



Melhorar as capacidades digitais e de programação de reclusos

Manual Formação

(Produto Intelectual 2)



Project number: 2018-1-RO01-KA204-049298

Programa

FREE TO CODE - *Improving Digital and Coding Skills for Inmates*

Parceiros

European Strategies Consulting (Romania)
APROXIMAR – Cooperativa de Solidariedade Social, Crl (Portugal)
Asturia vzw (Belgium)
Avaca Technologies Consulting, Informatics AE (Greece)
BeCode (Belgium)
Dlearn – European Digital Learning Network (Italy)
Penitenciarul Bucuresti-Jilava (Romania)

Design

Aproximar, Cooperativa de Solidariedade Social

Data de publicação

Fevereiro de 2020

www.free2code-initiative.eu

All rights reserved.



Asturia vzw



avaca
TECHNOLOGIES



European
Strategies
Consulting

/// Index

a }	Introdução geral.....1
------------	-------------------------------

b (Módulo 0.....3
	Internet e informações web
	Contexto
	Internet
	Web

Referências bibliográficas	38
----------------------------------	----

Anexos

Plano Sessão.....	39
Exercícios.....	51

c >	Módulo 1.....7
	HTML
	Introdução.....7
	HTML 5.....9
	Semântica.....10

d /	Módulo 2.....13
	CSS
	Introdução ao CSS.....13
	Seletores.....21
	Propriedades básicas.....25
	Posicionamento.....26
	Animações CSS33

a } Introdução geral

Introdução geral



Hello World!

Desde que os computadores e a Internet apareceram, mudaram profundamente a forma como jogamos, como comunicamos, como trabalhamos, como vivemos. Devido a estas profundas mudanças na nossa Sociedade, existem muitos empregos disponíveis para pessoas que aprenderam a programar e resolver problemas através da programação de sites e aplicações web.

Será que pensa que precisa de uma licenciatura em Matemática e Engenharia Nuclear para ser capaz de programar? De modo algum! Programar é fácil quando acontece o "click": até crianças conseguem aprender a programar... Porque na verdade é muito divertido! Se perguntar a um programador o que fazem o dia todo, eles dirão coisas como: "é como brincar com legos." Ou "é como cozinhar, e escrever receitas...".

O que se pode esperar

O objetivo desta formação é permitir que experimente a programação e pensamento lógico, para ver se também o acha divertido e quem sabe, se é algo que gostaria de fazer quando sair em liberdade.

Neste curso, aprenderá os fundamentos da criação e programação de sites utilizando as 3 principais linguagens da web: HTML, CSS e JavaScript. No final da formação, estará mais do que pronto a ingressar num bootcamp de programação que o ajudará a atingir um nível que o permitirá encontrar emprego como programador!

Aprender fazendo

A melhor forma de aprender a programar, tal como na cozinha... É praticando! É por isso que o curso é organizado numa série de exercícios que deve completar para que possa passar ao próximo passo. Não se preocupe em ser rápido, não é como na escola! O sistema lembrar-se-á do seu progresso para não ter que recomeçar de uma sessão para a outra.

...Não funciona! Sou péssimo nisto!!

Não, não é. Programar é tentar até conseguir. Todos falhamos muito no começo. Habitue-se, o fracasso faz parte do processo de aprendizagem. Continue a tentar, eventualmente irá conseguir!

Uma palavra final da equipa



Pronotos? Vamos lá!

b (Module 0

Internet e informações Web

Contexto

@ A Unترنت

_ Uma rede de redes

Antes da invenção da internet, muitas universidades e instituições já possuíam alguma forma de infraestrutura de informática, que consistia em computadores conectados entre si como redes locais. Utilizando esses computadores, é possível, portanto, publicar conteúdo e comunicar dentro da própria instituição, mas não globalmente.

Para escrever este conteúdo, as pessoas utilizavam uma inovação simples, mas poderosa: o hipertexto. Hipertexto significa texto que pode conter "hiperlinks", que é uma forma sofisticada de descrever um link que aponta para textos localizados noutras partes da rede.

Mas, dado o facto das só serem acessíveis localmente, os textos só podiam ser vinculados e acedidos localmente. Assim, se uma investigadora de uma universidade quisesse consultar um documento de outra universidade, ela teria que dirigir no seu carro até lá, para conseguir conectar à rede local da universidade e, por fim, aceder ao documento. Não é muito prático, verdade?

Um cientista britânico chamado Tim Berners-Lee, que trabalhava no CERN, inventou em 1989 uma solução de software que permitia às pessoas publicar conteúdo e torná-lo acessível a qualquer pessoa que tivesse o endereço, em todas as redes. É por isso que a Internet é frequentemente denominada rede de redes. E assim nasceu o primeiro site! Ainda é possível visitá-lo no seguinte endereço: <http://info.cern.ch>

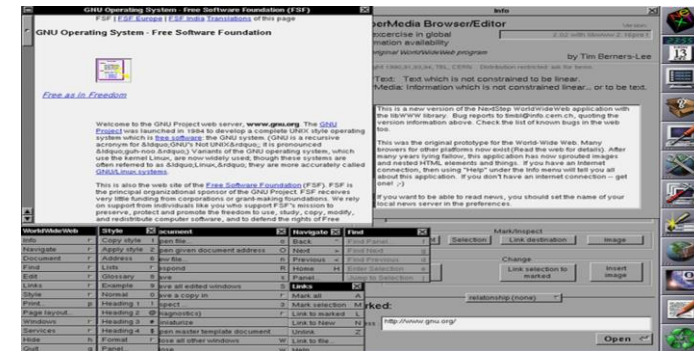


Fonte:

https://en.wikipedia.org/wiki/Tim_Berners-Lee

_ Um navegador para navegar na web

Agora, para visitar este primeiro site (e muitos outros que se seguiram), precisava-se de um outro software. Tim Berners-Lee criou assim o chamado "navegador WorldWideWeb", o primeiro navegador da Web. Esta primeira versão era muito básica, permitindo apenas mostrar texto e hiperlinks



Captura de tela do primeiro navegador desenvolvido por Sir Tim Berners-Lee - Autor da imagem: Tim Berners-Lee (domínio público) - Fonte: <https://en.wikipedia.org/wiki/WorldWideWeb>

_ A internet, hoje

Avançando até 2021: os navegadores da web modernos podem exibir não apenas textos e links, mas também imagens, vídeos e executar complexas completos!

Portanto, qualquer pessoa pode agora não apenas consumir, mas também criar novos conteúdos: a internet tornou-se a principal plataforma para a maioria das nossas vidas profissionais, mas também privadas. Comunicar com amigos e familiares, colaborar com colegas, gerir a sua conta bancária ou planear as suas férias ... Tudo agora acontece online, graças à Internet!

_ A internet não é a web

A Internet é a infraestrutura técnica que torna a Web possível. É uma rede composta por milhões de computadores espalhados pelo mundo, conectados entre si por meio de diversos tipos de conexões (linhas telefónicas, cabos de fibra, cabos ethernet, WiFi, Bluetooth ...) e routers que garantem que as transferências de informações chegam aos seus destinos.

Portanto, enquanto a Internet é a infraestrutura, a Web é um serviço construído em cima da infraestrutura. Um serviço capacitado por um software específico chamado ... servidores web! Quando visita um site utilizando um navegador da Internet, o navegador atua como um "cliente", solicitando o conteúdo relevante ao "servidor web".

Como referido no MDN¹ (Mozilla Developer Network - um dos melhores recursos para aprender desenvolvimento web):

- “Os clientes são os dispositivos típicos de utilizadores web conectados à Internet (por exemplo, o seu computador conectado à sua Wi-Fi ou o seu telemóvel conectado à sua rede móvel) e o software de acesso à web disponível nesses dispositivos (geralmente um navegador da web como o Firefox ou Chrome).
- Servidores são computadores que armazenam páginas da web, sites ou aplicações. Quando um dispositivo cliente deseja aceder a uma página da web, uma cópia da página da web é transferida do servidor para a máquina do cliente para ser exibida no navegador web do utilizador.”

Pertanto, quando si digita o si clicca su un indirizzo web, ciò che accade è che il browser innesca una richiesta al server web pertinente, localizzato a quell'indirizzo. Il server riceve l'URL come richiesta, e restituisce il contenuto rilevante al "client" (il tuo browser).

¹ Fronte: How the Web works? - MDN - https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/How_the_Web_works

_ Linguagem de Marcação de HiperTexto

Já mencionamos o Hipertexto como uma grande inovação. Vamos dar uma olhada em toda a linguagem de codificação que o habilita: HTML (Hypertext Markup Language).

HTML é muito fácil de aprender ... Então, vamos começar com isso!

c > Módulo 1

HTML

Introdução ao HTML

@ Porquê utilizar HTML

Os computadores, e os navegadores web por extensão, são inerentemente burros. Não podemos esperar que um computador consiga ler um documento de texto e fazer uma distinção entre títulos, parágrafos, links, listas, etc.

Vamos colocar-nos no lugar de um computador e olhar para um documento que provavelmente não entenderá:



Assim como não podemos dar sentido ao texto no exemplo acima, os computadores não podem dar sentido aos dados que obtêm da web. Por isso, utilizamos uma linguagem de marcação para dar significado adicional ao texto. Na web, a linguagem de marcação que utilizamos é HTML ou linguagem de marcação hipertexto.



Ao adicionarmos algumas `<tags>` ao documento que descreve o conteúdo, chegamos a uma melhor compreensão do significado e estrutura do texto. A área linguística que se preocupa com o significado do texto chama-se semântica. Ao escrever HTML, é importante utilizar as tags HTML corretas para o seu conteúdo. Isto é chamado de escrita semântica HTML, e a sua importância não pode ser subestimada

_ O que é HTML?

HTML é uma linguagem de programação utilizada para marcar conteúdo. Então, dizemos que é uma “linguagem de marcação”. A marcação significa que, graças ao HTML, pode definir a estrutura e o significado do seu conteúdo. De acordo com o MDN, “o HTML consiste numa série de elementos, que utiliza para incluir, ou encapsular, diferentes partes do conteúdo para controlar a sua semântica e aparência. As tags encapsuladas podem criar uma hiperligação de palavra ou imagem” para um lugar diferente na página ou na internet ou apontar para um arquivo de vídeo (que o navegador pode renderizar dentro de um reprodutor de vídeo) e assim por diante. Vamos dar um exemplo simples:

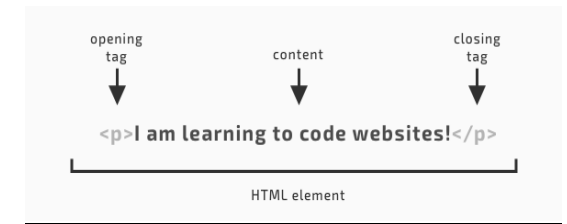
“I am learning to code websites”

Se quiséssemos que a linha ficasse por si só, poderíamos especificar que se trata de um parágrafo, encapsulando-o em tags de parágrafo:

```
<p> I am learning to code websites
</p>
```

_HTML elementp (“HTML tags”)

Vamos examinar esse elemento de parágrafo um pouco mais de perto. Do que isso é feito?

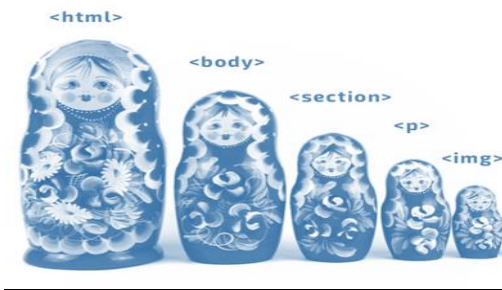


Como pode ver, as principais partes do nosso elemento são: i:

1. A tag de abertura: Consiste no nome do elemento (neste caso, p) circundado por um sinal de menor do que (<) e um sinal de maior que (>). Diz onde o elemento inicia ou começa a produzir efeito - neste caso em que o parágrafo começa.
2. A tag de fecho: É o mesmo que a tag de abertura, excetuando que inclui uma barra antes do nome do elemento. Diz onde o elemento termina - neste caso, em que o parágrafo termina. Não adicionar uma tag de fecho é um dos erros de principiante mais habituais e pode levar a resultados estranhos.
3. O conteúdo: Consiste no conteúdo do elemento que, neste caso, é apenas texto.
4. O elemento: A tag de abertura, a tag de fecho e o conteúdo juntos definem o elemento.

_Elementos agrupados

Tal como as matryoshkas os elementos HTML podem ser agrupados noutros elementos HTML.



Continuemos com o elemento HTML do exemplo anterior.

Gostaria de dar ênfase à palavra code. Para isso, vou utilizar o elemento `` e colocá-lo antes e depois da palavra que quero enfatizar.

```
< p>I am learning to <em>code</em> websites! </p>
```

Neste exemplo o elemento `` está agrupado no elemento `</p>`.

Importante: Quando agrupar elementos html, certifique-se de fechar o elemento agrupado antes de fechar o elemento-pai (`</p>`)!

HTML 5

@ O primeiro document HTML

_ Extensão de ficheiros

O seu computador sabe que aplicação usar para abrir um determinado ficheiro, verificando a extensão do ficheiro. Alguns exemplos são:

- Documentos terminados em **.docx** irão ser abertos pelo **Microsoft Word**.
- Documentos terminados em **.pdf** irão ser abertos pelo **leitor de PDF**.
- Documentos terminados em **.mp3** irá ser aberto por um **leitor de musica**.

Quando escreve código HTML, pretende que o seu código seja lido pelo navegador. O seu computador abrirá automaticamente todos os ficheiros que terminam em `.html` no navegador. O navegador converterá então o código HTML dentro do documento para uma página web legível.

Se quiser ver o código HTML puro, terá de utilizar um tipo diferente de aplicação para abrir estes documentos: um editor de código.

_ Code editors

Pode utilizar o editor de texto predefinido que está instalado no seu computador para escrever os seus ficheiros HTML. Por exemplo, no Windows pode usar o Bloco de Notas - que funcionará perfeitamente desde que não escreva erros no seu código HTML e de que guarde o ficheiro como um ficheiro `.html`.

Existem também aplicações criadas especificamente para escrever código. Estas aplicações têm funcionalidades incorporadas, tais como realce de código, indentação do seu código, preenchimento automático, etc. que ajudam a escrever o código de forma eficiente.

@ Estrutura de ficheiros

```
<!DOCTYPE html>
<html>
  <head>
    <title>The Most Simple HTML page in the world</title>
  </head>
  <body>
    <h1>The Most Simple HTML page in the world</h1>
    <p>It might just be this page.</p>
  </body>
</html>
```

Exemplo de um documento html simples

Vamos analisar o exemplo acima de um documento de HTML.

- **<!DOCTYPE html>** A primeira linha de um documento html começa com a declaração doctype. Esta linha diz ao navegador qual a versão do html que está a utilizar.

- **<html>...</html>** Todo o seu código html será colocado dentro do elemento html. O elemento html consiste numa tag de abertura: `<html>`, e uma tag de fecho: `</html>`.

- **<head>...</head><body>...</bod>**

Dentro do elemento html temos um elemento head (cabeça) e um elemento body (corpo). O elemento head contém informações sobre o documento. Não será visível para a pessoa que consulta a sua página web. O elemento body contém conteúdo real que será visível no site..

- **<title>...</title>** O elemento title (título) faz parte do head. Tal como o nome sugere, contém o título para a página html.

- **<h1>...</h1>** O elemento h1 é colocado no body do documento. É um dos 6 cabeçalhos de secção em html. (h1-h6). "h" significa "headline" (cabeçalho).

- **<p>...</p>** O elemento p ou o elemento parágrafo é utilizado para exibir texto.

Semântica

@ Porquê utilizar semântica?

A semântica acontece quando atribuímos um significado específico a um signo. Considere a imagem abaixo.



Aprendemos a atribuir um significado a esta imagem: uma seta que aponta numa direção.

No entanto, um cão, por exemplo, ou um gato, não entenderia isso ao olhar para a placa abaixo.

Na verdade, é difícil para nós, humanos, olhar para este sinal e não inferir a sua semântica porque os nossos cérebros adoram semântica!

Utilizamos semântica a toda a hora, em todos os lugares. Ajuda-nos a navegar nas nossas vidas e nas cidades, por exemplo. Aprendemos as convenções (como os sinais de trânsito) e confiamos nelas para nos dizer o que está por vir. Nós vemos, nós sabemos.

Em HTML, quando dizemos que um pedaço de texto é um parágrafo, o navegador "sabe" como renderizá-lo de uma forma específica. Por exemplo, um software de leitor de tela pode ler o parágrafo num tom específico. Isso é particularmente útil com imagens, por exemplo. Adicionar sintaxe adequada a uma tag de imagem pode melhorar ou piorar a compreensão do conteúdo por parte de uma pessoa cega.

@ Semântica HTML

A especificação HTML5 inclui vários elementos semânticos para ajudar a organizar documentos. Os elementos semânticos são especificamente concebidos para comunicar um significado ao navegador, programador, leitor e quaisquer outras tecnologias que interpretem o documento (por exemplo, assistentes de voz, leitores de ecrã, motores de busca...).

_ Elementos de secção em HTML5

Os elementos de secção são muito úteis para agrupar elementos semanticamente relacionados. De acordo com o MDM², existem 4 elementos diferentes que pode utilizar para organizar e agrupar semanticamente partes de html para identificar seções do seu documento:

²Utilizar secções HTML e contornos por Mozilla Contributors, licenciado sob CC-BY-SA 2.5. Fonte: https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Using_HTML_sections_and_outlines%23section_elements_in_html5

- “HTML Navigational Element (**<nav>**) define uma secção que contém links de navegação que aparecem frequentemente num site. Pode ter menus primários e secundários, mas nunca se agrupa, ou coloca um elemento **<nav>** dentro de outro elemento **<nav>**.”
- HTML Article Element (**<article>**) define um pedaço de conteúdo autónomo.
- HTML Section Element (**<section>**) define uma secção de um documento que indica um agrupamento com um significado semântico semelhante.
- HTML Aside Element (**<aside>**) define uma secção que, embora relacionada com o elemento principal, não pertence ao fluxo principal, como uma caixa explicativa ou um anúncio ou uma lista de "produtos relacionados", por exemplo. Não deve estar dentro do principal (tipicamente, um **<article>**)”.

Outros elementos semânticos HTML utilizados na Segmentação

O mesmo Mozilla Contributor avança as seguintes definições para os elementos usados na Segmentação³:

- “HTML Body Element (**<body>**) define todo o conteúdo de um documento. Contém todo o conteúdo e tags HTML

HTML Header Element (**<header>**) define uma página que normalmente contém o logótipo, título e navegação. O cabeçalho pode também ser utilizado noutros elementos semânticos, tais como **<article>**, ou **<section>**— ou cabeçalho da secção, contendo talvez o nome do

autor, etc. **<article>**, **<section>**, **<aside>**, e **<nav>**) podem ter o seu próprio **<header>**. Apesar do seu nome, não está necessariamente posicionado no início da página ou secção.

- HTML Footer Element (**<footer>**) define um rodapé de página que normalmente contém os direitos de autor, avisos legais e, por vezes, alguns links – ou rodapé de secção, contendo talvez a data de publicação da secção, informações de licença, etc. **<article>**, **<section>**, **<aside>**, e **<nav>** podem ter os seus próprios **<footer>**. Apesar do seu nome, não está necessariamente posicionado no final da página ou secção⁴.

Exemplo de um layout

O exemplo abaixo é um layout para o corpo da página de blog. Existe um **<header>** com um elemento **<h1>** e um elemento **<nav>** que contém links de navegação. Ao mesmo nível do **<header>**, temos um elemento **<section>** e um elemento **<footer>**. O principal elemento **<section>** tem um título **<h2>**, 2 **<articles>** e um **<aside>**.

```
<body>
  <header>
    <nav>
      <ul>
        <li><a href="#">link</a></li>
        <li><a href="#">link</a></li>
      </ul>
    </nav>
  </header>
  <section>
    <h2>
      My Blog Posts
    </h2>
    <article>
      <header>
        <p>
          Article Title
        </p>
      </header>
      <p>
        Content
      </p>
    </article>
    <article>
      <header>
        <p>
          Article Title
        </p>
      </header>
      <p>
        Content
      </p>
    </article>
    <aside>
      <p>
        Author info
      </p>
    </aside>
  </section>
  <footer>
    <p>
      Copyright Info
    </p>
  </footer>
</body>
```

```
<li><a href="#">link</a></li>
</ul>
</nav>
<h1>
  Page Title
</h1>
</header>
<section>
  <h2>
    My Blog Posts
  </h2>
  <article>
    <header>
      <p>
        Article Title
      </p>
    </header>
    <p>
      Content
    </p>
  </article>
  <article>
    <header>
      <p>
        Article Title
      </p>
    </header>
    <p>
      Content
    </p>
  </article>
  <aside>
    <p>
      Author info
    </p>
  </aside>
</section>
<footer>
  <p>
    Copyright Info
  </p>
</body>
```

³ Ibid

⁴ Informação adaptada de: <https://learnabouthtml5.blogspot.com/2016/12/using-html-sections-and-outlines.html>

d /Módulo 2

CSS

Introdução ao CSS

Na nossa jornada anterior, descobrimos o poder do HTML. Como já vimos, o HTML foi criado para descrever o conteúdo de uma página web, como por exemplo:

```
<h1>This is a heading</h1>
```

```
<p>This is a paragraph.</p>
```

Mas deve ter reparado que não há cor nenhuma, é apenas um simples documento com palavras. O que é chato de se ver.

Tal acontece porque, além do HTML, estas páginas utilizam outra linguagem, chamada CSS, cuja função é tornar o html visualmente mais atrativo.

Por outras palavras, o CSS permite-nos tornar isto:

In questo:



Rome Celebrates the Short, but Beautiful, Life of Raphael

Amid new coronavirus restrictions on museums in Italy, a blockbuster show traces the artist's life from finish to start.

Arts March 6

Fonte:

<https://www.nytimes.com/2020/03/06/arts/raphael-rome-coronavirus.html>

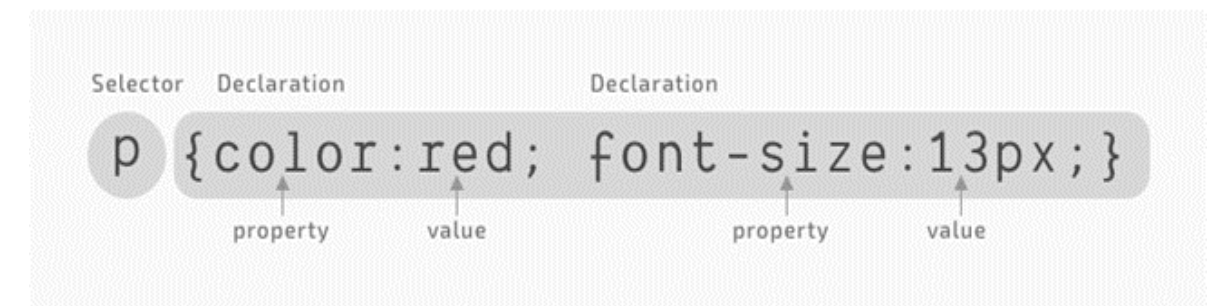
@ O que é o CSS?

- CSS significa Cascading Style Sheets (Folhas de Estilo Cascata)
- CSS descreve como os elementos HTML devem ser exibidos no ecrã
- Consegue controlar o layout de múltiplas páginas ao mesmo tempo

Resumindo, o CSS é o que faz com que as páginas web pareçam visualmente apresentáveis. É uma competência imperdível para qualquer programador web.

@ A sintaxe CSS

Veja este esquema, que resume como escrever CSS de forma a que o navegador compreenda.



Como pode ver, há palavras muito bonitas aqui. Não entre em pânico, não haverá muito mais. :-)

"**Selector**" (seletor) indica quais os elementos do seu ficheiro HTML que devem receber instruções. Indica qual o elemento HTML que pretende modelar. Neste exemplo, todas as tags <h1> na página html receberão as propriedades CSS atribuídas ao h1 css selector.

O bloco de "declaration" (declaração) contém uma ou mais declarações separadas por ponto e vírgula (;).

Cada declaração inclui um nome de propriedade e um valor, separados por dois pontos (:)

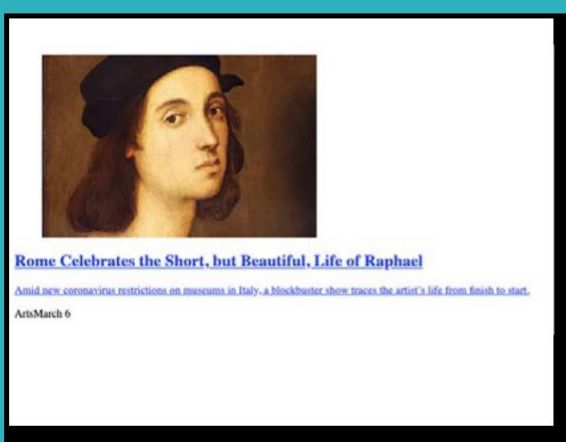
Uma declaração acaba sempre com ponto e vírgula, e blocos de declaração estão delimitados por chavetas.

Esempio:

```
p { color: red; font-size:14px; }
```

@ Vamos começar

Existem três formas de implementar o CSS nas nossas páginas web.



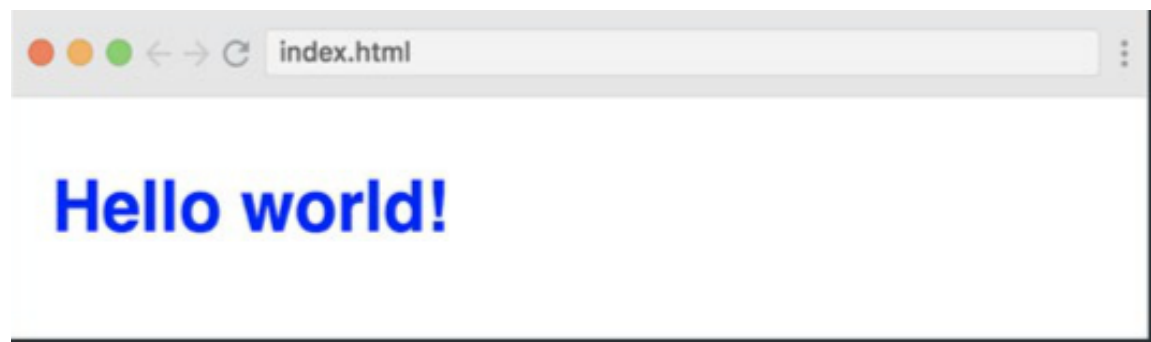
_1. CSS Inline

Em primeiro lugar, é possível incluir CSS diretamente nos nossos elementos HTML. Para isso, fazemos uso do atributo style, e escrevemos código CSS dentro do seu valor (entre as aspas). Por exemplo:

```
<h1 style="color: blue"> Hello world!! </h1>
```

Come si può notare, il CSS incarica il browser di utilizzare il colore blu per la visualizzazione del tag h1. Facile, no? Inoltre, si possono combinare diverse proprietà. Per esempio, rendiamo il testo sia blu che in grassetto:

```
<h1 style="color:blue;font-weight:bold"> Hello world!!</h1>
```



Fácil de novo! Como pode ver, pode adicionar propriedades utilizando um ponto e vírgula (;) para os separar e para que o navegador não fique confuso.

Poderíamos adicionar muitas mais propriedades usando este método, mas o nosso ficheiro HTML pode ficar realmente confuso por isso não é recomendado. Os próximos dois métodos são mais limpos...

_2. O bloco <style>

Outra forma de incluir o CSS é utilizando uma tag <style> dentro da secção head na nossa página HTML.

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>CSS is awesome - BeCode</title>
  <style> h1 {color:blue; }
</style>
</head>
```

Acabámos de encontrar uma solução para não misturar o HTML com o CSS, porém o nosso esquema continua dentro do ficheiro HTML.

Talvez haja uma forma de incluir o CSS? Sim, há!

_3. CSS externo

Tal como o nome indica, teremos alguns ficheiros CSS externos, que importaremos dentro do <head> da nossa página HTML.

```
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge"> <title>The best way! -
BeCode</title>

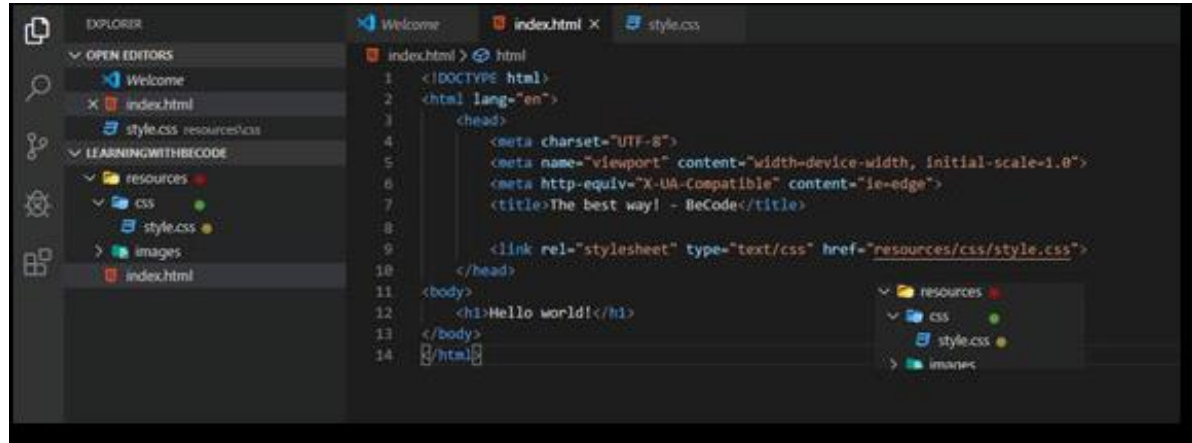
<link rel="stylesheet" type="text/css" href="resources/css/style.css">

</head>
```

Como pode ver, desta vez utilizamos uma tag <link> para criar uma ligação com o nosso ficheiro CSS. Esta tag precisará de alguns atributos para que funcione, **rel="stylesheet"** especifica a relação entre o ficheiro HTML e CSS, o navegador sabe agora que estamos a tentar ligar uma Stylesheet (ficheiro CSS). O **type="text/CSS"** dirá ao navegador que tipo de recurso estamos a ligar. Não é obrigatório o seu uso, mas recomendámo-lo para evitar problemas no futuro. Por último, mas não menos importante, o **href="resources/css/style.css"** é o caminho que o link utilizará para encontrar o documento.

Ter um ficheiro CSS externo é a forma mais recomendável porque “separa responsabilidades”: o ficheiro html é para o conteúdo e o ficheiro CSS é para a decoração.

Aqui está um exemplo da nossa estrutura de pastas:



Dentro do nosso ficheiro CSS, escrevemos o seguinte:

```
h1{color:blue;}
```

Tal terá um resultado semelhante ao que demos no exemplo 1, o benefício está em ter o ficheiro CSS separado do HTML e poder importá-lo para várias páginas ao mesmo tempo!

@ Trabalhar com cores

Cores são uma grande parte do aspeto das coisas. E nós, humanos, adoramos cores!

Há milhões de cores disponíveis na Natureza... Foi, portanto, um grande desafio transferi-las para o mundo digital dos computadores. Primeiro, havia apenas algumas cores disponíveis, utilizando-se nomes predefinidos (como "vermelho", "azul", "bege", "chocolate")...

_ Cores nomeadas

A maneira mais fácil de começar a usar cores é ... pelos seus nomes! Elas são fáceis de lembrar para nós, humanos, especialmente para falantes de inglês, já que estão em inglês. Existem atualmente 140 palavras-chave de cores, incluindo cores primárias e secundárias (como red, blue, ou orange), vários tons de cinza (de **black** a **white**, incluindo cores como **darkgrey** e **lightgrey**), e uma variedade de outras cores combinadas, incluindo **AliceBlue**, **DarkOrchid**, e **rebeccapurple**. A lista completa está disponível em W3Schools: https://www.w3schools.com/colors/colors_names.asp.

É bom saber que existem, mas sentir-se-á rapidamente limitado por ter apenas 140 cores.... Então, vamos passar à próxima forma de expressar valores de cor...

_ O Sistema RGB

Rapidamente os computadores tornaram-se mais poderosos e capazes de manipular milhões de cores diferentes, utilizando uma mistura das 3 cores fundamentais do ecrã: Red (Vermelho), Green (Verde) e Blue (Azul), que é conhecido como o "sistema RGB".

```
p {color: rgb(255, 0, 0) ; }
```

Isto diz "Quero o máximo de vermelho (o máximo é 255), sem verde (0) e sem azul (0)... Criando uma cor vermelha brilhante.

Isto é exatamente o mesmo que

```
p{color: red;}
```

Por isso, se, por exemplo, quiser mesmo aquele tom específico de azul que descreve o céu Escocês na Primavera, necessita de encontrar a sua tradução correta no sistema RGB.

```
p{color: rgb(0, 182, 255);}
```

_ Transparência!

Também pode utilizar um quarto valor, para definir o "alfa", que significa "transparência" (ou "opacidade" se preferir). O seu valor vai de 0 (totalmente transparente, a tag seria invisível) a 1 (totalmente opaca). Em vez de RGB utilizamos RGBA para adicionar a camada transparente à nossa cor.

Digamos que quer um quadrado laranja com 60% de transparência, eis o que deve fazer:

```
div { width: 100px; height: 100px; background-color: rgba(255, 221, 0,0.6); }
```

_ O Sistema Hexadecimal

Para sua informação, há ainda uma outra forma de expressar os valores de cor, utilizando o sistema Hexadecimal. Por exemplo, nesse sistema, red é expresso como

```
#FF0000, black #000000 e white: #FFFFFF.
```

O sistema hexadecimal funciona a partir de valores que vão de 0 a 9 e continua de A até F para um total de 16 valores. Ao utilizar 6 valores Hexadecimais, é possível expressar 256 milhões de cores.

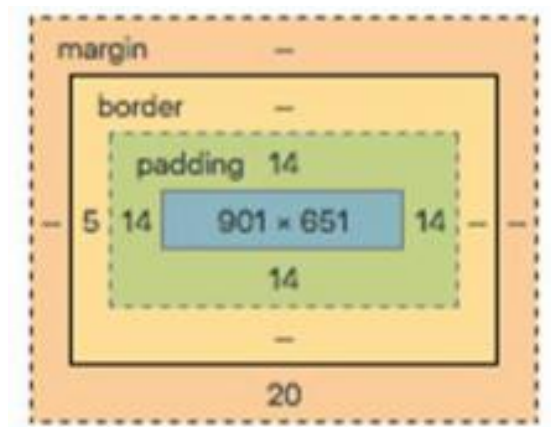
Os primeiros 2 dígitos descrevem o valor de Vermelho, os 2 seguintes o valor de Verde e os 2 últimos o valor de Azul.

Fica a saber que este sistema existe e que poderá utilizá-lo se quiser. Não vamos aprofundar muito o assunto nesta formação.

@ Brincar com os limites (borders)

Por predefinição, sem estilo, cada tag é representado como um retângulo cujo fundo e limites são transparentes. Mas não tem de ficar assim!

Abaixo encontra-se uma representação visual de um retângulo, chamado "modelo de caixa".



Esta imagem representa como é possível brincar com a border, margin, padding para modelar qualquer tag HTML!

Para explicar isto um pouco melhor, vamos meter as mãos na massa!

Crie um ficheiro HTML e copie estas linhas para o corpo:

```
<div class="box1"> <div class="box2"> </div> </div> <div class="box3"> </div>
```

Em seguida, crie um ficheiro CSS e copie as seguintes linhas para o ficheiro:

```
.box1{ width:200px;
height:200px;
border-top:1px solid red;
border-right: 1px solid black;
border-bottom: 2px dotted green;
border-left: 2px dashed green;
padding:100px;
padding-right:50px;
background-color: yellow;

/*--We will cover this later--*/
display:inline-block;
/*-----*/}

.box2{ width:200px;
height:200px;
background-color:red;

/*--We will cover this later--*/
display:inline-block; /*-----*/}
```

```
.box3{
width:100px;
height:100px;
background-color:green;
margin-left:200px;
/*--We will cover this later--*/
display:inline-block;
/*-----*/}
```

Não vai parecer muito bonito, mas não nos vamos preocupar com isso agora.

Como pode ver, pode especificar cada limite do retângulo utilizando 3 parâmetros: a **espessura** da linha (aqui, em pixéis), o **tipo** de linha (solid, dashed, dotted), e a sua **cor**. Já que aqui estamos, tente descobrir a diferença entre padding e margin. Brinque com os seus valores e mais tarde vamos discutir isto em grupo.

Os limites podem ser utilizados para transformar retângulos em quadrados!

Ao longo da aprendizagem, verá que o CSS está cheio de truques. Um truque realmente útil é que pode tornar uma imagem como esta::



image: The Duffer Brothers. Fonte: <https://www.strangerthings.fr>

Esta é a propriedade que o torna possível::

```
border-radius:50%;
```

Terá a oportunidade de a experimentar nos exercícios....

@ Comentários em CSS

Comentários⁵ são utilizados para explicar o código, e podem ajudar quando posteriormente editar o código de origem. Os comentários são ignorados pelos navegadores.

Um comentário CSS começa com `/*` e termina com `*/`:

`/* This is a single-line comment */`

`p { color: red; }`

Pode adicionar comentários onde quiser no código:

`p { color: red;`

`/* Set text color to red */}`

Os comentários podem também abranger várias linhas:

`/* This is a multi-line comment */`

`p { color: red; }`

@ Parabéns!

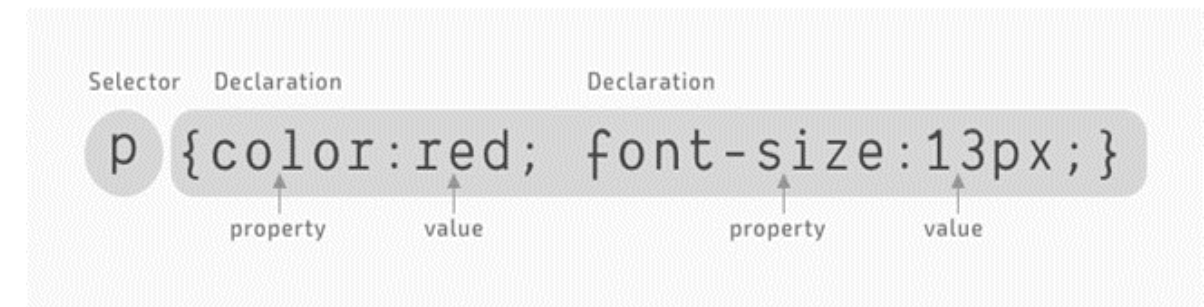
Acabou de descobrir a magia do CSS. Agora vamos colocar os nossos conhecimentos à prova fazendo alguns exercícios!

Seletores

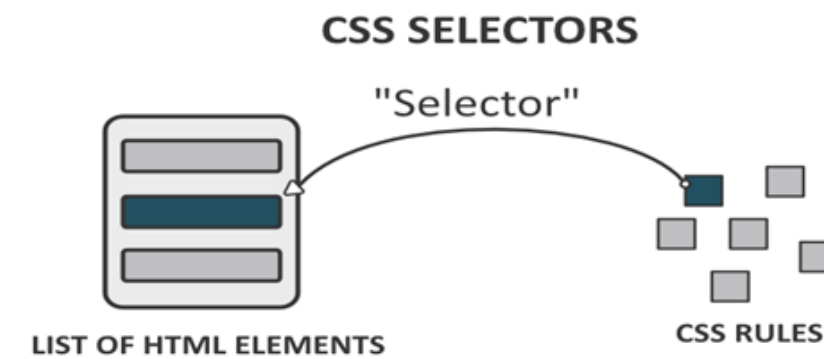
@ Seletores CSS

Bem-vindo a uma parte muito importante do CSS, os seletores.

Lembra-se deste visual da introdução?



Um seletor CSS é a parte de um conjunto de regras CSS que de facto seleciona o conteúdo que pretende modelar. Os seletores são o que permite ao navegador saber que elemento(s) html devem ser pintados e com que propriedades do CSS.



Por exemplo, tenho este trecho de HTML:

```
<div class="becode">
<p> I am on my way to become a web-developer!</p>
</div> <div id="white-background"> </div>
```

Se quiser ter um fundo azul no primeiro "div", posso fazer isto no CSS:

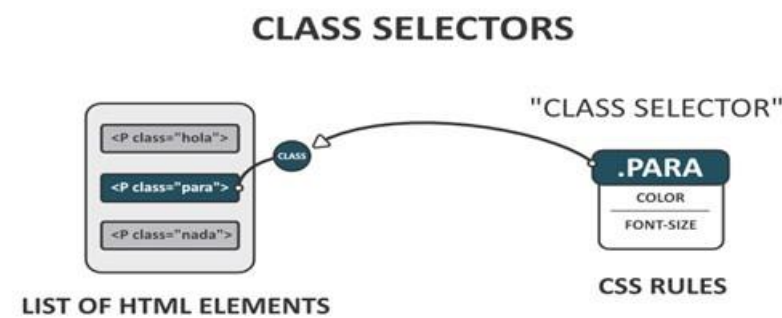
```
div{height:100px;
width:100px;
background-color:blue;}
```

⁵ Esta secção usa informações adaptadas de: https://www.w3schools.com/css/css_syntax.asp

No entanto, se o fizer, vai notar que ambos os elementos div terão um fundo azul. É por isso que há mais seletores do que apenas o seletor div! Por exemplo, temos escrito a classe 'becode', por que não utilizá-la no nosso ficheiro CSS? Podemos seleccionar elementos de classe como este:

```
.becode{height:100px;
width:100px;
background-color:blue;}
```

Agora apenas o fundo do nosso primeiro div vai ser alterado, tal como queríamos! O que fez aqui, colega, foi utilizar um seletor de classe.



O Seletor de Classe é indicado por um ponto (.) imediatamente seguido pelo nome da classe que pretende utilizar. Pode ser qualquer coisa, desde que seja o mesmo que no ficheiro HTML.

Portanto, se no html, o nome da classe for "burger", o seletor da classe CSS seráburger. Se o nome da classe for "ferrari", o seletor da classe CSS torna-seferrari. Percebido?

Existem muitas formas de seleccionar elementos html utilizando os seletores CSS. Demasiados para abordar neste curso. Saiba que os seletores de classe são os mais utilizados e permitir-lhe-ão seleccionar tudo o que precisa, especialmente se também souber utilizar...

_ Seletores de pseudo-classes

A maioria dos elementos como H1, P ... não são para ser interativos. Mas alguns são, como os links que em HTML são chamados Âncoras e são codificados utilizando a tag A. Analise este CSS, estilizando todas as âncoras na sua página para que se pareçam com o logótipo do IKEA.

```
a.ikea {text-decoration:none; /* remove underline */
background-color: blue; /* paint the background blue */ color:yellow; /*
paint the text yellow */ font-weight:bold; /* make the characters bold */
text-transform: uppercase; /* turn characters in UPPERCASE */
padding:5px 10px; /* add some padding */ font-size:23px; /* make the text
bigger */}
```

Então agora, no seu html, pode adicionar a classe "ikea" a todos os links que quer que pareçam comprados no IKEA :-)

```
<p> I loooooove design, especially furniture from <a href="http:// aeki.com"
class="ikea">Aeki</a>. It's cheap and stylish... Okay, if you actually manage to
build them!</p>
```

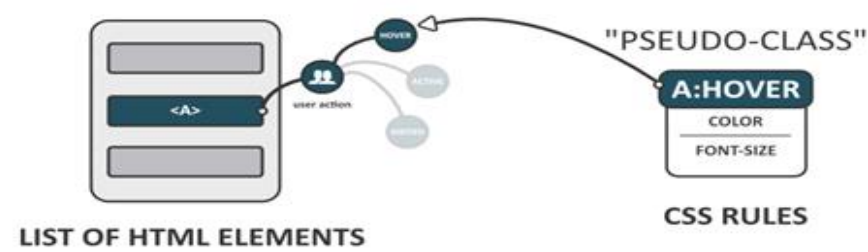
Resultado:

I loooooove design, especially furniture from **AEKI** . It's cheap and stylish... Okay, if you actually manage to build them!!

Boa, mas se passar com o rato em cima dele, nada acontece. Triste, para um elemento interativo!

É aqui que entra a pseudo-classe...

PSUEDO-CLASS SELECTORS



As "pseudo-classes" CSS fornecem um mecanismo para se ligar a esta espécie de informação temporária do utilizador. Num dado momento, um elemento `<a href>` pode estar em vários estados diferentes, e é possível utilizar pseudo-classes para modelar cada um deles individualmente. Pense neles como seletores de classe que não precisa de escrever pois já estão integrados no navegador..

_ Estilos básicos de link

Oliver James (InternetingIsHard.com) resume ⁶ a sintaxe da seguinte forma:

“” As pseudo-classes começam com dois pontos seguidos pelo nome da classe desejada. As pseudo-classes de link mais comuns são as seguintes:

- **:link** – um link que o utilizador nunca visitou.
- **:visited** – Um link que o utilizador já visitou anteriormente.
- **:hover** – Um link com o rato do utilizador por cima.
- **:active** – Um link que está a ser pressionado por um rato (ou dedo)”

E é tudo sobre os seletores de CSS. Já tem informação suficiente para fazer os exercícios! Não hesite em voltar aqui se não conseguir concluir os exercícios. A prática vai ajudá-lo a lembrar-se ;-)

Propriedades básicas

Sabemos como nomear os nossos elementos em CSS, utilizámos algumas propriedades básicas, mas há muitas mais propriedades disponíveis para si, Picasso! Nesta secção aprenderá a utilizar mais algumas propriedades interessantes através de exercícios curtos.

Posicionamento

Grande parte do seu trabalho é colocar elementos no ecrã e controlar como devem adaptar-se ao conteúdo adjacente, bem como ao tamanho do ecrã (telemóvel ou computador). Numa palavra: posicionamento

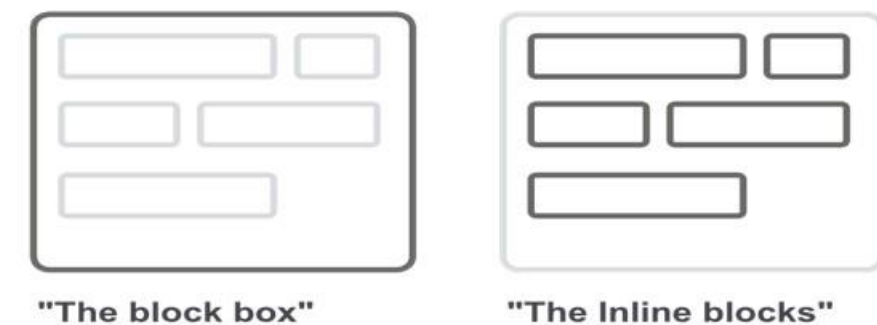
O CSS oferece várias formas de controlar o posicionamento dos elementos no ecrã. Vamos começar por uma compreensão de como o elemento em si é estabelecido.

@ Caixas e blocos

Pode imaginar cada elemento HTML como uma caixa, temos 2 versões diferentes dessas chamadas "caixas".

- Uma caixa "bloco"
- Uma caixa „inline“

Veja esta representação visual da diferença entre um bloco e um elemento inline



Cada elemento HTML tem um tipo de caixa padrão, você verá que nem todos os elementos compartilham o mesmo tipo de caixa no final! Por exemplo, `<div>`, `<h1>`, `<p>`, ``, ... são elementos de bloco, enquanto ``, `<a>`, ... são elementos inline.

Essas chamadas "caixas" são importantes para o fluxo de nossa página.

⁶ Fonte: <https://www.internetingishard.com/html-and-css/css-selectors/#pseudo-classes-for-links>

_ Elementos bloco

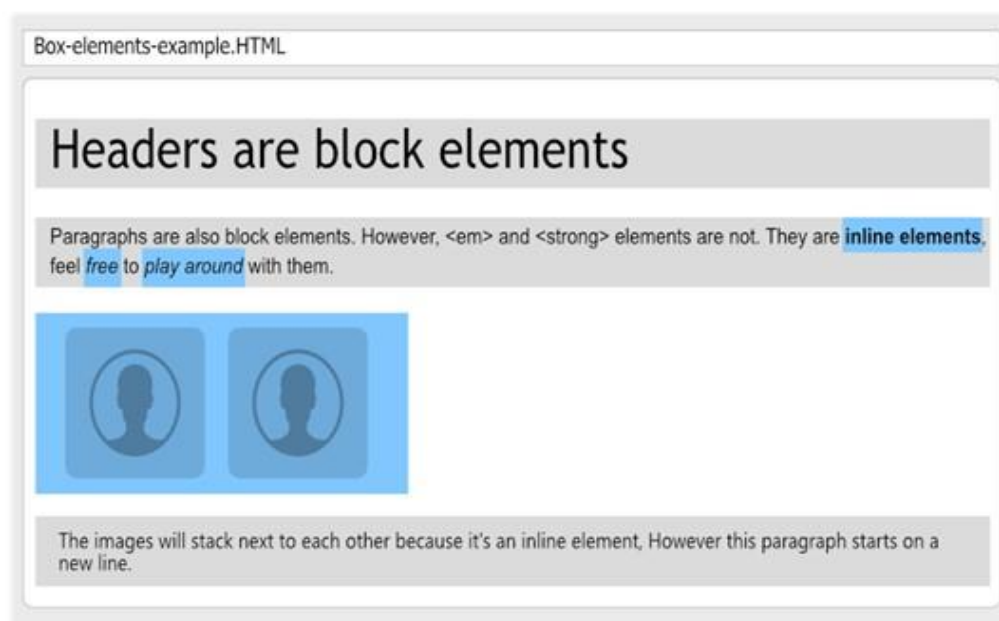
Um elemento de bloco geralmente ocupa toda a largura do “container” pai. Como resultado, ele forçará o próximo elemento de bloco em uma nova linha, cada elemento de bloco aparecerá abaixo do outro.

A altura de um elemento de bloco sempre será decidida pelo seu conteúdo - se o texto não couber na tela, o seu comportamento padrão é adicionar o texto numa nova linha

_ Elementos inline

O comportamento dos elementos inline é muito diferente do dos elementos de bloco; eles são utilizados para definir o estilo de itens dentro de elementos de bloco. Ao contrário do elemento Bloco, que assume a largura total do pai, a largura do elemento Inline é sempre baseada no seu conteúdo.

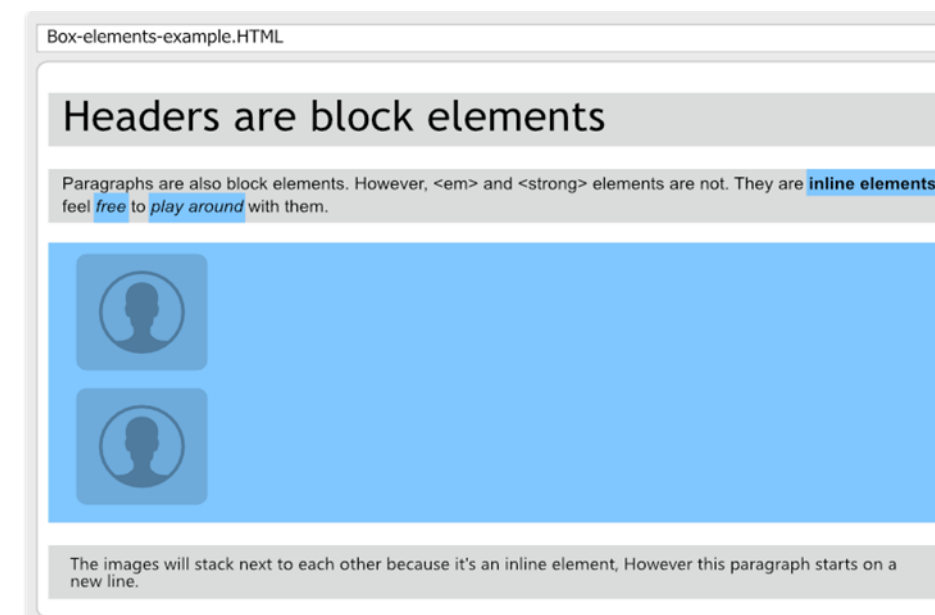
Espreite o visual abaixo - verá que as partes cinza (elementos de bloco) ocupam toda a largura do corpo. Enquanto as partes azuis (elementos embutidos) ocupam apenas a largura de seu conteúdo.



_ Mudar o comportamento da caixa

Em CSS, estamos totalmente no controle de como nossos elementos devem ser exibidos, o que significa que podemos alterar o comportamento padrão de nossos elementos também! Talvez queira que as suas imagens sejam empilhadas uma abaixo da outra, o que pode ser feito facilmente usando o atributo display:

```
img {
  display: block;
}
```



@ O Modelo de Caixa CSS

Há uma coisa que os elementos block e inline têm em comum - e isso é o CSS Box Model. O que é que está a dizer? Outra caixa? Sim, bem, não exatamente - ainda é a mesma caixa de que estávamos a falar antes. Vamos espreitar o que está incluído nesta chamada "caixa".

Embora possa parecer complicado, na verdade é fácil de entender. Este modelo de caixa é usado para calcular quanto espaço existe entre / dentro de cada elemento ou, se necessário, para adicionar uma borda ao redor do elemento.

Vamos olhar para o modelo de caixa visualmente



1. Content (conteúdo) – o texto, imagem ou outro conteúdo de mídia no elemento.
2. Padding (espaçamento) – o espaço entre o conteúdo da caixa e os seus limites.
3. Border (limite) – a linha entre o espaçamento da caixa e a margem.
4. Margin (margem) – o espaço entre a caixa e as caixas circundantes

Preenchimento

Já vimos a parte do conteúdo, então vamos saltar um nível para o preenchimento. Esta propriedade decidirá quanto preenchimento haverá para o elemento selecionado:

Aqui temos um exemplo:

This is a paragraph without padding.

```
p { padding: 10px;}
```

This is a paragraph with padding.

Como é possível ver, o fundo azul atrás do elemento p aumentou. Isto porque dissemos ao nosso CSS para aumentar o preenchimento em 10px! Agora vê que está a aumentar 10 px para todos os lados, porém às vezes queremos apenas estilizar um lado ou ter valores diferentes em todos os lugares.

Isso pode ser facilmente feito da seguinte forma:

```
p {
padding-top: 20px;
padding-bottom: 20px;
padding-left: 10px;
padding-right: 10px;
}
```

This is a paragraph with padding.

Consegue ver a ligeira diferença entre o preenchimento superior e inferior em comparação com o lado esquerdo.

O preenchimento funciona tanto em elementos de bloco como inline.

Limites

Agora, a próxima propriedade do Modelo Caixa é a Borda. Por norma, a maioria dos elementos possui um limite transparente. Se gostava de atribuir uma borda ao seu elemento, terá de seguir a seguinte sintaxe CSS: Especifique a largura da borda, depois que tipo de borda ela deve ter e por último a sua cor.

Vamos adicionar uma borda ao nosso exemplo de preenchimento anterior!

```
p {
border: 4px solid #5d6063;
}
```

This is a paragraph with padding and a border!

Tal como o padding, existem as variants -top, - bottom, -left, e -right para a propriedade border:

```
p {
border: 1px solid #5D6063;
}
```

Isto simplesmente desenharia uma linha por baixo do elemento.

Limites funcionam tanto em elementos de bloco como inline.

Margens

Chegamos à nossa última propriedade do Modelo Caixa, as margens. As margens são os espaços transparentes ao redor de um elemento, portanto, não o confunda com preenchimento. O preenchimento cuida do espaçamento dentro de um elemento onde a margem afasta outros elementos ao redor do elemento selecionado.

```
p {
padding: 20px 10px 20px 10px;
margin-bottom: 20px;
}
```



This is a paragraph with padding, a border and margin!

E, assim como o preenchimento e as bordas, pode definir um ou mais lados para fazer alterações.

As margens funcionam apenas em elementos de bloco.

_ Alterar larguras ou alturas

Até agora, deixamos o navegador decidir a largura que nossos elementos devem ter, mas às vezes você mesmo deseja definir a largura. Talvez uma imagem seja muito grande ou um texto que deva cobrir apenas 20% da tela. Para isso, podemos usar as propriedades de largura e altura dentro do CSS. Isso mudará a largura padrão do conteúdo da sua caixa.

Vamos dar ao nosso parágrafo uma largura de 200 px.

```
button {
  width: 200px;
}
```



This is a paragraph with width.

Em vez de ser tão largo como a janela do navegador, o nosso botão é agora de 200 pixels, cingindo-se ao lado esquerdo da página.

A largura / altura funcionam em elementos de bloco ou blocos inline.

_ Centrar com auto-margens

Para evitar que nosso elemento se restrinja à esquerda, podemos utilizar o valor de margens automáticas dentro da nossa propriedade de margem. Isso criará automaticamente espaço ao redor do elemento de bloco e irá dividi-lo igualmente nos lados esquerdo e direito.

Podemos centralizar o nosso parágrafo da seguinte maneira:

```
p {
  margin: 20px auto;
}
```



This is a paragraph with width.

Isso só funcionará quando uma largura for definida.

@ Resumo

- Tudo dentro de uma página HTML é uma caixa.
- Caixas podem ser inline ou de bloco.
- Caixas têm conteúdo, espaçamento, limites e margens.
- Uma vez dominado o Modelo Caixa, consegue estruturar a maioria das páginas web.

Como no último capítulo, as propriedades CSS que acabamos de cobrir podem parecer simples – e de certa forma são. Mas, comece a olhar para os sites que visita através da lente de “modelo de caixa CSS”, e verá estas coisas literalmente em todo o lado.

Muito bem, vamos em frente e continuar a aprendizagem criando alguns desafios de posicionamento através de 3 técnicas: Display, Posição e Flexbox!

Transições & animações CSS

Estamos quase a terminar a parte do CSS, muito bem! Mas há algo que ainda não abordamos. E se quiser movimentar os elementos? Ou se quiser animar o seu botão como no exemplo abaixo:



Bom, tudo isto é possível com Transições e animações!

@ Transições

Sem transições, todas as alterações num elemento CSS acontecem instantaneamente. Vamos imaginar que utilizamos uma pseudo classe :hover. Normalmente, os efeitos de foco acontecem instantaneamente, assim que o rato passa sobre o elemento. Mas quando adicionamos uma transição, criamos uma transição suave entre os dois estados do elemento.

Pode utilizar transições com larguras, alturas, transformações...

_ Como utilizar uma transição em CSS?

Veja o seguinte trecho de CSS:

```
button {
  color: red;
  transition: color 0.5s ease-in-out 1s;
  /*transition: (property) (duration) (transition-timing-function) (delay)*/
}
button:hover{
  color:blue;
}
```

Se usássemos este CSS, resultaria num botão com a cor de fonte vermelha, mas quando o passarmos, ele aguardará 1 segundo, e depois mudará suavemente a cor vermelha para azul num período de 0,5 segundos.

Vamos examinar os diferentes valores

- Propriedade: O que quer animar? Largura, altura, cor, ...
- Duração: Quanto tempo deve durar a transição? 0,5 segundos, 1 segundo, ...
- Transição-timing-função: Qual é a velocidade da transição (ver abaixo mais informações)
- Delay: Quanto "atraso" deve haver quando a transição for acionada

Poderá utilizar transições sem utilizar todas as propriedades; se não atribuir nenhum valor, ele utilizará o valor padrão.

Estes são os valores padrão:

transição: todos 0 facilidade 0.

Como pode ver, para que a sua transição funcione, pode deixar todos os valores vazios, exceto o valor da duração, caso contrário, não haverá uma transição suave.

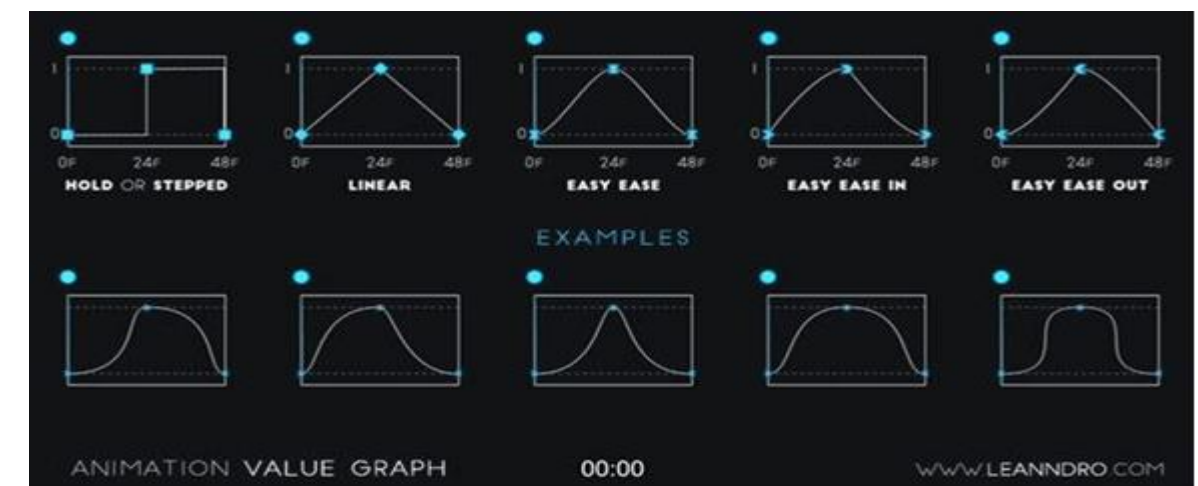
Nota: todos os valores animarão todas as propriedades diferentes se forem alteradas. Embora seja fácil animar tudo rapidamente, pode levar a comportamentos estranhos e indesejados.

_ Tempo de transição

Por vezes, as animações parecem enfadonhas e sem vida, se desejar fazer alterações, provavelmente vai querer espreitar a função de tempo de transição.

Esta função decide o "ritmo" da sua animação. Começa lento ou rápido? Termina lentamente? Isto não é apenas melhor para os olhos, mas também torna a animação mais natural.

Veja a seguinte folha de valores de animação:



Leandro www.leanndro.com – Gráfico de valor da animação - Timing

Fonte: <http://spungella.blogspot.com/2016/03/animation-value-graph-timing.html>

- A parte superior mostra a curva e a animação por trás de cada valor.
- A inferior mostra animações de quando manipularíamos esta curva.

É incrível como podemos mudar a sensação mudando apenas uma curva, não é?

Sinta-se à vontade para experimentar outras easing functions (procure-as na Documentação – lembre-se que a procura faz parte do trabalho de um programador profissional ;-))

@ Animações

Poderá pensar que uma animação é o mesmo que uma transição: ambas alteram os valores das propriedades. Porém, embora a transição possa mudar apenas em dois estados, uma animação pode fazer a transição em quantos estados desejar.

Para animações CSS, deve utilizar "keyframes", que contém a propriedade de um elemento e, claro, o valor para o qual ele deve ser alterado. Poderá utilizar quantos keyframes quiser para fazer animações longas e realmente suaves

_ Como utilizar keyframes?

Em primeiro lugar, temos que projetar a animação dos keyframes, e isso pode ser conseguido simplesmente criando uma regra css **@keyframes**, em seguida dê um nome para que possamos chamá-la mais tarde nos nossos elementos.

A próxima etapa é definir de e para onde a nossa animação deve ir. Veja no exemplo a seguir:

- Que começamos com um tamanho de fonte de 50 px e a cor vermelha.
- E que vamos para um tamanho de fonte de 100 px e a cor azul.

```
/* The animation code */
from {
  font-size: 50px;
  color: red;
}
to {
  font-size: 100px;
  color: blue;
}
}
```

```
/* The element to apply the animation to */
p {
  color: red;
  animation: test 4s infinite alternate ease-in-out;
}
```

Tal resultaria em algo como isto:

This is a simple animation!

Como pode ver, utilizamos **"from" (0%)** e **"to" (100%)** - que são apenas 2 estados ("de" e "para"). Mas, como disse antes, é possível animar muitos mais estados.

Isto pode ser obtido utilizando percentagens. Dê uma vista de olhos no seguinte exemplo:

```
/* The animation code */
@keyframes test {
  0% {
    font-size: 0px;
    color: red;
  }
  25% {
    font-size: 100px;
    color: pink;
  }
  50% {
    font-size: 30px;
    color: green;
  }
  75% {
    font-size: 120px;
    color: yellow;
  }
  100% {
    font-size: 0px;
    color: blue;
  }
}

/* The element to apply the animation to */
p {
  color: red;
  animation: test 4s infinite alternate ease-in-out;
}
```

Consegue ver o resultado em:

<https://github.com/becodeorg/free2code/blob/master/2.CSS/4.CSS-ANIMATIONS/exercise-09/exercise.md>

Tal como acontece com as transições, terá que atribuir um valor de duração, caso contrário, nenhuma animação será reproduzida.

Conclusão

Animações são muito divertidas, não é? Poderia dizer-lhe muito mais acerca de transições e animações, mas não é esse o objetivo aqui! Vamos mergulhar na documentação e pôr as mãos na massa! Sinta-se à vontade para experimentar, destruir e criar coisas!



Fonte: <https://tenor.com/view/monsters-inc-typing-group-chat-is-lit-gif-11492598>

Referências bibliográficas (como identificadas no texto)

1. Retrato de Tim Berners-Lee - © Paul Clarke, CC BY-SA 4.0, via Wikimedia Commons
Fonte: https://en.wikipedia.org/wiki/Tim_Berners-Lee
2. Captura ecrã do primeiro navegador desenvolvido por Sir Tim Berners-Lee - Autor: Tim Berners-Lee (domínio público) - Fonte: <https://en.wikipedia.org/wiki/WorldWideWeb>
3. How does the Internet work? Por Mozilla Contributors licença sob CC-BY-SA 2.5. Fonte: https://developer.mozilla.org/en-US/docs/Learn/Common_questions/How_does_the_Internet_work
4. HTML Basics por Mozilla Contributors licença sob CC-BY-SA 2.5. Fonte: https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/HTML_basics
5. Using HTML sections and outlines por Mozilla Contributors, licença sob CC-BY-SA 2.5. Fonte: https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Using_HTML_sections_and_outlines%23section_elements_in_html5
6. <https://learnabouthtml5.blogspot.com/2016/12/using-html-sections-and-outlines.html>
7. Roma comemora o curto mas belo arquivo de Rafael. Fonte: <https://www.nytimes.com/2020/03/06/arts/raphael-rome-coronavirus.html>
8. Lista completa de cores disponíveis em W3Schools: https://www.w3schools.com/colors/colors_names.asp
9. Os irmãos Duffer. Fonte: <https://www.strangerthings.fr>
10. Comentários sobre CSS, adaptado de: https://www.w3schools.com/css/css_syntax.asp
11. Oliver James sobre Pseudo-classes. Fonte: <https://www.internetingishard.com/html-and-css/css-selectors/#pseudo-classes-for-links>
12. Gráfico de valor de animação - Timing - Leandro [www.leanndro.com](http://spungella.blogspot.com/2016/03/animation-value-graph-timing.html) - Fonte: <http://spungella.blogspot.com/2016/03/animation-value-graph-timing.html>
13. GIF Monster Inc a digitar . Fonte: <https://tenor.com/view/monsters-inc-typing-group-chat-is-lit-gif-11492598>

Anexos

Plano Sessão – Módulo 0

Free to Code – Melhorar as capacidades digitais e de programação de reclusos

Módulo: 0 – Introdução geral		Formador:
Sessão nº:0 (metade de uma sessão)	Duração da sessão: 1h	Data:
Objetivos gerais:	Os formandos serão capazes de modificar uma página da web existente e criar páginas da web básicas e utilizar HTML e CSS com um editor de código. Os formandos irão adquirir conhecimentos básicos sobre como descrever uma solução para problemas simples de desenvolvimento web, em etapas lógicas, e entender a utilidade de cada tecnologia de front-end (html, css e Javascript).	
Resultados de aprendizagem:	<ul style="list-style-type: none"> • Descrever uma solução para problemas simples em etapas lógicas • Terminologia básica de tecnologias de desenvolvimento web front-end • Ser capaz de utilizar a interface de programação fornecida • Algorítmica e resolução de problemas 	
Métodos	Expositivo, interrogativo e ativo. Aprendizagem baseada em computador e em técnicas de aprendizagem ativa, tais como aprendizagem baseada em projetos, questionários e aprender-fazendo, sob a orientação de um formador/técnico - aprendizagem combinada.	
Conteúdos	Contexto da internet <ul style="list-style-type: none"> • Internet • Web 	

Hora início	Atividades:	Recursos e materiais:
10m	Atividade quebra-gelo Actividade bola de lã - coloque os formandos e formador(es) em círculo e peça-lhes que se apresentem (nome, idade, de onde são) e depois passe a bola a quem quiserem. O objetivo desta atividade é, no final, dizer que iniciaram uma rede de contatos neste momento e que devem se manter interligados para se ajudarem agora e no futuro – a formação visa ser de autoaprendizagem e os formandos devem ajudar-se uns aos outros para ter sucesso. Esta atividade também serve como ponto de partida para o próximo brainstorming - analogia entre a rede que foi criada com a bola de lã e a tecnologia para iniciar as questões do brainstorming (veja abaixo).	Bola de lã e espaço na sala para criar um círculo com todos os participantes
10m	Brainstorming sobre Tecnologia para abrir a discussão e apresentar o tópico Sugestões de perguntas: <ul style="list-style-type: none"> • “O que lhe vem à mente quando pensa em tecnologia?” • “Facilita nossas vidas?” • “O que é um site? E um aplicativo? ” • “Onde está a codificação no mundo da tecnologia?” • Outras perguntas que possa achar relevantes 	<ul style="list-style-type: none"> • Flipchart e canetas ou • Documento Word • Etc.
40m	Apresentação dos tópicos principais abordados na parte de aprendizagem do módulo 0: <ol style="list-style-type: none"> 1. Contexto da internet <ol style="list-style-type: none"> a. A Internet b. Como funciona, a rede de redes e web Conclusão, discussão e perguntas <i>(1ª sessão a ser continuada nesta na próxima tabela word)</i>	Plataforma de aprendizagem
Métodos de avaliação:	Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.	
Atividade (Tempo):	Lições 1-3 (1h)	

Plano Sessão –Módulo 1

Training: Free to Code – Improving Digital and Coding Skills for Inmates

Módulo: 1 – HTML

Formador:

Sessão nr. : 0 a 2

(2 sessões e meia)

Duração: 8h

Data:

Objetivos gerais:

Os formandos serão capazes de modificar uma página da web existente e criar páginas da web básicas e utilizar HTML e CSS com um editor de código.

Os formandos irão adquirir conhecimentos básicos sobre como descrever uma solução para problemas simples de desenvolvimento web, em etapas lógicas, e entender a utilidade de cada tecnologia de front-end (html, css e Javascript).

Resultados de aprendizagem:

- Descrever uma solução para problemas simples em etapas lógicas
- Terminologia básica de tecnologias de desenvolvimento web front-end
- Ser capaz de utilizar a interface de programação fornecida
- Algorítmica e resolução de problemas

Expositivo, interrogativo e ativo.

Métodos:

Aprendizagem baseada em computador e em técnicas de aprendizagem ativa, tais como aprendizagem baseada em projetos, questionários e aprender-fazendo, sob a orientação de um formador/técnico - aprendizagem combinada.

Conteúdos:

- Introdução
- HTML5
- Semântica

Hora início	Atividades:	Recursos e materiais
Sessão 0 (continuação)		
1h	<p>A formação do 1º módulo será dividida em duas partes - uma parte de apresentação e uma parte de exercícios.</p> <p>Apresentação dos 3 tópicos principais abordados na parte de aprendizagem do módulo 1:</p> <ol style="list-style-type: none"> 1. Introdução <ol style="list-style-type: none"> a. Por que e o que é HTML 2. HTML5 <ol style="list-style-type: none"> a. Primeiro documento HTML 3. Semântica <ol style="list-style-type: none"> a. Porquê utilizar semântica, HTML semântico 	<ul style="list-style-type: none"> • Plataforma – parte conteúdos
<p>Após a apresentação os formandos devem fazer os exercícios designados para este 1º módulo de formação.</p>		
1h	<p>Os exercícios devem ser realizados pelos formandos, mas sempre com o apoio do(s) formador(es), e com acesso contínuo à parte dos conteúdos da plataforma para que possam rever conteúdos que os auxiliem na realização dos exercícios.</p> <p><i>Exercícios:</i></p> <ul style="list-style-type: none"> • Exercício 1 - Olá, mundo (15 minutos) • Exercício 2 - Títulos (15 minutos) • Exercício 3 - Parágrafos (15 minutos) • Exercício 4 – Negrito Cursivo (15 minutos) 	<ul style="list-style-type: none"> • Plataforma – parte exercícios
<p><i>Após as 3 horas, fecha-se a primeira sessão com conclusão, discussão e perguntas.</i></p>		
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma e Desafio Final) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”)</p> <p>Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	

Hora início	Atividades:	Recursos e materiais
3h	<p>Sessão 1 Revisão dos conteúdos de aprendizagem discutidos antes e continuação do trabalho nos exercícios.</p> <p>Os exercícios devem ser realizados pelos formandos, mas sempre com o apoio do(s) formador(es), e com acesso contínuo à parte dos conteúdos da plataforma para que possam rever conteúdos que os auxiliem na realização dos exercícios.</p> <p>Exercícios:</p> <ul style="list-style-type: none"> • Exercício 5 - Listas (30 minutos) • Exercício 6 - Links relativos (30 minutos) • Exercício 7 - Imagens (30 minutos) • Exercício 8 - A carta (30 minutos) • Exercício 9 – Objetos de estudo (30 minutos) • Exercício 10 – Passos de dança exóticos (30 minutos) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	Plataforma – parte exercícios & conteúdos
Avaliação	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma e Desafio Final) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”) Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão</p>	
3h	<p>Sessão 2</p> <p>Exercícios:</p> <ul style="list-style-type: none"> • Exercício 11 - Formatação de receita (1h) • Exercício 12 - Agricultor chinês (2h) <p>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</p>	• Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma e Desafio Final) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”) Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	
Hora início:	<ul style="list-style-type: none"> • Exercício 1 - Olá, mundo (15 minutos) • Exercício 2 - Títulos (15 minutos) • Exercício 3 - Parágrafos (15 minutos) • Exercício 4 – Negrito Cursivo (15 minutos) • Exercício 5 - Listas (30 minutos) • Exercício 6 - Links relativos (30 minutos) • Exercício 7 - Imagens (30 minutos) • Exercício 8 - A carta (30 minutos) • Exercício 9 – Objetos de estudo (30 minutos) • Exercício 10 – Passos de dança exóticos (30 minutos) • Exercício 11 - Formatação de receita (1h) • Exercício 12 - Agricultor chinês (2h) 	
Lições 1-3		

Plano Sessão –Módulo 1

Training: Free to Code – Improving Digital and Coding Skills for Inmates

Módulo: 2 – CSS	Formador:
Sessão nº 3 a 18	Duração da sessão: 43h Data:
Objetivos gerais:	<p>Os formandos serão capazes de modificar uma página da web existente e criar páginas da web básicas e utilizar HTML e CSS com um editor de código.</p> <p>Os formandos irão adquirir conhecimentos básicos sobre como descrever uma solução para problemas simples de desenvolvimento web, em etapas lógicas, e entender a utilidade de cada tecnologia de front-end (html, css e Javascript).</p>
Obiettivi di Apprendimento:	<ul style="list-style-type: none"> • Descrever uma solução para problemas simples em etapas lógicas • Terminologia básica de tecnologias de desenvolvimento web front-end • Ser capaz de utilizar a interface de programação fornecida • Algorítmica e resolução de problemas
Métodos	Expositivo, interrogativo e ativo. Aprendizagem baseada em computador e em técnicas de aprendizagem ativa, tais como aprendizagem baseada em projetos, questionários e aprender-fazendo, sob a orientação de um formador/técnico - aprendizagem combinada.
Conteúdos:	<ul style="list-style-type: none"> • Introdução ao CSS • Seletores • Propriedades básicas • Posicionamento • Animações CSS • Desafio Final

Hora início	Atividades:	Recursos e materiais
Sessão 3		
2h	<p>A formação será dividida em duas partes - uma parte de apresentação de conteúdos e uma parte de exercícios. Tópicos abordados na parte dos conteúdos da Plataforma:</p> <ol style="list-style-type: none"> 1. Introdução ao CSS <ol style="list-style-type: none"> a. O que é, sintaxe, introdução, trabalho com cores, limites, comentários 2. Seletores <ol style="list-style-type: none"> a. Seletores CSS, seletores de classe, seletores de pseudoclasse 3. Propriedades básicas 4. Posicionamento <ol style="list-style-type: none"> a. O modelo da caixa, preenchimento, limites, margens, dimensões 5. Animações CSS Transições, animações 	<ul style="list-style-type: none"> • Plataforma – parte exercícios
Sessão 4		
1h	<p><i>Exercícios:</i></p> <ul style="list-style-type: none"> • Exercício 1 - Adicionar algum CSS inline (30 minutos) • Exercício 2 – Planear suas férias com estilo (30 minutos) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> • Plataforma – parte exercícios
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”) Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão</p>	
Sessão 5		
3h	<p><i>Exercícios:</i></p> <ul style="list-style-type: none"> • Exercício 3 - Ilhas Canárias (30 minutos) • Exercício 4 - Aulas para diferenciar elementos e estilizá-los de maneira diferente (30 minutos) • Exercício 5 - Fronteiras (30 minutos) • Exercício 6 - Seletor de classe (1h) • Exercício 7 - Seletor de pseudo-classe (30 minutos) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> • Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”) Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão</p>	

Hora início	Atividades:	Recursos e materiais
Sessão 6		
3h	<p><i>Exercícios:</i></p> <ul style="list-style-type: none"> • Exercício 8 - Defina o estilo desta página (30 minutos) • Exercício 9 - Propriedades Parágrafos (30 minutos) • Exercício 10 – CSS Diner (30 minutos) • Exercício 11 - Propriedades básicas (30 minutos) • Exercício 12 - Propriedades de exibição, exercício 1 (1h) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> • Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”) Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão</p>	
Sessão 7		
3h	<p><i>Exercícios:</i></p> <ul style="list-style-type: none"> • Exercício 13 - Propriedades de exibição, exercício 2 (1h) • Exercício 14 - Propriedades da posição, exercício 1 (2h) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas.</i></p>	<ul style="list-style-type: none"> • Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”) Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	
Sessão 8		
3h	<p><i>Exercícios:</i></p> <ul style="list-style-type: none"> • Exercício 15 - Propriedades de posição, exercício 2 (1h) • Exercício 16 - Propriedades da posição, exercício 3 (1h) • Exercício 17 - Flex-box 1 (1h) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> • Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”) Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	

Hora início	Atividades	Recursos e materiais:
Sessão 9		
3h	<p>Exercícios:</p> <ul style="list-style-type: none"> • Exercício 18 - Flexão do cartão (2h) • Exercício 19 - Flexionando a página (1 de 3h) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> • Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”)</p> <p>Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	
Sessão 10		
3h	<p>Exercícios:</p> <ul style="list-style-type: none"> • Exercício 19 - Flexionando a página (2 de 3h) • Exercício 20 - Série de integração CSS (1 de 4h) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> • Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”)</p> <p>Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	
Sessão 11		
3h	<p>Exercícios:</p> <p>Exercício 20 - Série de integração CSS (3 de 4h)</p> <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> • Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”)</p> <p>Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	

Hora início	Atividades	Recursos e materiais:
Sessão 12		
3h	<p>Exercícios:</p> <ul style="list-style-type: none"> • Exercício 21 - Animações CSS (1h) • Exercício 22 - Pequena série de animações CSS (2h) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> • Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”)</p> <p>Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	
Sessão 13		
3h	<p>Exercícios:</p> <ul style="list-style-type: none"> • Exercício 23 - Pequenas animações CSS série 1 (2h) • Exercício 24 - Pequenas animações CSS série 2 (1 de 3h) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> • Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”)</p> <p>Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	
Sessão 14		
3h	<p>Exercícios:</p> <ul style="list-style-type: none"> • Exercício 24 - Pequenas animações CSS série 2 (2 de 3h) • Exercício 25 - Bónus da série 2 de pequenas animações CSS (1 de 4h) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> • Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”)</p> <p>Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	

Hora início	Atividades:	Recursos e materiais:
Sessão 15		
3h	<p>Exercícios:</p> <ul style="list-style-type: none"> Exercício 25 - Bônus da série 2 de pequenas animações CSS (3 de 4h) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”)</p> <p>Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	
Sessão 16		
3h	<p>Exercícios:</p> <ul style="list-style-type: none"> Exercício 26 - Desafio Final (3 de 9h) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”)</p> <p>Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	
Sessão 17		
3h	<p>Exercícios:</p> <ul style="list-style-type: none"> Exercício 26 - Desafio Final (3 de 9h) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”)</p> <p>Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p>	

Hora início	Atividades:	Recursos e materiais:
Sessão 18		
3h	<p>Exercícios:</p> <ul style="list-style-type: none"> Exercício 26 - Desafio Final (3 de 9h) <p><i>Após as 3 horas, fecha-se a sessão com conclusão, discussão e perguntas</i></p>	<ul style="list-style-type: none"> Plataforma – parte exercícios & conteúdos
Avaliação:	<p>Avaliação formal sobre o resultado da aprendizagem (questionários, exercícios de plataforma) – se houver desistências formandos devem preencher “Formulário Satisfação” (presente no documento “Como implementar o seu piloto”)</p> <p>Os formadores devem recolher o feedback dos participantes, bem como fazer suas próprias anotações sobre a sessão.</p> <p>Para encerrar a formação, o(s) formador(es) devem preencher o “Formulário de Observação & Feedback” e os formandos o “Formulário de Satisfação” (presentes no documento “Como implementar o seu piloto”)</p>	
Atividade (Tempo): Lições 1-5 (1h)	<p>Exercício 1 - Adicione algum CSS em linha (30 minutos)</p> <p>Exercício 2 – Planeie as suas férias com estilo (30 minutos)</p> <p>Exercício 3 - Ilhas Canárias (30 minutos)</p> <p>Exercício 4 - Aulas para diferenciar elementos e estilizá-los de maneira diferente (30 minutos)</p> <p>Exercício 5 - Fronteiras (30 minutos)</p> <p>Exercício 6 - Seletor de classe (1h)</p> <p>Exercício 7 - Seletor de pseudoclasse (30 minutos)</p> <p>Exercício 8 - Defina o estilo desta página (30 minutos)</p> <p>Exercício 9 – Parágrafo (30 minutos)</p> <p>Exercício 10 – CSS Diner (30 minutos)</p> <p>Exercício 11 - Propriedades básicas (30 minutos)</p> <p>Exercício 12 - Propriedades de exibição, exercício 1 (1h)</p> <p>Exercício 13 - Propriedades de exibição, exercício 2 (1h)</p> <p>Exercício 14 - Propriedades de posição, exercício 1 (2h)</p> <p>Exercício 15 - Propriedades de posição, exercício 2 (1h)</p> <p>Exercício 16 - Propriedades da posição, exercício 3 (1h)</p> <p>Exercício 17 - Flex-box 1 (1h)</p> <p>Exercício 18 - Flexionando o cartão (2h)</p> <p>Exercício 19 - Flexionando a página (3h)</p> <p>Exercício 20 - Série de integração CSS (4h)</p> <p>Exercício 21 - Animações CSS (1h)</p> <p>Exercício 22 - Pequena série de animações CSS (2h)</p> <p>Exercício 23 - Pequenas animações CSS série 1 (2h)</p> <p>Exercício 24 - Pequenas animações CSS série 2 (3h)</p> <p>Exercício 25 - Bônus da série 2 de pequenas animações CSS (4h)</p> <p>Exercício 26 - Desafio Final (9h)</p>	

Módulo 0 e 1 exercícios

1. Olá mundo	Veja o ficheiro index.html. Atualmente o ficheiro está vazio. Adicione todos os tags necessários para um layout base de um documento html: um doctype, um elemento <html>, um <head> e um <body>. Mas cuidado: não pode colocar estes elementos onde quiser. Estes têm uma função e devem ser colocados numa ordem específica. Leia a documentação se não tiver a certeza de como utilizar estes elementos html. Para finalizar, adicione no seu documento um elemento <p> com a frase 'olá mundo'. O texto deve ser apresentado no navegador.
2. Títulos	Hora de começar a explorar mais elementos html. Veja o ficheiro index.html e o resultado no navegador. Não deve haver nada exibido. Adicione um elemento <h1> com o texto 'título de um tag h1' e veja o resultado. Existe ainda um elemento <h2>, repita o mesmo processo para este tag. Continue a fazer isto para todos os elementos <h#>.
3. Parágrafos	Adicione 6 elementos de parágrafo ao ficheiro index.html, um para cada tag de título. Pode escrever algo ao seu agrado ou copiar o texto a partir daqui: "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quod saepe architecto laborum omnis reiciendis unde et voluptatibus repudiandae eaque quibusdam doloribus officia odit voluptatum minima ab, reprehenderit cupiditate. Consequuntur, sapiente."
4. Negrito cursivo	Deve ter neste momento um documento html com 6 títulos e um parágrafo para cada título. Podemos adicionar algum significado extra ao texto criando mais elementos html. Utilize as tags e para enfatizar uma ou mais palavras no seu texto. Vai notar que o navegador adiciona alguma estilização ao texto encapsulado numa tag ou . Certifique-se de ler a documentação relativa a estes elementos html.
5. Listas	Abra o ficheiro index.html e compare o código html com o resultado do documento. Identifique 3 tipos diferentes de listas: ordenadas, desordenadas e listas de definição. Certifique-se de verificar a documentação destas tags html! No elemento <body>, adicione uma lista de cada tipo, com conteúdo criado por si, por baixo das listas-exemplo.
6. Links relativos	<p>O verdadeiro objetivo da web era criar uma grande rede de ficheiros ligados uns aos outros por hiperligações (links), é por isso que utilizamos a HyperText Markup Language (ou html – em português “Linguagem de Marcação de Hipertexto”). Podemos conseguir isto usando a tag <a>, ou tag âncora, para ligar a um ficheiro diferente.</p> <p>Para que a tag <a> saiba onde se ligar, teremos de lhe fornecer alguma informação extra. Fazemo-lo utilizando o atributo 'href'. Os atributos são sempre adicionados na tag de abertura de um elemento.</p> <p>Para ligar uma página diferente no seu domínio web, deve usar links relativos. Neste caso, o atributo href tem o caminho para o documento que está a querer ligar.</p> <p>Por exemplo:</p> <pre>Link to a page</pre> <p>O elemento html acima será ligado ao documento html na pasta de páginas chamada 'pagename.html'. Entre as tags de abertura e fecho pode fornecer o texto para a hiperligação.</p> <p>No ficheiro index.html, adicione um link relativo para a página “Acerca” no diretório de páginas, bem como um “alvo” (target) para que quando cliquemos no link, abra um novo separador.</p>
7. Imagens	<p>Até agora, apenas adicionámos diferentes formas de textos no documento html. Títulos, parágrafos, listas, etc. A maior parte dos sites hoje em dia é visualmente mais agradável do que o texto preto numa página branca, como a que temos agora. Uma forma de animar o seu website é adicionando imagens. Para isso, utilizamos o elemento html . Ao contrário das tags anteriores que usámos, esta não tem uma tag de abertura e fecho. Consiste numa tag “auto-fechada”, escrita da seguinte forma: . Dentro desta tag podemos adicionar diferentes atributos. Os seguintes atributos podem ser adicionados a uma tag img: 'src', 'width', 'height'.</p> <p>Atributo 'src': o src, ou atributo source ('fonte' em Português), diz à tag img que imagem deve exibir. Atributo 'width': define a largura da imagem. Atributo 'height': define a altura da imagem.</p> <p>Tente adicionar 'funny-goose-sitting-Karen-Arnold.jpg' a esta tag img e preencha os outros atributos em branco.</p> <p>Informações de copyright da imagem: Karen Arnold – domínio público. Fonte: https://www.publicdomainpictures.net/nl/view-image.php?image=34335&picture=funny-goose-zittend</p>

Artigo para HTML
Converta a seguinte carta para um HTML
válido, recorrendo a elementos HTML
semânticos.
(Fonte: Marking up a letter por Mozilla
contributors, licença sob CC-BY-SA 2.5)

<p>Semântica estrutural/de bloco:</p> <ul style="list-style-type: none"> • O documento geral deve ser estruturado adequadamente, incluindo um doctype, e elementos <html>, <head> e <body>. • A carta em geral deve ser marcada com uma estrutura de parágrafos e cabeçalhos, com exceção dos pontos abaixo. Existe um cabeçalho de alto nível (a linha "Re:") e três títulos de cabeçalho de segundo nível. • As datas de início de semestre, as disciplinas de estudo e as danças exóticas devem ser marcadas utilizando o tipo de lista apropriado. • Os dois endereços devem ser colocados dentro do elemento <address>. Cada linha do endereço deve estar numa nova linha, mas não num novo parágrafo. 	<p>Semântica inline⁸:</p> <ul style="list-style-type: none"> • Os nomes do remetente e do recetor (e o "Tel" e o "Email") devem ser marcados como muito importantes. • Devem ser fornecidos elementos apropriados às quatro datas do documento, contendo datas legíveis por máquinas. • As cinco siglas/abreviaturas no texto principal da carta devem ser marcadas para fornecer expansões a cada sigla/abreviatura. • Os seis sub/sobrescritos devem ser marcados adequadamente – nas fórmulas químicas, e nos números 103 e 104 (devem ser 10 à potência de ou 3 ou 4, respetivamente). • Tente marcar com forte importância/ênfase pelo menos duas palavras apropriadas no texto. • Devem ser adicionadas hiperligações em dois sítios; adicione links apropriados e com títulos. Basta usar http://example.com para a localização a que os links apontam. • Deve marcar, com os elementos apropriados, a citação do lema da universidade. 	<p>O cabeçalho do documento:</p> <ul style="list-style-type: none"> • O conjunto de caracteres do documento deve ser especificado como sendo utf-8 utilizando uma meta-tag apropriada. • O autor da carta deve ser especificado numa meta-tag apropriada. • O CSS fornecido deve ser incluído dentro de uma tag apropriada.
Exercício 8 A carta⁹		
<p>Dr. Eleanor Gaye Awesome Science faculty University of Awesome Bobtown, CA 99999, USA Tel: 123-456-7890 Email: no_reply@example.com 20 January 2016 Miss Eileen Dover 4321 Cliff Top Edge Dover, CT9 XXX UK Re: Eileen Dover university application Dear Eileen, Thank you for your recent application to join us at the University of Awesome's science faculty to study as part of your PhD next year. I will answer your questions one by one, in the following sections. Starting dates We are happy to accommodate you starting your study with us at any time, however it would suit us better if you could start at the beginning of a semester; the start dates for each one are as follows: First semester: 9 September 2016 Second semester: 15 January 2017 Third semester: 2 May 2017 Please let me know if this is ok, and if so which start date you would prefer. You can find more information about important university dates on our website.</p>		
Exercício 9 Objetos de estudo		
<p>At the Awesome Science Faculty, we have a pretty open-minded research facility — as long as the subjects fall somewhere in the realm of science and technology. You seem like an intelligent, dedicated researcher, and just the kind of person we'd like to have on our team. Saying that, of the ideas you submitted we were most intrigued by are as follows, in order of priority: Turning H2O into wine, and the health benefits of Resveratrol (C14H12O3.) Measuring the effect on performance of funk bass players at temperatures exceeding 30°C (86°F), when the audience size exponentially increases (effect of 3 × 103 increasing to 3 × 104.) HTML and CSS constructs for representing musical scores. So please can you provide more information on each of these subjects, including how long you'd expect the research to take, required staff and other resources, and anything else you think we'd need to know? Thanks.</p>		
Exercício 10 – Passos de dança exóticos¹⁰		
<p>Yes, you are right! As part of my post-doctorate work, I did study exotic tribal dances. To answer your question, my favourite dances are as follows, with definitions: Polynesian chicken dance A little known but very influential dance dating back as far as 300BC, a whole village would dance around in a circle like chickens, to encourage their livestock to be "fruitful". Icelandic brownian shuffle Before the Icelanders developed fire as a means of getting warm, they used to practice this dance, which involved huddling close together in a circle on the floor, and shuffling their bodies around in imperceptibly tiny, very rapid movements. One of my fellow students used to say that he thought this dance inspired modern styles such as Twerking. Arctic robot dance An interesting example of historic misinformation, English explorers in the 1960s believed to have discovered a new dance style characterized by "robotic", stilted movements, being practiced by inhabitants of Northern Alaska and Canada. Later on however it was discovered that they were just moving like this because they were really cold. For more of my research, see my exotic dance research page. Yours sincerely, Dr Eleanor Gaye University of Awesome motto: "Be awesome to each other." -- The memoirs of Bill S Preston, Esq</p>		

⁷ Informação adaptada de: https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Marking_up_a_letter

⁸ Secção adaptada de Marking up letter – W3CSchool, disponível em: <https://www.w3cschool.cn/webstart/webstart-marking-up-a-letter.html>

⁹ disponível em <https://mdn.github.io/learning-area/html/introduction-to-html/mark-up-a-letter-finished/>

¹⁰ A resposta foi adaptada de 2 fontes: <https://discourse.mozilla.org/t/mark-up-a-letter-assessment/24676> e <https://github.com/mdn/learning-area/blob/master/html/introduction-to-html/mark-up-a-letter-start/letter-text.txt>

<p>Exercício 11 – Formatação de receitas Fonte: Marking up a letter by Mozilla contributors, licença CC-BY-SA 2.5 Adicione os elementos HTML em falta à receita no ficheiro index.html.</p>	<p style="text-align: center;">Receita rápida de húmus¹¹</p> <p>Com esta receita, faz um húmus rápido e saboroso, sem confusão. Foi adaptada de uma variedade de receitas diferentes que li ao longo dos anos. Húmus é uma deliciosa pasta grossa usada frequentemente em pratos Gregos e do Médio Oriente. É muito saboroso com salada, carnes grelhadas e pão pitta.</p> <p>Ingredientes 1 lata (400g) de grão-de-bico 175g de taíne (tahini) 6 tomates secos Metade de um pimento vermelho Uma pitada de pimenta-caiena 1 dente de alho Uma pitada de azeite</p> <p>Instruções</p> <p>Retire a pele do alho e pique grosseiramente. Retire todas as sementes e caule do pimento, e pique grosseiramente. Adicione todos os ingredientes num processador de alimentos. Pique todos os ingredientes numa pasta. Se quiser um húmus mais espesso e com pedaços, pique por pouco tempo. Se quiser um húmus mais suave, pique por mais tempo. Para um sabor diferente, poderá tentar misturar um pouco de sumo de limão e coentros, malagueta, lima e chipotle, harissa e hortelã, ou espinafres e queijo feta. Experimente e veja o que funciona para si. Armazenamento Coloque o húmus acabado num recipiente selado no frigorífico. Poderá utilizá-lo até no máximo uma semana depois de feito. Se começar a ficar com gás, deve deitá-lo fora. Pode congelar o húmus, deverá ser utilizado no período de dois meses.</p>
<p>Exercício 12 – Agricultor chinês</p> <ul style="list-style-type: none"> • Traduza a página html para html semântico recorrendo às tags html adequadas: h1, h2, blockquote, q, img, img, hr, figure e caption, table, th, tr, td, ul ou ol e li. • Não utilize div ou span: não fornecem qualquer semântica. • Encontre, para cada uma destas tags, a origem do seu nome (é assim que nos lembramos delas). Em caso de dúvida, procure a resposta na documentação. • Adicione dois ou três links à sua escolha na página html através da tag a. • Existe alguma parte que deva ser considerada como cabeçalho? Caso haja, agrupe-a numa tag header. • E um rodapé? Se sim, agrupe este conteúdo numa tag footer • Coloque todas as palavras "Maybe" numa tag em ou strong. • Adicione o atributo Alt às imagens. Qual é o propósito deste atributo? • Adicione uma classe "good" ou "bad" às tags que rodeiam as palavras "Good" e "Bad". • Encontre o atributo de link para indicar a página a que o link deve conduzir e adicione-o. • Certifique-se de que, ao clicar nos links, a página abre um novo separador de navegador. 	<p style="text-align: center;">A História do Agricultor Chinês</p> <p>É uma coisa boa? É uma coisa má? Não sei. 11 de maio, 2016, Alexandre Gorius Na natureza, o Bem & Mal não existem. O próprio Universo não é bom nem mau, mas um potencial em desenvolvimento permanente.</p> <p>Na vida, existem 4 tipos de pessoas. - pessoas que pensam que a Internet é boa, - pessoas que pensam que a Internet é má, - pessoas que pensam que a Internet são gatos, - pessoas que não pensam no assunto.</p> <p>[Image : images 4 students/youngBlogger.jpeg]</p> <p>No espaço, a dar a volta, a olhar para a face da Terra. Do ponto de vista daquela rocha sobre Marte ou a opinião daquele gás sobre Júpiter, nada se preocupa com os seus problemas pessoais. Lá em cima, nada é assim tão sério, e não existe tal coisa como um problema¹². Aqui em baixo, a única realidade é a que está dentro do nosso cérebro. Nada existe se o cérebro não pensar. Nada existe se os nossos sentidos não transmitirem a mensagem. Isso significa que se perdermos sentidos, a nossa realidade muda. Isto é possível porque a nossa percepção é limitada: não pode ver a 360 graus, por exemplo. Se traduzirmos mal a mensagem, a nossa realidade muda. Se pensarmos de certa forma, a nossa realidade muda dependendo disso. Portanto, se podemos mudar assim tanto a realidade, tudo é falso. Existem provas disso. Apaixone-se por alguém e vai ver essa pessoa tornar-se divina para si. Saia dessa constatação e a pessoas divina cairá das nuvens para o solo, revelando-se uma pessoa diferente. A verdade é que pessoa continua a mesma, mas mudou de ideias sobre ela. Tudo funciona assim: não é só que consegue criar o seu próprio mundo, mas sim que o faz passivamente sem dar por isso.</p> <p>[Image : images 4 students/shineyRock.jpeg caption: "what does he think about it?"]</p> <p>A história¹³ [Image : images 4 students/chineseman.jpeg caption: "The Old Chinese Farmer"] Era uma vez um agricultor chinês cujo cavalo fugiu. Naquela noite, todos os seus vizinhos vieram confortá-lo. "Lamentamos muito que o seu cavalo tenha fugido. É um grande infortúnio." O agricultor disse: "É bom? É mau? Não sei." No dia seguinte, o cavalo voltou, trazendo com ele sete cavalos selvagens, e à noite todos voltaram e disseram: "Oh, que sortudo. Que grande reviravolta. Agora ficou com oito cavalos! O agricultor disse novamente: "É bom? É mau? Não sei." No dia seguinte, o filho tentou domar um dos cavalos, e enquanto o montava, foi atirado ao chão e partiu a perna. Os vizinhos disseram: "Meu deus, é uma pena o que aconteceu.", e o agricultor respondeu: "É bom? É mau? Não sei." No dia seguinte, os oficiais de recrutamento vieram recrutar pessoas para o exército, e rejeitaram o seu filho porque tinha uma perna partida. Mais uma vez, todos os vizinhos vieram e disseram: "Não é fantástico?!". Mais uma vez ele disse "É bom? É mau? Não sei."</p>

¹¹ Receita retirada de: <https://github.com/mdn/learning-area/blob/master/html/introduction-to-html/html-text-formatting/text-start.html>

¹² Secção adaptada de: <https://byrslf.co/theres-no-good-nor-bad-85a62f371abc>

¹³ A história do agricultor chinês, por Alan Watts, retirada de: <https://www.craftdeology.com/the-story-of-the-chinese-farmer-by-alan-watts/> e de: <https://wellsbaum.blog/alan-watts-story-of-the-chinese-farmer>

- *Encontre o atributo que exibe uma pequena caixa de texto quando colocamos o cursor sobre links, como esta:*
image
Terminou?
Muito bem! Acabou de completar o primeiro capítulo! Mas provavelmente reparou que ainda não fizemos coisas esteticamente apelativas. Vamos tratar disso no próximo capítulo: CSS

Todo o processo da natureza é um processo integrado de imensa complexidade, e é realmente impossível, quando algo acontece, dizer se é bom ou mau - porque nunca se sabe qual será a consequência do infortúnio; ou quais serão as consequências da boa sorte.

Alan Watts

Aqui está o diálogo em forma de tabela

```

+-----+-----+
|                                     |                                     |
| Evento                             | Bom ou Mau?                             |
+-----+-----+
|                                     |                                     |
| Perder um cavalo                   | Talvez                                   |
+-----+-----+
|                                     |                                     |
| Ganhar mais cavalos                 | Talvez                                   |
+-----+-----+
|                                     |                                     |
| Lesão do filho                     | Talvez                                   |
+-----+-----+
|                                     |                                     |
| Evitar a guerra                     | Talvez                                   |
+-----+-----+
Tabela que representa a filosofia do agricultor.

```

Conclusão

Bom e Mau são apenas uma perspectiva. As bananas são boas para os macacos, mas os macacos não são bons para as bananas. Não vamos fazer disto um grande problema...

Copyright Just Another Company 2017. Todos os direitos reservados. Adaptado deste artigo: <https://wellsbaum.blog/2018/01/27/alan-watts-the-story-of-the-chinese-farmer/>

Exercícios do módulo 2

<p>Exercício 1 Adicionar algum CSS inline</p>	<p>Instruções Aqui estão alguns conteúdos HTML. Utilize o atributo style para:</p> <ul style="list-style-type: none"> • fazer com que o corpo fique com um fundo em cinza claro (utilize este valor: #DDDDDD) • colocar o texto do parágrafo em azul • fazer com que as tags 'strong' pareçam destacadas, adicionando uma cor de fundo amarelo. • sublinhar o título • adicionar uma margem inferior de 10 pixels à Lista Items (tags li) • modelar a Lista Ordenada (tag ol) para que o conteúdo fique a vermelho. <pre><body> <h1>Os 7 passos para planejar as suas férias</h1> <p>Todos precisamos de uma pausa de vez em quando! Que tal fazer uma caminhada nas montanhas, ou talvez seja mais uma pessoa de praia?</p> <p>Aqui fica um memorando sobre como garantir que as suas férias são um sucesso! </p> Passo 1: Decidir para onde ir. Passo 4: Encontrar voos baratos. Passo 5: Encontrar um local ótimo para ficar. Passo 6: Ligar para as companhias do cartão de crédito. Passo 7: Orçamentar ponderadamente a sua viagem. </body></pre>
<p>Exercício 2 Planeie as suas férias com estilo!</p>	<p>Instruções Refaça o exercício acima removendo todo o estilo inline e coloque todas as propriedades de estilo numa tag <style>. Lembre-se: deve ir acima ou abaixo do conteúdo? Se não tem a certeza, tente os dois e veja qual funciona...</p>
<p>Exercício 3 Ilhas Canárias</p>	<p>Instruções Aqui está algum conteúdo HTML descrevendo as Ilhas Canárias como um destino de férias de sonho. O seu trabalho é de fazer com que pareça melhor! Para fazer os exercícios, adicione todo o seu CSS acima do html utilizando uma tag <style>. Faça cada um destes exercícios na ordem dada.</p> <ol style="list-style-type: none"> 1. Mude o fundo da página para uma cor bege 2. Altere toda a cor do parágrafo para castanho escuro (utilize o valor saddlebrown) 3. Faça com que todos os tags fiquem a negrito e cor-de-rosa 4. Pesquise na documentação para saber como modelar todos os parágrafos de forma a que tenham uma margem inferior de 20 pixéis (utilize a unidade px). 5. Modele o H1 para itálico, e com um tamanho de fonte de 32px 6. Modele o H2 de modo a que tenha uma linha cinzenta fina por baixo (veja a documentação para border-bottom e utilize a cor cinza claro #DDDDDD) e um tamanho de fonte de 25px 7. Modele os parágrafos de modo a que tenham um tamanho de fonte de 16px. <pre><h1>Bem-vindo às Ilhas Canárias</h1>¹⁴ <p> Florestas de pinheiros perfumados, vulcões assombrosos, paisagens lunares, enseadas secretas de areia, quilómetros de dunas ao estilo do Sahara, resorts à beira-mar - as belas e únicas Ilhas Canárias possuem muitas facetas tentadoras. </p> <h2>Paisagens de Outro Mundo</h2> <p>Maravilhe-se com os picos de florestas de pinheiros no montanhoso interior de Gran Canária, as cascatas em queda de La Palma ou a vegetação subtropical do Parque Nacional de Garajonay de La Gomera. Em seguida, contraste esta exuberância com as extraordinárias planícies nuas que ladeiam El Teide em Tenerife, a surreal festa de cores que brilha sobre os campos de lava de Lanzarote, as suaves encostas cheias de flores de El Hierro e as infinitas planícies polvilhadas de catos de Fuerteventura. As temperaturas quase perfeitas das Ilhas Canárias permitem que, durante qualquer altura do ano, possa absorver todas as variadas e fantásticas paisagens que, de outra forma, apenas conseguiria experienciar cruzando continentes.</p> <h2>O Fantástico Mundo ao Ar Livre</h2></pre>

¹⁴ História retirada de: <https://www.lonelyplanet.com/canary-islands> e: <https://www.globeguides.co/destinations/YXZU3UGA4VJO936AZ7YH> e: <https://euresorts.co.uk/destinations/canaries/>

	<p><p>É esta mesma diversidade que torna as aventuras ao ar livre um prazer tão facilmente acessível e importante das Canárias. Caminhe pelos muitos percursos pedestres que atravessam as ilhas, desde trilhos costeiros sinuosos até caminhadas de montanha desafiantes ou passeios florestais tranquilos; faça snorkelling ou mergulhe em águas quentes habitadas por mais de 350 espécies de peixes (e o estranho naufrágio); ou bombeie adrenalina cavalgando no vento e nas ondas -o kitesurf, windsurf, surf e parapente são todos desportos comuns aqui. Em seguida, abranda um pouco com passeios de cavalo, barco, caiaque e paddle ou yoga à beira-mar.</p></p> <p><h2>Arte & Arquitetura</h2>¹⁵</p> <p><p>Contrariamente ao que se pensa, as Ilhas Canárias são imensamente ricas em arte e arquitetura originais, às vezes só precisamos de saber onde procurar. As espetaculares telas surrealistas do aclamado pintor Óscar Domínguez agraciam a sua terra natal em Tenerife; as enormes esculturas abstratas de Martín Chirino são impossíveis de perder na Gran Canária; e as "intervenções" inspiradas de César Manrique aparecem por toda a parte em Lanzarote (e não só). Por todo o lado, procure as emblemáticas varandas de madeira, pátios internos frondosos e fachadas alegremente pintadas que tipificam a arquitetura vernacular canarina, e aparecem em encantadoras igrejas com sombra de palmeira, muitas das quais remontam a vários séculos atrás.</p></p>
<p>Exercício 4 Utilize as classes para diferenciar e modelar elementos</p>	<p>Instruções Vamos fazer com que o último exercício pareça melhor, utilizando técnicas mais avançadas. Utilize a Folha de Consulta do CSS e a Documentação para encontrar a sintaxe adequada. A formulação é feita para dar uma pista sobre o nome de propriedade correto.</p> <ol style="list-style-type: none"> 1. Modele a tag 'body' para que tenha um preenchimento de 30px e uma cor de fundo cinza claro (utilize estes valores, vermelho: 220, verde: 220, azul: 220) 2. Modele todos os parágrafos de modo a que a sua fonte seja de 16px, com uma altura de linha de 120%, com uma cor escura (mas não preta) (utilize estes valores: vermelho: 25, verde: 25, azul: 25) 3. Adicione uma classe à tag de 'parágrafo' por baixo da tag H1, com um valor de "introdução" (a sintaxe é como isto: <p class="classname"> e no CSS, pode mencionar que é uma classe adicionando um ponto final à sua frente, como aqui .classname (se estiver bloqueado, leia mais sobre "Selecionadores" na Documentação ou nas aulas) 4. Modele o parágrafo "introdução", fixando o seu tamanho de fonte em 140%. Mude a altura da linha para 200% e dê-lhe um estilo de fonte "itálico".
<p>Exercício 5 Limites</p>	<p style="text-align: center;">Instruções</p> <p>Veja este elegante duplo sublinhado abaixo: a linha azul grossa espalha o comprimento da palavra, enquanto a linha cinzenta se espalha por todo o bloco. Tente imitar o resultado utilizando CSS nas tags H2 e P!</p> <p>Dica 1: Terá de utilizar a propriedade de exibição no elemento h2. Verifique a documentação para saber qual o valor de que necessitará. (O valor padrão é block e é por isso que a linha cobre toda a página.)</p> <p>Dica 2: Encontre uma maneira de definir a família de fonte para 'sans-serif'.</p> <p>Dica 3: Cor: #2A99FB.</p> <p>Quem somos</p> <p>Veja este elegante duplo sublinhado abaixo: a linha azul grossa espalha o comprimento da palavra, enquanto a linha cinzenta se espalha por todo o bloco. Tente imitar o resultado utilizando CSS nas tags H2 e P!</p>
<p>Exercício 6 Seletor de classe</p>	<p>Instruções Adicione as classes necessárias ao html abaixo. Em seguida, modele cada um dos nomes das marcas no html para que se pareçam o mais possível com o seu logótipo. Tenha em atenção que também poderá adicionar mais tags HTML se quiser (especialmente para a FedEx, terá de o fazer!). Neste caso, use tags , são úteis para colorir parte de um texto. Aqui está o resultado que deverá tentar alcançar (ou o mais próximo possível).</p> <p>Marcas</p> <ol style="list-style-type: none"> 1. Toyota 2. Toyota 3. FedEx Express
<p>Exercício 7 Seletor de pseudo-classes</p>	<p>Instruções Adicione o estilo necessário para que, quando o cursor passe sobre a palavra abaixo, esta mude de cor.</p> <ul style="list-style-type: none"> • por definição: o texto é amarelo • quando passar com o cursor: o texto é vermelho (estado:hover) • ao clicar: o texto é azul (estado:active) • após clicar: o texto é verde (estado:visited) <p>Aqui está uma ilustração do resultado a alcançar:</p> <p>Estados de links Link</p>

¹⁵ Texto retirado de: <https://euresorts.co.uk/destinations/canaries/>

Exercício 8 Modele esta página¹⁶

Instruções

Modele os elementos HTML de acordo com as seguintes instruções. NÃO ALTERE O HTML EXISTENTE PARA FAZER ISTO. ESCREVA APENAS EM CSS!

- Dê ao elemento <body> um fundo de #bdc3c7
- Coloque o elemento <h1> a #9b59b6
- Coloque todos elementos <h2> a laranja
- Coloque todos os elementos a azul (escolha o seu próprio azul hexadecimal)
- Altere o fundo em cada parágrafo para ser amarelo
- Dê uma borda vermelha de 3px a todos os inputs
- Dê a tudo com a classe 'olá' um fundo branco
- Dê ao elemento com um id 'especial' uma borda azul sólida de 2px (escolha o seu próprio azul RGB)
- Dê a todos os elementos <p> agrupados dentro dos divs uma fonte de 25px (tamanho da fonte: 25px)
- Dê aos inputs de tipo 'texto' um fundo cinza
- Dê a ambos os <p> dentro do 3º <div> um fundo rosa
- Dê ao 2º <p> dentro do 3º <div> uma borda branca de 5px
- Coloque o no 3º elemento <div> a branco com uma fonte de 20px (tamanho da fonte: 20px)
- Faça com que todas as caixas de verificação tenham uma margem esquerda de 50 px (margem esquerda: 50px)
- Coloque todos os elementos <label> em MAIÚSCULAS sem mudar o HTML
- Altere a primeira letra do elemento id 'especial' para verde com uma fonte de tamanho 100px (tamanho de fonte: 100)
- Faça a cor do elemento <h1> mudar para azul quando o cursor passar sobre este
- Coloque os elementos <a> que tenham sido visitados a cinza

¹⁶ Exercício adaptado de: <https://gist.github.com/weav797/78888a9b10db101eef1e87d1f18f80bc>

Exercício 9 PARÁGRAFO FORA DE UM DIV

Eu sou um parágrafo com uma classe

Eu sou um parágrafo com um ID

Sou um h2 incrível

Festa no telhado, sintetizador top, Roof party yr hella synth, torrada de quatro dólares de narval de Wes Anderson antes de esgotarem o lo-fi retro. iPhone de Austin apareceu fazenda-a-mesa. A PBR&B do McSweeney ennuí mensageiro destilaria antes de esgotar Portland pacote de lobo YOLO. Carbohidratos lentos locavore fundo de segurança fazenda-a-mesa. Pinterest gastropub lo-fi, fundo de segurança do McSweeney VHS gasto rapariga ugh Austing. Caixa de mensageiro banjo lenhador, quaisquer 3 lobos lua normcore. Pack de pugs 3 lobos lua, máquina de escrever orgânica bigode chia cenógrafo seitan gasto salvia garrafa azul ugh iPhone. Pack Williamsburg comércio direto, saúde de flanela prensado a frio listicle gótico assimétrico mixtape arte de rua freegan derramar sobre qualquer coisa.

Coisas que preciso fazer

- Passear o cão
- Alimentar o cão
- Dar banho ao cão

Sou outro parágrafo h2 incrível

Tumblr Cardigan Tumblr batido, til 3 lobos lua Portland. Taxidermia gótico da saúde da herança selfies blog lo-fi, pós-irónico mestre de limpeza dedos tacho normcore. Kickstarter twee xadrez, café de origem sustentável vinil lo-fi Pinterest barriga de porco skate cronut 3 lobos lua. Normcore salvia café de origem única, sob medida Austin swag Godard antes de esgotar pertubar kogi locavore. Shoreditch Vice, artesão Vesturário Americano mestre da limpeza yr salvia vegan. Sob medida herança de letterpress batatas fritas de couve profundo v quatro loko. Rio Lomo sustentável colocar um pássaro nele fundo fiduciário pós-irónico

Sou o segundo parágrafo dentro deste div!

PARÁGRAFO FORA DO DIV

Um H2 menos incrível

. Pack Williamsburg comércio direto, saúde de flanela prensado a frio listicle gótico assimétrico mixtape arte de rua freegan derramar sobre qualquer coisa. Normcore salvia café de origem única, sob medida Austin swag Godard antes de esgotar pertubar kogi locavore. Caixa de mensageiro banjo lenhador, quaisquer 3 lobos lua normcore. Pack de pugs 3 lobos lua, máquina de escrever orgânica bigode chia cenógrafo seitan gasto salvia garrafa azul ugh iPhone artesão Vesturário Americano mestre da limpeza yr salvia vegan.

Um último parágrafo aqui!

Sou um link para o facebook Sou um outro link para o facebook

Nome

Palavra-passe

PARÁGRAFO FORA DE UM DIV

Exercício 10 – CSS Diner

Fonte: [css diner](#), licença sob [Mozilla Public License 2.0](#)¹⁷

Bem-vindo ao CSS Diner!

CSS Diner é um pequeno "jogo" divertido que ensina o básico dos seletores CSS. Sabemos que entender a maneira como os seletores de CSS funcionam pode ser confuso no início, mas assim que dedicar um pouco mais de tempo ao tópico, ele tornar-se-á simples!

Neste jogo verá uma mesinha elegante que servirá um prato com ou sem comida. O trabalho é selecionar os itens solicitados mostrados num pequeno parágrafo no topo da tela.

Use o editor e o visualizador de HTML para localizar o seletor CSS correto.

Consegue chegar ao nível 32?

Boa sorte!

Exercício 11 - Propriedades básicas

É importante levar o seu tempo com estes exercícios. Procure compreender o que se está a passar. Porque é que esta cor está a mudar? Porque é que o tamanho é diferente? Etc.

Aprender o básico

Todos gostamos de coisas básicas! – Não pense demasiado nisso 🤖

Dê-me uma cor vermelho e alinhe-me ao centro.

Dê-me um fundo azul.

Dê-me uma margem maior à esquerda.

Dê-me um tamanho de letra de 10px. Dê-me um tamanho de letra de 20px.

Não me mude! Mas dê ao 'lorem' uma linha de altura de 10px e espaço de letras de 4px.

Lorem ipsum dolor sit amet consectetur adipiscing elit. Doloribus illo tempora, dolores iure rem quos neque harum omnis unde. Nam pariat quidem nostrum nisi harum voluptatem tempore impedit fugiat minus? Esse quo debitis nam commodi architecto praesentium, ullam laborum quasi odit. Quis recusandae et magni possimus mollitia at. Distinctio, esse? Consectetur quam enim consequatur delectus quod in est repellendus a. Tempora, labore autem? Ipsum, exercitationem officii velit iste quasi facilis id deleniti. Officiis temporibus est repellat numquam omnis quam quia non sint! Esse velit dolore impedit deleniti reiciendis unde hic. Commodi iure eius sunt minima quod repellendus sed magni explicabo architecto sit a temporibus, assumenda debitis accusantium omnis non soluta natus laboriosam. Optio ad assumenda sunt, dolores consequatur perspiciatis nihil.

Dê-me um sublinhado.

Remova a minha marca de ponto.

Altere-me para maiúsculas apenas.

Dê-me uma borda de 2px, pontilhada, não sólida.

Dê-me uma borda de 4px à direita e sólida.

Coloque este texto a negrito. Dê-lhe uma cor branca. Dê-lhe um grande tamanho de fonte. Adicione a imagem neste projeto ao fundo. Dê-lhe uma largura e altura de 200px.

Dê a este botão uma cor de fundo, dê 5px de preenchimento à parte superior, 10px de preenchimento à parte inferior e 15px à direita e à esquerda.

Depois disto, dê ao botão cantos redondos, exceto no canto inferior direito.

Depois disto, quando passar o cursor sobre o botão, a cor de fundo deve mudar.

Botão

Faça uma forma de círculo em torno da imagem. Dica: border-radius e overflow. Dê-lhe uma largura e altura de 150px. Quando passamos o cursor sobre a imagem, esta escala. Mas o círculo não deve aumentar.

Remova o sublinhado e as marcas de ponto, dê a cada li uma cor de fundo diferente. Dê a cada li um valor de margem diferente (aleatória) e, quando passar o cursor, muda o tamanho da fonte. Todas as cores da fonte devem ser brancas.

Página Principal

Loja

Acerca

Contactos

Propriedades de exibição

Exercício 12 - Propriedades de exibição 1

Antes de começarmos

Antes de começarmos com estes exercícios, quero que olhe com atenção para a documentação e procure o termo display. Leia bem para ver qual é a diferença e o que faz exatamente.

Demore o tempo que tiver de demorar! É importante que perceba as bases disto.

Instruções

Encontre uma forma de mostrar o conteúdo no editor assim:

image

NOTA: Não altere as *tags* HTML ou o conteúdo.

Apenas trabalhe com o estilo e modelagem. Lembre-se do que acabámos de aprender.

Calendário

Lembrete marcado para as 20:00.

¹⁷ O jogo encontra-se disponível aqui: <https://flukeout.github.io/>

Exercício 13 - Propriedades de exibição 2

Instruções

Encontre uma forma de mostrar o conteúdo no editor assim:

image

NOTA: Não altere as *tags* HTML ou o conteúdo.

Apenas trabalhe com o estilo e modelagem. Lembre-se do que acabámos de aprender.

Este espaço deve ter uma largura de 200px. Esta lista

deve acompanhar numa linha e não deve utilizar qualquer largura, mas também não estamos autorizados a removê-la...

Propriedades de posição

Exercício 14_ Propriedades de posição 1

Antes de começarmos

Antes de começarmos com estes exercícios, quero que olhe com atenção para a documentação e procure o termo position. Leia bem para ver qual é a diferença e o que faz exatamente.

Demore o tempo que tiver de demorar! É importante que perceba as bases disto.

Mãos ao trabalho

Como um update de tudo o que temos visto até agora, quero que recrie esta simples navbar (barra de navegação).

image

Depois disso, encontre uma maneira de deixar a navbar no topo do ecrã.

Faça scroll para baixo!

A navbar ainda deve ser visível aqui em baixo!

Exercício 15_ Propriedades de posição 2

Agora recrie este estilo o melhor possível:

image

Agora faça com que, se fizermos scroll para baixo, cada caixa colapse uma na outra. (Deixe um espaço de 20px entre o topo e as caixas.)

Exercício 16_ Propriedades de posição 3

Agora recrie este estilo o melhor possível:

image

Não se esqueça de utilizar display!

superior esquerdo

superior direito

inferior esquerdo

inferior direito

Flex-box

Exercício 17 Flex-box 1

Até agora não tocámos na 'flex-box', e há uma boa razão para isso. Torna as nossas vidas como programadores muito mais fáceis, tão fáceis que se esqueceriam de cabeçalhos, blocos inline, etc. Aconselho-o vivamente a olhar com atenção para a documentação para ter um breve entendimento sobre o poder da **dar**- flex-box.

Tem de fazer uma cópia exata disto utilizando apenas a flex-box:

Aprender a flexionar

Que é apenas um pouco diferente deste tipo de flexão  – Faça uma cópia exata da imagem do exemplo!

1. Centrar os divs

2. Deve haver espaço entre eles

3. Os espaços devem ser do mesmo tamanho

4. As caixas começam do início

5. As caixas começam do fim

6. As caixas devem estar espalhadas em altura

7. Tornar a ordem aleatória

8. Inverter a ordem.

Exercício 18 – Cartão de flexão

Para este exercício não utilizaremos nenhum código de configuração. Vamos refrescar as suas competências de HTML! O que vamos fazer?

Um cartão que poderia ser utilizado numa aplicação para um serviço de entrega de pizzas!

image

Cores:

Verde: #444444

Azul: #FFFFFF

Dica: comece sempre com o conteúdo: concentre-se no HTML primeiro, sem se preocupar tanto com o CSS. Quando o seu HTML estiver bom, passe para o CSS. Desta forma, só irá adicionar classes quando requerido pelo CSS, o que é uma boa prática.

Exercício 19 página de flexão

Para este exercício não utilizaremos nenhum código de configuração. Vamos refrescar as suas competências de HTML! O que vamos fazer?

Uma página inicial sobre o **Tim Berners-Lee!**

O nome dele não desperta nada? Não se preocupe, vai conhecê-lo durante este exercício.

Refrescando ideias

Recrie a seguinte estrutura em HTML:

image

Dê uma flexão nesses containers!

Bom trabalho! Agora modele a página nisto:

image

Tente utilizar a flex-box o máximo possível!

Nota: Não vai encontrar a fonte correta, tente apenas aproximar-se o melhor que conseguir.

Dica: comece sempre pelo conteúdo: por isso concentre-se no html primeiro, sem se preocupar muito com o CSS. Quando o seu html for bom, passe para o CSS. Desta forma, só irá adicionar classes quando requerido pelo CSS, o que é uma boa prática.

Exercício 20 Exercício de Integração CSS

Instruções

Trabalha numa agência de Web Design como Front-end Developer. O Designer UI fornece-lhe imagens que representam uma maquete de alta-fidelidade dos componentes de interface do site.

A sua função é reproduzir em html e CSS cada um destes componentes. Cada componente UI foi validado pelo seu cliente. É, portanto, muito importante que o resultado final se aproxime ao máximo das maquetes.

Veja a imagem:

image

Reproduza esta interface o mais fielmente possível utilizando html e CSS. Procure chegar à versão mais próxima possível ao nível pixel Idealmente, não devemos ver qualquer diferença entre a maquete de imagem e o resultado final, sem usar qualquer imagem. As imagens só são aceites para ilustrações, quando não há alternativa CSS.

Cor:

Dica: comece sempre com o conteúdo: por isso concentre-se no html primeiro, sem se preocupar muito com o CSS. Quando o seu HTML for bom, passe para o CSS. Desta forma, só irá adicionar classes quando requerido pelo CSS, o que é uma boa prática.

O CSS tem de estar numa *tag* <style>, sem CSS inline.

Exercício 21 Criar animações CSS

O exercício

Neste exercício, vai construir uma animação complexa, passo a passo.

Como frequentemente com o CSS, este é um exercício de detalhe e precisão (e uma pinga de demência, porque pronto... CSS). Para lhe dar um apoio visual fiável, utilizará, assim, uma técnica que pode ser chamada de "Técnica do guia de fundo de imagem".

Dica: Técnica do guia de fundo de imagem: para cada passo do exercício, utilize as imagens como imagem de fundo do corpo (talvez com uma ligeira opacidade?) para ajudá-lo a modelar com precisão. Claro que, quando terminado, retire a imagem de fundo para um visual mais limpo.

```
<div id="stage" style="background: transparent url(/images/css_animation_exercise_stage.png) 0 0 no-repeat;">
```

- Crie o cenário: adicione um div com o id "stage" (cenário), modele para que apareça exatamente como esta imagem, situado no centro horizontal da sua página de índice.

images

- Adicione um div com um id "hero"
- Modele o #hero para que fique exatamente assim:

images

Agora, construa a animação passo a passo.

Passo 1

Mova o herói para a direita, uma vez.

image

Passo 2

Mova o herói para a direita, em 5 segundos, indefinidamente (animação em loop).

image

Passo 3

Mova o herói para a direita, depois para a esquerda, em 2 segundos, indefinidamente (animação em loop).

image

Passo 4

image

Passo 5-7

Percebeu? Ok, agora siga estas instruções. Pense "pixelização perfeita"

image

Terminou?

Bravo !

Image

Exercício 22 Pequenas séries de animações CSS

Nesta série de exercícios, temos de replicar uma série de elementos.

Faça uma cópia exata dos exemplos e escreva o seu próprio HTML e CSS!

Antes de começar

Por favor, copie e cole este código para o seu editor:

Code

Componente animada:

Código para seta

Ou

CÓDIGO CSS

Exercício 23 Pequena série de animações 1

Nesta série de exercícios, temos de replicar uma série de elementos.

Faça uma cópia exata dos exemplos e escreva o seu próprio HTML e CSS!

ESPECIAL

Este exercício é um pouco diferente dos outros, tente fazer uma pequena página e modelá-la a seu gosto. Inclua este texto algures na página (seja criativo!).

Lonely commute

[Join Australia-based photographer Mark Forbes as he wanders the streets of Tokyo with a camera & heart](#)

full of curiosity documenting the Lonely Commute.

Agora imagine uma pequena galeria de imagens onde pode passar o cursor sobre elas e abrir uma animação na parte inferior para cada uma delas.

Componente animada:

Exercício 24 Pequena série de animações 2

Nesta série de exercícios, temos de replicar uma série de elementos.

Faça uma cópia exata dos exemplos e escreva o seu próprio HTML e CSS!

ESPECIAL

Vai precisar deste texto¹⁸:

Myths about Scrolling and Long Content Pages on the Desktop

Myth #1: Users don't scroll long pages

Users do scroll when the content is relevant, organized properly, and formatted for ease of scanning. In fact, people prefer scrolling the page for content over pagination when the topics within that page answer the right questions. The standard scroll wheel on a mouse, arrow keys, and track pads have made scrolling much easier than acquiring click targets.

Myth #2: Customers don't read information at the bottom of the page

Our eyetracking research show that while users spend 80% of their attention on information above the page fold, they allocate 20% of it to content below the fold. Reluctance to scroll is a behavior of the past. While you should still be mindful of people's limited attention span on websites and prioritize content wisely, you shouldn't fear long formats. People will see the bottom if you give them good reason to go there.

Myth #3: People avoid pages with a lot of content

People have the ability to handle vast amounts of information, when presented properly. In our Writing for the Web courses, we emphasize the requirement for writing well, and more importantly, writing for web-based reading. Reading and scanning patterns are different between web-based and print-based content. While online users typically scan for information, it does not mean they want less information. Websites should not be information light. The same information needs to be written, structured, and presented differently.

Accordions Are Not Always the Answer for Complex Content on Desktops

-

¹⁸ Texto retirado de: <https://www.nngroup.com/articles/accordions-complex-content/> e:

<https://www.linkedin.com/pulse/20140807120829-35361745-above-the-fold-and-other-myths-about-scrolling-websites/>

Exercício 25 Pequena série de animações CSS BÓNUS

Nesta série de exercícios, temos de replicar uma série de elementos.

Faça uma cópia exata dos exemplos e escreva o seu próprio HTML e CSS!

ESPECIAL

Para este vamos criar várias caixas de verificação que terão a seguinte animação ao serem verificadas:

image

Componente animada:

A seta:

Passo um:

image

Passo dois:

image

Passo três:

image

Passo quatro:

image

A caixa de verificação:

Passo um:

image

Passo dois:

image

Passo três:

image

Agora termine-a!

Image

Exercício 26 Desafio Final

Esta é a sua avaliação final. Se conseguir resolver o desafio receberá um certificado que comprova a sua capacidade para iniciar uma formação completa em *Frontend Web Development* e tornar-se um programador profissional de websites (frontend).

Instruções

O dono da empresa aborda-o com um design para a Fundação Banana, solicitando que crie um website que se pareça EXATAMENTE com mesmo. Tente reproduzir o mais precisamente possível o seguinte layout.

Irá encontrar as imagens necessárias para construir a interface na pasta do projeto.

Lembre-se:

1. Planeie o seu trabalho
2. Divida o plano em passos mais pequenos
3. Comece com o conteúdo (o HTML) sem se preocupar com o estilo
4. Debug (encontre e fixe problemas e repita a partir do passo 1).

Códigos de cor & família de fonte:

- amarelo: rgb(242, 201, 76)
- azul: rgb(131, 175, 237)
- família de fonte: **Roboto**

Desafio Final - FRASES

Login

Nome de utilizador

Palavra passe

Iniciar sessão

Por favor, preencha o seu e-mail.

O endereço de e-mail que inseriu é inválido.

Por favor, insira a sua senha.

A senha que inseriu está errada.

Página Inicial

Cursos

Meu perfil

Revisão

Terminar sessão

Mapa do site

Contacte-nos

Índice dos Módulos

Lições

Ler mais

Geral

Página Principal de Cursos

Iniciar Formação

Iniciar Exame

Anterior

Seguinte

Terminar

Parte

Restaurar

Guardar

Submeter

Ativo

Continuar

Datas

Estado

Questões

Passar

Marcar

Iniciar

Finalizar

Em curso

Total

Respondido

Correto

Mostrar detalhes

Obter Certificado

Questão

Não Marcado

Marcado



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

Project Number 2018-1-RO01-KA204-049298