CENTRO PAULA SOUZA

ETEC PROF. MARIA CRISTINA MEDEIROS

Técnico em Informática para Internet Integrado ao Ensino Médio

Prof. Cíntia Maria de Araújo Pinho

MATERIAL DDW

Passo a Passo criação de Web site Pessoal usando HTML5 e CSS3

Ribeirão Pires

2018

# Iniciando com HTML

A Internet está cada vez mais presente no nosso dia-a-dia, não só porque temos cada vez mais acesso, mas também mais conteúdo online. Temos desde sites pessoais e portfólios até grandes portais. Temos blogs e redes sociais.

Conforme a Internet foi crescendo, ficou mais robusta, isto é, as tecnologias que a sustentam são cada vez mais poderosas e versáteis.

Para a criação de conteúdo para a Internet, HTML, CSS e JavaScript são as tecnologias dominantes atualmente. Se você quer colocar um site online, muito provavelmente terá que saber mexer com essas tecnologias.

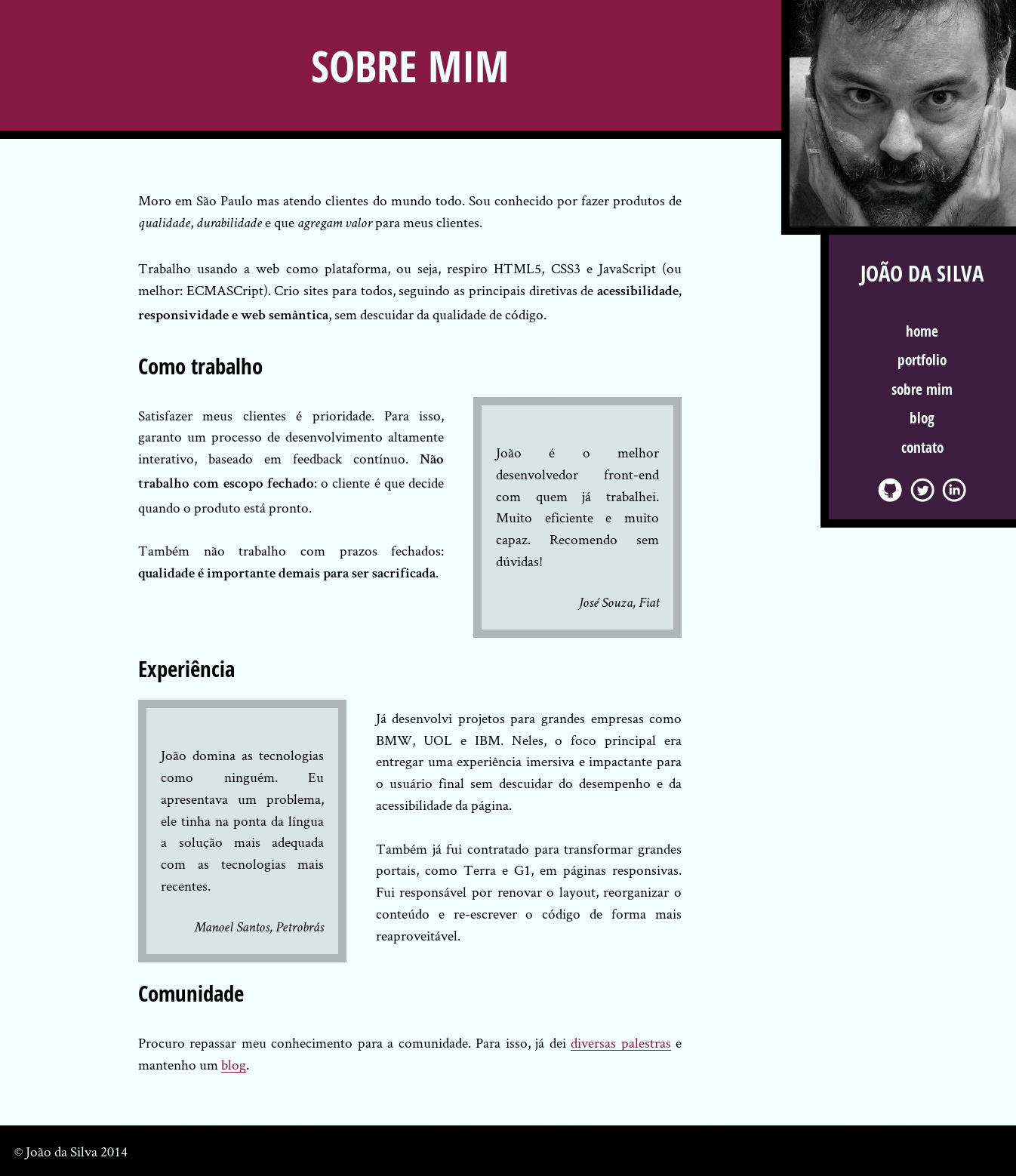
Neste curso, vamos ver diversas coisas que conseguimos fazer apenas com HTML e CSS. E como toda tecnologia pode ser bem ou mal utilizada, veremos quais cuidados devemos tomar ao desenvolver páginas para a Internet, quais boas práticas podemos seguir e como elas impactam para nossos usuários finais.

Para estudar essas tecnologias, vamos trabalhar em um projeto de um site pessoal.

## HTML

Quando pensamos em fazer uma página, pensamos em fazer isso pois queremos expor algum **conteúdo** para nossos visitantes. No nosso caso, em que estamos desenvolvendo um site pessoal, queremos divulgar nosso trabalho para o mundo, nossos conhecimentos e qualidades.

Para isso, podemos fazer uma página como a da figura abaixo, com um texto descrevendo nosso trabalho, nossa foto e alguns links para nosso visitante explorar o site.



Todo esse conteúdo e visual precisa ser descrito de alguma forma para o navegador conseguir mostrá-lo para o usuário. É aí que entram em cena o HTML e o CSS.

# Começando o site

Vamos começar a implementar o site pessoal do João da Silva. Para isso, usaremos algumas imagens já preparadas pelo designer do site e um texto escrito pelo próprio João da Silva. Baixe o pacote com esses arquivos de imagem disponibilizado pela professora.

Ao contrário do que foi indicado acima, vocês usarão apenas como base os textos do João da silva. Na verdade vocês irão construir textos próprios falando sobre cada um de vocês, assim ao final terão um site pessoal que servirá como portfólio pessoal dos trabalhos realizados por vocês.

Nosso primeiro passo vai ser **transformar o texto numa página HTML simples**. Então descompacte o arquivo que você baixou, crie um arquivo chamado bio.html, copie o texto do arquivo bio.txt do pacote para dentro desse novo arquivo e utilize as tags <h1>, <h2> e <p> para formatá-lo. Você pode, por enquanto, **tirar os dois trechos** em que clientes do João da Silva o recomendam; formataremos eles depois.

Coloque também as informações que o navegador necessita para exibir corretamente a página: as tags <meta>, <link> e <title>. Não esqueça também do DOCTYPE.

Vamos começar analisando o conteúdo: temos um título (sobre mim), dois parágrafos de texto, um subtítulo, mais texto... Num editor de texto, conseguimos selecionar o título e aplicar um tamanho de fonte maior, uma cor diferente, centralizar etc. Quando você fizer isso, seu editor de texto provavelmente gravará suas escolhas junto do texto:

Sobre mim = fonte 24pt, branco, centralizado

Moro em São Paulo... = fonte 12pt, preto, justificado

Trabalho usando a web... = fonte 12pt, preto, justificado

Como trabalho = fonte 18pt, preto, à esquerda

No caso do nosso navegador, precisamos fazer isso também: indicar que um determinado trecho do texto tem uma fonte diferente, uma cor diferente... Mas mais ainda: podemos indicar que esse primeiro trecho (sobre mim) é um título da página. Assim, podemos evitar o retrabalho de definir todo o estilo de novo para o próximo título: basta configurarmos, posteriormente, o estilo dos títulos. O mesmo vale para os parágrafos de texto, subtítulos etc.

Sobre mim = título

Moro em São Paulo... = parágrafo

Trabalho usando a web... = parágrafo

Como trabalho = subtítulo

## Tags

O HTML serve justamente para indicar **o que significa** cada trecho de texto que digitamos. Para o nosso exemplo, poderíamos colocar essas informações no texto da seguinte forma:

<h1>Sobre mim</h1>

<p>Moro em São Paulo...</p>

<p>Trabalho usando a web...</p>

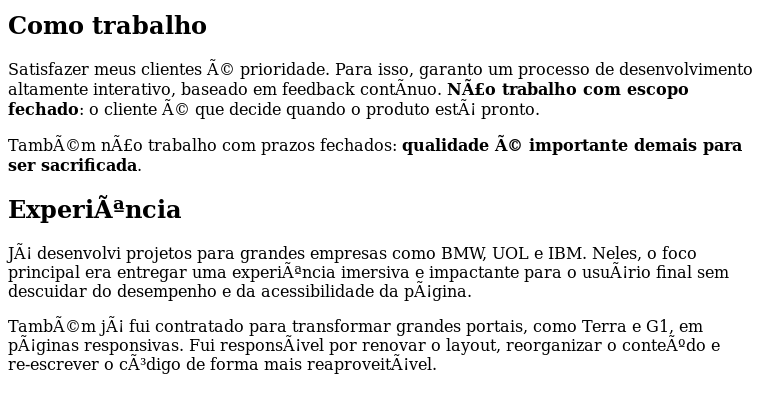
<h2>Como trabalho</h2>

Repare nas marcações que colocamos: <h1>, <h2> e <p>. Essas marcações são o que chamamos de **tags** do HTML, e são o bloco de construção fundamental da linguagem.

Repare que usamos também </h1>, </h2> e </p>. Quando colocamos essas marcações, estamos indicando o final de uma tag ou, como dizemos, **fechamos** a tag que estava aberta.

# Estrutura básica

Apesar de já podermos salvar esse código num arquivo com a extensão .html e abri-lo num navegador, possivelmente teremos problemas, como na imagem abaixo.



Isso acontece porque, apesar de termos escrito código HTML correto, o navegador não tem todas as informações necessárias para mostrar nossa página corretamente.

A informação faltante mais visível aos nossos olhos é a codificação dos caracteres. Repare que toda letra que deveria aparecer acentuada está com problema. Isso acontece porque, atualmente, existem várias formas de representar caracteres acentuados em código binário, que é como o computador guarda nosso arquivo no disco. Precisamos, então, indicar qual a codificação que estamos usando em nossa página.

Da mesma forma que usamos tags para indicar o significado de cada trecho de texto, vamos usar uma tag para indicar a codificação da página: a tag <meta>.

<meta charset="utf-8">

No exemplo acima, estamos indicando que nossa página usa a codificação de caracteres **UTF-8**. Essa é uma codificação bastante popular e completa, e recomendamos que você sempre use ela.

Uma outra informação faltando, mais sutil, é o **título** da página. Não o título "Sobre mim", que já está lá, mas um título na janela do navegador, para que nosso visitante saiba qual página está aberta naquela janela do navegador.

Página sem título e sem ícone

Novamente, vamos usar uma tag para passar essa informação para o navegador: a tag <title>.

<title>Biografia - João da Silva</title>

Uma outra informação legal de passarmos para o navegador, para ajudar ainda mais nosso visitante, é mostrar um ícone correspondente à nossa página, para facilitar a identificação da janela.



Para fazer isso, podemos colocar um arquivo chamado favicon.png na mesma pasta do nosso arquivo .html. E, para garantir que o navegador vai mostrá-la, podemos colocar mais uma tag na nossa página: a tag <link>.

<link rel="icon" href="favicon.png">

Repare que a tag <link> é um pouco mais complexa que as outras que já usamos. Ela tem dois **atributos**: rel e href. O atributo rel diz que tipo de conteúdo que queremos referenciar com ela (um ícone, no nosso caso). Já o atributo href diz onde esse conteúdo se encontra. Usaremos novamente essa tag mais para a frente.

Com isso, nosso HTML já fica mais ou menos assim:

<meta charset="utf-8">

<title>Biografia - João da Silva</title>

<link rel="icon" href="favicon.png">

<h1>Sobre mim</h1>

<p>Moro em São Paulo...</p>

<p>Trabalho usando a web...</p>

<h2>Como trabalho</h2>

Meio bagunçado, não? As informações que são necessárias para o navegador estão misturadas com as informações que queremos mostrar para o visitante do site. Para organizar melhor essas informações, existem duas tags fundamentais da linguagem HTML: <head> e <body>.

A tag <head> serve para agrupar informações para o navegador entender melhor nossa página, como o título da página, o ícone e a codificação. Em outras palavras, as meta-informações da página devem ficar dentro dessa tag.

Já a tag <body> serve para agrupar as informações que queremos mostrar para os usuários, ou seja, o conteúdo que queríamos mostrar originalmente.

Usando essas duas tags, podemos reorganizar nosso código da seguinte forma:

<head>

<meta charset="utf-8">

<title>Biografia - João da Silva</title>

<link rel="icon" href="favicon.png">

</head>

<body>

<h1>Sobre mim</h1>

<p>Moro em São Paulo...</p>

<p>Trabalho usando a web...</p>

<h2>Como trabalho</h2>

</body>

Repare um detalhe no código escrito acima: as tags que estão dentro das tags <head> e <body> foram escritas com alguns espaços à frente. Escrever código assim é uma boa prática, conhecida como **indentação**. Ela nos ajuda a enxergar melhor a estrutura da nossa página e vai ser bem importante conforme nossa página vai ficando mais complexa.

O código fica melhor organizado usando as tags <head> e <body>, mas ainda falta uma tag essencial para embrulhar toda a nossa página e indicar para o navegador que estamos escrevendo código HTML em nosso arquivo: a tag <html>. Colocando-a no nosso código, chegamos ao seguinte resultado:

<html>

<head>

<meta charset="utf-8">

<title>Biografia - João da Silva</title>

<link rel="icon" href="favicon.png">

</head>

<body>

<h1>Sobre mim</h1>

<p>Moro em São Paulo...</p>

<p>Trabalho usando a web...</p>

<h2>Como trabalho</h2>

</body>

</html>

Pronto, agora o navegador sabe que estamos usando HTML. Mas estamos esquecendo de um detalhe: o HTML é uma linguagem já com uma certa idade, que passou por reformulações para se aperfeiçoar e ficar mais poderosa. Assim, não basta dizer para o navegador que estamos usando HTML; precisamos dizer **qual versão do HTML** estamos usando. É aí que entra uma instrução especial para o navegador, que parece uma tag mas não é: o **DOCTYPE**.

Como o nome diz, essa instrução serve para dizer para o navegador (ou qualquer programa que queira ler nosso arquivo) qual tipo de documento ele está lendo. Ou seja, essa instrução não é exclusiva da linguagem HTML, mas vem de uma linguagem mais genérica chamada SGML (do inglês Standard Generalized Markup Language). Sendo assim, nessa instrução vamos dizer para o navegador que estamos usando HTML e qual versão do HTML estamos usando. Mas qual versão estamos usando?

Neste curso, vamos estudar a versão mais recente do HTML, que é a versão 5. Mais para a frente, vamos estudar as diferenças dessa versão para as anteriores e por que devemos usá-la atualmente.

Precisamos, então, colocar a instrução DOCTYPE correspondente ao HTML 5 no nosso código HTML. Então colocamos **no início do documento** o seguinte código:

<!DOCTYPE html>

E pronto! Agora o navegador consegue entender corretamente nosso código HTML. Repare que não há nada nesse código que diga que é a versão 5 do HTML que estamos usando; isso é proposital. A organização que define como deve ser a linguagem HTML - a **W3C** - quis facilitar a nossa vida e decidiu criar o DOCTYPE acima, que diz para o navegador que estamos usando **a última versão do HTML disponível**. Assim, quando surgir uma nova versão do HTML, nossa página não precisa ser alterada.

Resumindo, nosso código, no final, vai ficar assim:

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Biografia - João da Silva</title>

<link rel="icon" href="favicon.png">

</head>

<body>

<h1>Sobre mim</h1>

<p>Moro em São Paulo...</p>

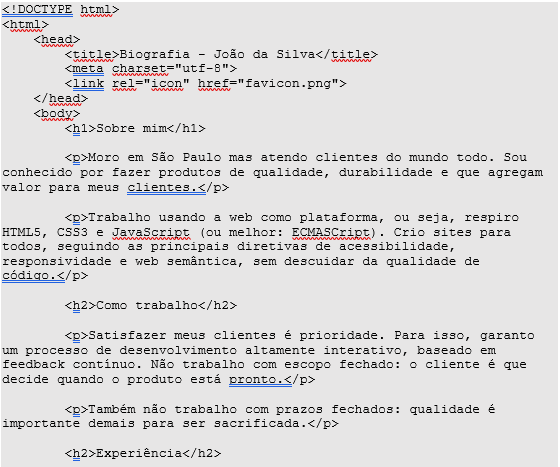
<p>Trabalho usando a web...</p>

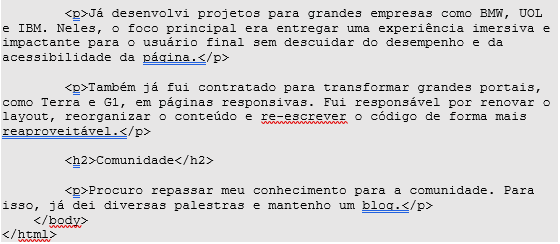
<h2>Como trabalho</h2>

</body>

</html>

Essa estrutura, com DOCTYPE e tag <html> com <head> e <body>, deve ser seguida por todo documento HTML para que os navegadores interpretem-no corretamente.





**Só não esqueça que ao invés de falar do João, você irá falar de si mesmo.**

## Realce de texto

Na página que estamos desenvolvendo, temos bastante texto escrito. Com tanto texto, talvez o usuário fique meio perdido, por isso é importante *realçar* alguns termos que merecem mais destaque.

No primeiro parágrafo, por exemplo, temos o seguinte texto:

Moro em São Paulo mas atendo clientes do mundo todo. Sou conhecido por fazer produtos de qualidade, durabilidade e que agregam valor para meus clientes.

Aqui, seria interessante dar destaque para os termos "qualidade", "durabilidade" e "agregam valor", deixando-os em *itálico*, por exemplo. Para fazer isso, usamos a tag <em>:

Sou conhecido por fazer produtos de <em>qualidade</em>

Neste outro parágrafo

Satisfazer meus clientes é prioridade. Para isso, garanto um processo de desenvolvimento altamente interativo, baseado em feedback contínuo. Não trabalho com escopo fechado: o cliente é que decide quando o produto está pronto.

É importante ressaltar aqui a parte "Não trabalho com escopo fechado", com ainda mais destaque que os termos anteriores, deixando-os em negrito. Para isso, podemos usar a tag <strong>:

<p><strong>Não trabalho com escopo fechado</strong>: o cliente é que decide quando o produto está pronto</p>

Aplique essas duas tags em seu código, neste e em outros lugares que achar relevante.

Obs.: Como vocês estarão falando de si mesmos coloquem o diferencial de cada um, para fazer esse realce de texto, por exemplo: sei trabalhar em equipe, sou muito criativo para criação de Layouts, tenho muita facilidade em desenvolver textos, trabalho com back end e front end, trabalho com banco de dados, conhecimento avançado em Word e Excel, etc...

Com isso, já conseguimos representar o conteúdo básico da página. Mas repare que algumas partes do texto são mais importantes, têm destaque. Novamente, para indicar que uma parte do texto tem uma característica especial, vamos colocar uma tag ao redor dela:

<p>Moro em São Paulo mas atendo clientes do mundo todo. Sou conhecido por fazer produtos de <em>qualidade</em>, <em>durabilidade</em> e que <em>agregam

valor</em> para meus clientes.</p>

Repare que, para dar uma ênfase ao texto, usamos a tag <em>. Em outro trecho, onde temos um texto com um destaque **mais forte**, podemos usar a tag <strong>:

<p>... Crio sites para todos, seguindo as principais diretivas de

<strong>acessibilidade</strong>, <strong>responsividade</strong> e <strong>web semântica</strong>, sem descuidar da qualidade de código.</p>

Em ambos os casos, veja que colocamos **tags dentro de tags**. Esse é um recurso que vamos explorar bastante ao longo do curso para poder detalhar bem as informações da nossa página.

# Estilo

Com o código HTML, conseguimos definir **o quê** vamos mostrar, mas não **como** vamos mostrar. Esse é um papel que cabe ao CSS, tecnologia que vamos abordar no próximo capítulo.

O site do João da Silva será composto de diversas páginas, como visto em aula. Vamos começar também a página do blog. Crie o arquivo blog.html. Nele, deve haver o seguinte texto: (use esse texto como exemplo, porém depois você irá criar uma página de blog com algum assunto de tecnologia que vocês goste, algo que você se identifique)

Blog

**O essencial de design responsivo**

Design responsivo tem ganho cada vez mais atenção como técnica de desenvolvimento de páginas web para dispositivos móveis. Mas como começar?

**Por que fazer páginas acessíveis?**

Tem se falado cada vez sobre acessibilidade na web. Há diversas técnicas e diretivas a serem seguidas, inclusive da própria W3C. Mas vale a pena investir nisso?

**JavaScript não obstrusivo**

JavaScript é uma linguagem essencial hoje em dia para criar páginas ricas, interativas e dinâmicas. Porém, há alguns cuidados que devem ser tomados ao colocarmos código JavaScript em nossas páginas.

# Responda as perguntas abaixo:

1. Que tipo de linguagem é o HTML? Ou, em outras palavras, qual o propósito da linguagem HTML?
2. Linguagem de programação, ou seja, serve para ensinar o computador a fazer algo
3. Linguagem de consulta, ou seja, serve para especificar informações às quais queremos ter acesso
4. Linguagem de estilização, ou seja, serve para definir a aparência de um conteúdo
5. Linguagem de marcação, ou seja, serve para indicar o que cada trecho de um documento representa

A linguagem HTML serve para **marcar conteúdo**, ou seja, especificar o que cada letra, palavra e frase **significa**. Usamos ela para criar páginas pois, assim, o navegador e outras ferramentas que trabalham na Internet conseguem analisar e processar nosso conteúdo, apresentando-o de forma adequada para os usuários.

1. Qual a estrutura básica de uma página HTML?
2. tag html com DOCTYPE, body e head dentro, nessa ordem
3. DOCTYPE e tag html com tags body e head dentro, nessa ordem
4. Tag html com DOCTYPE, head e body dentro, nessa ordem
5. DOCTYPE e tag html com tags head e body dentro, nessa ordem
6. Tag html com DOCTYPE e tags body, header e footer dentro, nessa ordem
7. Tag html com DOCTYPE e tags header, body e footer dentro, nessa ordem
8. DOCTYPE e tag html com tags header, body e footer dentro, nessa ordem

<!DOCTYPE html>

<html>

<head>

tags com informações para o navegador

</head>

<body>

conteúdo para o usuário

</body>

</html>

1. Qual a função da instrução DOCTYPE em um arquivo HTML?
2. Nenhuma, colocamos apenas para garantir compatibilidade
3. Ativar o processamento do estilo da página
4. Indicar a codificação usada no texto do arquivo
5. Indicar que estamos usando a linguagem HTML e qual versão da linguagem estamos usando

A instrução DOCTYPE serve para indicar qual linguagem estamos usando no nosso documento e em qual versão. Por isso é essencial para que o computador processe corretamente nosso arquivo.

Atualmente, usamos a instrução <!DOCTYPE html> para indicar que usamos a última versão do HTML. Antigamente, era necessário usar <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">, por exemplo, para indicar que estávamos usando a versão 4 do HTML na sua variante *estrita*.

1. Qual a codificação mais recomendada para escrevermos páginas para a Web? Por que?
2. UTF-8, por ser uma codificação mais simples para o computador entender
3. ISO-8859-1, por ser a codificação usada nos textos em português normalmente
4. -8859-1, por ser uma codificação abrangente e bem suportada
5. UTF-8, por ser uma codificação abrangente e bem suportada

A codificação UTF-8 abrange praticamente todos os sistemas de escritas usadas pelo mundo todo. Além disso, todos os computadores vêm com suporte a essa codificação. Assim, se queremos que nossas páginas sejam acessíveis por todos, a codificação UTF-8 é a melhor opção.

1. Qual a diferença entre a tag <h1> e a tag <title>? Qual o propósito de cada uma?
2. A tag <title> serve para passar para o navegador qual o título da aba/janela em que a página será exibida. Já a tag <h1> serve para mostrar para o usuário qual o título principal do conteúdo da página.
3. A tag <h1> serve para passar para o navegador qual o título da aba/janela em que a página será exibida. Já a tag <title> serve para mostrar para o usuário qual o título principal do conteúdo da página.
4. Nenhum, ambas são usadas para falar que é um título