**CSS – CONTENT**

CSS (Cascading Style Sheets) é a linguagem utilizada para descrever o layout dos sites. Embora seja mais comumente utilizada no desenvolvimento de páginas web e interfaces de utilizador de aplicações online, qualquer tipo de documento XML pode utilizar CSS. No início, essa função era desempenhada pelo próprio HTML. À medida que os sites se tornaram cada vez mais complexos, o código tornou-se altamente ilegível e complicado para manutenção. Então, ficou claro que algum novo método deveria ser introduzido para resolver esse problema. A solução surgiu em meados da década de 1990 na forma de CSS, então chamado de CHSS.

Existem várias versões de CSS. Cada uma delas foi atualizada em relação à anterior e proporcionou novas oportunidades. Até agora, foram lançadas as versões CSS1, CSS2, CSS3 e CSS4. A versão CSS4 não existe por si só, mas há vários módulos do quarto nível, por isso é mencionada quando se fala sobre versões CSS.

Na maioria dos casos, o código CSS é encontrado em um documento separado que é incluído quando necessário. Esse também é o melhor método de trabalhar com CSS. Também é possível usar CSS diretamente em um documento cujo design é alterado. Isso pode ser feito no cabeçalho do documento ou na tag do elemento cuja aparência é afetada.

Embora os benefícios de usar CSS sejam inquestionáveis, ele também tem algumas limitações. Diferentes navegadores da Internet interpretam o código de maneiras diferentes, por isso é necessário usar métodos diferentes para obter consistência na aparência.

Exemplo de definição da fonte da classe “text”:

***Traduzido com a versão gratuita do tradutor -*** [***DeepL.com***](https://www.deepl.com/?utm_campaign=product&utm_source=web_translator&utm_medium=web&utm_content=copy_free_translation)

**HTML – CONTENT**

A chamada linguagem para marcação de hipertexto, ou seja, HTML (Hyper Text Markup Language), é usada principalmente quando é necessário criar uma página web específica com o uso de diferentes editores. E os editores HTML, além do Adobe Cold Fusion, incluem FrontPage e Dreamweaver, Microsoft Visual Basic Developer e outros.

A versão atual é identificada como HTML 4.01, e a padronização é tratada pelo W3C, ou seja, o World Wide Web Consortium.

Para entender claramente a definição deste termo, deve-se primeiro definir os chamados hipertextos. Basicamente, trata-se de um tipo especial de conteúdo que, além de palavras específicas, certamente contém uma imagem ou um vídeo, ou uma faixa de áudio. E, a este respeito, é bom notar que o HTML é, na verdade, um conjunto menor de SGML, que é a abreviatura de Standard General Markup Language. Ou seja, este tipo de linguagem é diretamente responsável por determinar a aparência de todas as páginas que se encontram em vários documentos da World Wide Web e também é aplicado em situações em que é necessário estabelecer uma ligação entre documentos.

A definição comum do termo HTML é: HTML é uma linguagem padronizada, usada para definir a estrutura de textos, mídia e objetos incorporados em páginas da web e e-mails. Muitas vezes, ouve-se dizer que os caracteres HTML, ou seja, as chamadas tags, também são chamados de códigos, mas geralmente esse uso da palavra é incorreto. A razão para isso está no facto de que a definição de um código de computador é bastante diferente e que as tags HTML devem marcar a estrutura de um conteúdo específico, com o objetivo de que o navegador da Internet, ou seja, o Agente do Utilizador, possa primeiro reconhecer a estrutura específica e, imediatamente depois, exibi-la da maneira prevista e no local predefinido, ou seja, exatamente na janela do navegador da web que está a ser utilizada no momento.

Além do HTML, também é utilizado CSS (Cascading Style Sheets), bem como inúmeras linguagens de script, como, por exemplo, JavaScript ou VBScript, e DOM (Document Object Model). O objetivo de aplicar todos estes, mas também muitos outros acréscimos, é melhorar um determinado site, ou seja, modificá-lo para que fique o melhor possível.

Da mesma forma, além do HTML, também são utilizados PHP (Hypertext Preprocessor) e ASP (Active Server Pages), para que a comunicação entre um determinado site e o navegador seja a melhor e mais rápida possível.

A aplicação básica desta linguagem encontra-se na situação em que é necessário separar elementos específicos numa página, pelo que, utilizando tags HTML, é possível definir com precisão onde o título deve estar localizado, por exemplo, mas também todos os outros itens que esta página contém. E dentro dos padrões que a linguagem mencionada utiliza, já existem elementos implementados cujo objetivo principal é descrever as características de uma página específica, tais como palavras-chave ou descrição do conteúdo, ou seja, informações sobre o autor e outros. Todas essas informações se aplicam aos chamados metadados e são definitivamente separadas com muita precisão do conteúdo da própria página, ou seja, do documento HTML.

A criação de um documento HTML específico definitivamente começa com a definição do seu tipo, ou seja, com a determinação do DTD (Document Type Definition), que é determinado com base no padrão usado nessa ocasião para a sua criação.

Quanto às tags HTML, elas são usadas principalmente para deixar claro para o navegador de que forma um conteúdo específico deve ser visível. Normalmente, essas tags são colocadas no início e no final de um documento específico e são exibidas entre estes dois símbolos < >. E entre estas duas tags, é claro, há tanto o conteúdo de uma página específica, ou seja, o documento, quanto todas as outras tags que ele contém. A característica destas tags é que elas não diferenciam entre maiúsculas e minúsculas, ou seja, não são sensíveis a elas, como é o caso, por exemplo, da URL. A tag de cabeçalho é a chamada tag head, e essa parte certamente não será exibida ao utilizador no ecrã, mas é usada principalmente para definir com precisão as propriedades de um documento específico.

Ao determinar o tamanho das letras num caso específico, é utilizada a chamada tag de cabeçalho, e o tamanho da fonte é determinado com base em 6 níveis disponíveis, ou seja, de h1 a h6.

*Traduzido com a versão gratuita do tradutor -* [*DeepL.com*](https://www.deepl.com/?utm_campaign=product&utm_source=web_translator&utm_medium=web&utm_content=copy_free_translation)

**React JS – CONTENT**

React JS é uma biblioteca JavaScript popular para a criação de interfaces de utilizador. Foi desenvolvida pelo Facebook e lançada em 2013 como um projeto de código aberto. Desde então, o React ganhou imensa popularidade entre os programadores devido à sua eficiência, flexibilidade e facilidade de utilização.

O conceito central por trás do React é a arquitetura baseada em componentes. Isso significa que a interface do utilizador é dividida em pequenos componentes reutilizáveis, tornando-a mais fácil de gerir e manter. O React também usa um DOM (Document Object Model) virtual, que permite uma renderização mais rápida e melhor desempenho.

O React não se limita apenas à criação de aplicações web, ele também pode ser usado para desenvolver aplicações móveis com o React Native. Isso permite que os programadores usem a mesma base de código para aplicações web e móveis, economizando tempo e esforço.

O React JS é usado principalmente para criar interfaces de utilizador, mas também pode ser usado para desenvolver aplicações móveis com o React Native. Alguns sites e empresas populares que usam o React incluem Facebook, Instagram, Netflix, Airbnb e Dropbox.

Porquê usar o React Js?

Existem várias razões pelas quais os programadores escolhem o React JS para construir os seus projetos. Algumas das principais vantagens incluem:

Eficiência: O uso de um DOM virtual e fluxo de dados unidirecional torna o React altamente eficiente, resultando em velocidades de renderização mais rápidas e melhor desempenho.

Flexibilidade: O React pode ser usado tanto para desenvolvimento web como móvel, tornando-o uma escolha popular entre os programadores que preferem trabalhar com uma única linguagem para várias plataformas.

Reutilização: com a sua arquitetura baseada em componentes, o React permite a criação de componentes de interface do utilizador reutilizáveis, poupando tempo e esforço no desenvolvimento.

Comunidade ativa: como mencionado anteriormente, o React tem uma comunidade grande e ativa que oferece suporte e recursos para programadores. Isso torna mais fácil encontrar ajuda ou soluções ao enfrentar desafios ao trabalhar com o React.

**JavaScript – CONTENT**

JavaScript é uma linguagem de script interpretada extremamente popular que foi a linguagem mais procurada em 2023, com 29% das ofertas de emprego exigindo experiência em programação com JavaScript ou TypeScript. Existem várias razões para o sucesso do JavaScript, começando pelo facto de ser um padrão aberto, não controlado por nenhum fornecedor específico, com inúmeras implementações e uma sintaxe fácil de aprender.

Hoje em dia, praticamente tudo o que é interativo ou animado numa página web é renderizado em JavaScript. Graças a frameworks do lado do servidor, como Node.js, Deno e bun, a linguagem também se estende para além do navegador. É usada para escrever código para tudo, desde clientes a servidores e aplicações baseadas na nuvem.

Esta é uma visão geral do JavaScript, incluindo uma breve história, para que tipos de aplicações é utilizado, ferramentas para trabalhar com JavaScript e recursos para aprender JavaScript.

Uma breve história do JavaScript

Como o próprio nome indica, o JavaScript é uma linguagem de script. As linguagens tradicionais, como C++, são compiladas antes de serem executadas na forma binária executável, com o compilador a verificar se há erros em todo o programa antes de o processo ser concluído. As linguagens de script, por outro lado, são executadas uma linha de cada vez por outro programa chamado interpretador. As linguagens de script começaram como uma série de comandos shell usados para executar outros programas. A sua flexibilidade e facilidade de uso tornaram-nas um tipo popular de linguagem de programação por si só, e linguagens como JavaScript tornaram-se importantes com o surgimento da web.

Muitos programadores novos em JavaScript questionam-se sobre a parte Java do seu nome. Resumidamente, Java e JavaScript surgiram durante o mesmo momento histórico e ambos foram fundamentais para o nascimento da web. A sua história está interligada, mas as duas linguagens são separadas. Brendan Eich, criador do JavaScript, explicou por que o JavaScript era considerado um complemento importante para o Java no navegador Netscape original:

Havia pessoas que argumentavam veementemente que o Java era bom para programadores que criavam componentes, mas há um público muito maior de pessoas que escrevem scripts ou talvez copiem um script de outra pessoa e o ajustem. Essas pessoas são menos especializadas e podem ser pagas para fazer algo além de programação, como administrar uma rede, e escrevem scripts em tempo parcial ou como atividade secundária. Se estão a escrever pequenos trechos de código, elas só querem fazer o seu código com o mínimo de complicações.

Originalmente chamado LiveScript, a linguagem foi criada com uma sintaxe superficialmente semelhante à do Java, a fim de explorar a comunidade emergente de programadores Java. Embora o seu uso tenha crescido em paralelo com o Java e o desenvolvimento inicial da web, o JavaScript foi considerado por muito tempo um coadjuvante inferior à linguagem de programação Java, mais robusta.

Em 1997, a ECMA (European Computer Manufacturers Association) emitiu uma norma que definia a linguagem que qualquer pessoa poderia implementar. A razão para esta medida foi que o «Java» continuava a ser uma marca registada da Sun, cuja licença de utilização pertencia apenas à Netscape. A nova versão padronizada do JavaScript foi apelidada de ECMAScript, mas, na prática, a maioria das pessoas conhece a linguagem como JavaScript.

Para que serve o JavaScript?

O JavaScript foi desenvolvido para ajudar a criar sites interativos, e essa continua a ser uma das suas principais utilizações atualmente. Os programadores incorporam o JavaScript no código dos seus sites, onde é executado pelo interpretador integrado no navegador do utilizador. O código JavaScript cria ou modifica HTML em tempo real para produzir conteúdo dinâmico e interagir com armazenamentos de dados back-end. Quando usado dessa forma, o JavaScript é chamado de linguagem do lado do cliente, sendo uma de suas aplicações mais lucrativas os jogos para dispositivos móveis. Estruturas do lado do servidor, como Node.js, estendem o JavaScript para aplicações mais escaláveis e orientadas a eventos.

Perguntas de entrevista sobre JavaScript

Como o JavaScript é tão popular, os gerentes de contratação de tecnologia normalmente esperam que os candidatos saibam como usá-lo. Você pode até ser solicitado a escrever um programa rápido em JavaScript como parte de uma entrevista. Felizmente, pode encontrar muitas listas de exemplos de perguntas de entrevista online — a Toptal tem uma boa, por exemplo. Também recomendamos este breve tutorial de JavaScript de Naren Yellavula no dev.bits(). Ele se concentra no que precisa saber para poder responder a perguntas típicas de entrevista.

Vantagens de usar JavaScript

Ninguém dirá que o JavaScript é uma linguagem perfeita, mas é muito útil. É fácil escrever código funcional rapidamente em JavaScript e, embora as linguagens interpretadas muitas vezes não tenham um bom desempenho em comparação com o código compilado, o JavaScript tem uma vantagem especial aqui, pois foi inicialmente criado para navegadores da web. Três das maiores empresas de software do mundo — Microsoft, Apple e Google — competem para produzir interpretadores JavaScript de alto desempenho. Como resultado, o JavaScript é uma linguagem notavelmente robusta e escalável, mesmo considerando suas origens humildes.