nível de nidificação ("nesting") onde são definidas.

Observando o seguinte exemplo:

```
$myColor: red;

h1 {
  $myColor: green;
  color: $myColor;
}

p {
  color: $myColor;
}
```

Figura 124 *– Variável SASS de cor de letra (**Fonte:** https://www.w3schools.com/sass/sass_variables.php)*

A cor do texto dentro da tag <p> será vermelha ou verde? Pois, será vermelha!

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

A outra definição, $myColor: green, insere-se dentro da regra <h1> e estará apenas disponível aí.

Desta forma, o resultado CSS será o seguinte:

```
h1 {
  color: green;
}

p {
  color: red;
}
```

Figura 125 *– Variável SASS relativa ao tipo de letra (**Fonte:** https://www.w3schools.com/sass/sass_variables.php)*

Este é o comportamento por defeito para o âmbito das variáveis em SASS.

Como usar o SASS !global

O comportamento padrão para o âmbito variável pode ser anulado utilizando a função *!global*. !global indica que uma variável é global, o que significa que é acessível a todos os níveis. Observe-se o seguinte exemplo (o mesmo que acima; mas com o !global acrescentado):

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```
$myColor: red;

h1 {
  $myColor: green !global;
  color: $myColor;
}

p {
  color: $myColor;
}
```

Figura 126 *– !global no SASS (**Fonte:** https://www.w3schools.com/sass/sass_variables.php)*

Agora a cor do texto dentro da tag <p> será verde!

Assim, o resultado seria o seguinte:

```
h1 {
  color: green;
}

p {
  color: green;
}
```

Figura 127 *– Resultado da função !global na cor verde (**Fonte:** https://www.w3schools.com/sass/sass_variables.php)*

Regras "*nested*" do SASS

O SASS permite fazer o "*nesting*" (encadear um seletor dentro de outro) de seletores da mesma forma que o HTML. Vejamos um exemplo de um código SASS para a navegação de um site:

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```
nav {
  ul {
    margin: 0;
    padding: 0;
    list-style: none;
  }
  li {
    display: inline-block;
  }
  a {
    display: block;
    padding: 6px 12px;
    text-decoration: none;
  }
}
```

***Figura 128** – O "nesting" do SASS (**Fonte:** https://www.w3schools.com/sass/sass_nesting.php)*

Note-se que, em SASS, o "*nesting*" dos seletores ul e li foi feito dentro do seletor nav. Enquanto no CSS, as regras são definidas uma a uma (não é feito o "*nesting*"):

```
nav ul {
  margin: 0;
  padding: 0;
  list-style: none;
}
nav li {
  display: inline-block;
}
nav a {
  display: block;
  padding: 6px 12px;
  text-decoration: none;
}
```

***Figura 129** – Disposição do "nesting" em SASS (**Fonte:** https://www.w3schools.com/sass/sass_nesting.php)*

Sendo possível fazer o "*nesting*" de propriedades em SASS, a leitura torna-se mais limpa e simples em relação ao CSS padrão.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Propriedades em "nesting" do SASS

Muitas propriedades do CSS têm o mesmo prefixo, como font-family, font-size e font-weight ou text-align, text-transform e text-overflow. Com o SASS, poderão digitar-se como propriedades em "*nesting*":

```
font: {
  family: Helvetica, sans-serif;
  size: 18px;
  weight: bold;
}

text: {
  align: center;
  transform: lowercase;
  overflow: hidden;
}
```

Figura 130 *– Propriedades em "nesting" do SASS (**Fonte:** https://www.w3schools.com/sass/sass_nesting.php)*

O "*transpiler*" SASS converterá o código acima referido em CSS normal:

```
font-family: Helvetica, sans-serif;
font-size: 18px;
font-weight: bold;

text-align: center;
text-transform: lowercase;
text-overflow: hidden;
```

Figura 131 *– O "transpiler" do SASS em ação (**Fonte:** https://www.w3schools.com/sass/sass_nesting.php)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

A diretiva @import e os "*partials*"

O SASS incorpora o código CSS de forma DRY ("Don't Repeat Yourself" - Não Se Repita a Si Mesmo). Uma forma de digitar o código em DRY passa por manter o código respetivo em ficheiros separados.
É possível criar pequenos ficheiros com trechos de CSS para incluir noutros ficheiros SASS. Exemplos de tais ficheiros podem ser: ficheiro de reset, variáveis, cores, tipos de letra, tamanhos de fontes, etc.

Importar Ficheiros no SASS

Assim como o CSS, o SASS também suporta a diretiva @import. A diretiva @import permite incluir o conteúdo de um arquivo noutro. A diretiva CSS @import tem um grande inconveniente devido a questões de desempenho, dado que cria um pedido HTTP extra cada vez que é invocada. No entanto, a diretiva SASS @import inclui o ficheiro no CSS e, assim, não é necessário invocar o HTTP de novo enquanto o ficheiro executa.

```
@import filename;
```

Figura 132 *– Importar ficheiros no SASS (**Fonte:** https://www.w3schools.com/sass/sass_import.php)*

Dica: Não é necessário especificar uma extensão de ficheiro, dado que o SASS assume automaticamente que é feita uma referência a um ficheiro .sass ou .scss. Também é possível importar ficheiros CSS. A diretiva @import importa o ficheiro e quaisquer variáveis ou mixins definidas no ficheiro importado podem então ser utilizadas no ficheiro principal.

Poderás importar todos os ficheiros que precisares no ficheiro principal:

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```
@import "variables";
@import "colors";
@import "reset";
```

Figura 133 *– Importação de vários elementos usando a diretiva @import (**Fonte:** https://www.w3schools.com/sass/sass_import.php)*

Vejamos um exemplo: Vamos supor que temos um arquivo de *reset* chamado "reset.scss", semelhante ao exemplo seguinte:

```
html,
body,
ul,
ol {
  margin: 0;
  padding: 0;
}
```

Figura 134 *– Características de um elemento prestes a ser importado no SASS (**Fonte:** https://www.w3schools.com/sass/sass_import.php)*

E agora queremos importar o ficheiro "reset.scss" para outro ficheiro denominado "standard.scss". **Solução**: acrescentar a diretiva @import no topo de um ficheiro. Desta forma, o conteúdo do mesmo terá um âmbito global:

```
@import "reset";

body {
  font-family: Helvetica, sans-serif;
  font-size: 18px;
  color: red;
}
```

Figura 135 *– Resultado do ato de importação (**Fonte:** https://www.w3schools.com/sass/sass_import.php)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Assim, quando o ficheiro "standard.css" for transposto, o código CSS assemelhar-se-á ao seguinte:

```
html, body, ul, ol {
  margin: 0;
  padding: 0;
}

body {
  font-family: Helvetica, sans-serif;
  font-size: 18px;
  color: red;
}
```

Figura 136 *– O ficheiro "standard.css" após a sua transposição (**Fonte:** https://www.w3schools.com/sass/sass_import.php)*

Os "partials" no SASS

Por defeito, o SASS transporta todos os ficheiros .scss diretamente. No entanto, quando for pretendido importar um ficheiro, não é necessário que o mesmo seja transposto diretamente. O SASS tem um mecanismo pronto para lidar com a situação. Se o nome de um ficheiro iniciar com um "*underscore*" (_), o SASS não o transporá. Os ficheiros nomeados desta forma são chamados de "*partials*".

Desta forma, um ficheiro "*partial*" em SASS é invocado após colocação prévia de um "*underscore*":

```
_filename;
```

Figura 137 *– Exemplo de invocação de um ficheiro "partial" (**Fonte:** https://www.w3schools.com/sass/sass_import.php)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

O exemplo seguinte mostra um ficheiro "partial" de SASS denominado "_colors.scss". (Este ficheiro não será transposto diretamente para "color.css"):

```
$myPink: #EE82EE;
$myBlue: #4169E1;
$myGreen: #8FBC8F;
```

Figura 138 *– O ficheiro SASS "colors.scss" (**Fonte:** https://www.w3schools.com/sass/sass_import.php)*

Desta feita, se for importado o ficheiro "*partial*", deverá omitar-se o "*underscore*". O SASS entenderá que deve importar o arquivo "_colors.scss":

```
@import "colors";

body {
  font-family: Helvetica, sans-serif;
  font-size: 18px;
  color: $myBlue;
}
```

Figura 139 *– Resultado após importação do ficheiro "partial" (**Fonte:** https://www.w3schools.com/sass/sass_import.php)*

O "@mixin" e o "@include" em SASS

Os "*mixins*" em SASS

A diretiva @mixin permite criar código CSS que deve ser reutilizado em todo o website.

Definição de um "*mixin*"

Um "*mixin*" é definido com a diretiva @mixin.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```
@mixin name {
  property: value;
  property: value;
  ...
}
```

Figura 140 *– O @mixin em SASS (**Fonte:** https://www.w3schools.com/sass/sass_mixin_include.php)*

O exemplo a seguir cria um "*mixin*" chamado "*important-text*":

```
@mixin important-text {
  color: red;
  font-size: 25px;
  font-weight: bold;
  border: 1px solid blue;
}
```

Figura 141 *– O @mixin em SASS e seus elementos (**Fonte:** https://www.w3schools.com/sass/sass_mixin_include.php)*

Usar um *mixin*

A diretiva @*include* é usada para incluir um arquivo *mixin*.

```
selector {
  @include mixin-name;
}
```

Figura 142 *– A diretiva @include (**Fonte:** https://www.w3schools.com/sass/sass_mixin_include.php)*

Assim, de forma a incluir o *mixin* important-text criado acima, dever-se-á:

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```
.danger {
  @include important-text;
  background-color: green;
}
```

Figura 143 *– A importância do important-text (**Fonte:** https://www.w3schools.com/sass/sass_mixin_include.php)*

O "*transpiler*" SASS converterá o código acima referido em CSS normal:

```
.danger {
  color: red;
  font-size: 25px;
  font-weight: bold;
  border: 1px solid blue;
  background-color: green;
}
```

Figura 144 *– Conversão do SASS em CSS normal (**Fonte:** https://www.w3schools.com/sass/sass_mixin_include.php)*

Um *mixin* também pode incluir outros *mixins*:

```
@mixin special-text {
  @include important-text;
  @include link;
  @include special-border;
}
```

Figura 145 *– Presença de um mixin noutro mixin (**Fonte:** https://www.w3schools.com/sass/sass_mixin_include.php)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Como passar variáveis para um mixin

Os *mixins* aceitam argumentos. Desta forma, é possível passar variáveis para um *mixin*:

```
/* Define mixin with two arguments */
@mixin bordered($color, $width) {
  border: $width solid $color;
}

.myArticle {
  @include bordered(blue, 1px);  // Call mixin with two values
}

.myNotes {
  @include bordered(red, 2px); // Call mixin with two values
}
```

Figura 146 *– Passar variáveis para um mixin através de argumentos (**Fonte:** https://www.w3schools.com/sass/sass_mixin_include.php)*

Nota que os argumentos são definidos como variáveis e, de seguida, usados como os valores (cor e largura) da propriedade de limite de margem.

Após a compilação, o código CSS terá o seguinte aspeto:

```
.myArticle {
  border: 1px solid blue;
}

.myNotes {
  border: 2px solid red;
}
```

Figura 147 *– Código de CSS após compilação (**Fonte:** https://www.w3schools.com/sass/sass_mixin_include.php)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

A diretiva @extend no SASS

A diretiva @extend permite compartilhar um conjunto de propriedades CSS de um seletor para outro. A diretiva @extend é especialmente útil caso existam elementos de estilo quase idênticos, que diferem apenas em alguns pequenos detalhes.
O seguinte exemplo de SASS cria primeiro um estilo básico para botões (este estilo será utilizado para a maioria dos botões). De seguida, criaremos um estilo para um botão "Report" e um estilo para um botão "Submit". Tanto o botão "Report" como o botão "Submit" herdam todas as propriedades CSS da classe .button-basic, através da diretiva @extend. Além disso, têm as suas próprias cores definidas, como pode ser visto no exemplo seguinte:

```
.button-basic  {
  border: none;
  padding: 15px 30px;
  text-align: center;
  font-size: 16px;
  cursor: pointer;
}

.button-report  {
  @extend .button-basic;
  background-color: red;
}

.button-submit  {
  @extend .button-basic;
  background-color: green;
  color: white;
}
```

Figura 148 *– A diretiva extend no SASS (**Fonte:** https://www.w3schools.com/sass/sass_extend.php)*

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Após a compilação, o código CSS terá o seguinte aspeto:

```
.button-basic, .button-report, .button-submit {
  border: none;
  padding: 15px 30px;
  text-align: center;
  font-size: 16px;
  cursor: pointer;
}

.button-report  {
  background-color: red;
}

.button-submit  {
  background-color: green;
  color: white;
}
```

***Figura 149** – A diretiva extend após compilação no código CSS (**Fonte:** https://www.w3schools.com/sass/sass_extend.php)*

Ao utilizar a diretiva @extend, não é necessário especificar várias classes para um elemento no código HTML, como por exemplo "Report This". É apenas necessário especificar ".button-report" para obter ambos os conjuntos de estilos.

A diretiva @extend possibilita que o código em SASS se mantenha bastante *DRY*.

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.