

Curso lógica de programação

CODE - X



Lógica de programação - CSS



PALAVRA DO ORIENTADOR

Olá! Sou formado nas áreas técnicas de Química, Meio Ambiente e Tecnologia da Informação, além de possuir graduação em Licenciatura em Matemática. Minha paixão pelo conhecimento e pelo ensino sempre me motivou a buscar a excelência nas minhas áreas de atuação. Ao longo da minha jornada, tive a vantagem de acumular experiências valiosas, tanto no campo profissional quanto acadêmico, e acredito que o aprendizado é um processo contínuo e enriquecedor.

É com grande satisfação que ofereço apoio e orientação a você, que está começando ou se aprofundando em sua carreira e nos estudos. Compreendo os desafios e as conquistas desta fase e fico muito feliz por poder compartilhar um pouco da minha trajetória e conhecimento.

Sinta-se à vontade para contar comigo durante essa jornada intensa de estudos e trabalho. Estou aqui para ajudar no que for necessário, oferecendo orientação, apoio e incentivo. Juntos, podemos transformar desafios em oportunidades de crescimento e alcançar grandes objetivos. Conte sempre comigo!

SUMÁRIO

Introdução.....	06
1. Seletores.....	08
2. Propriedades e Valores	09
3. Cascata e Especificidade.....	09
4. Unidades e Medidas.....	10
5. Modelo de Caixa (Box Model)	11
6. Layouts e Fluxo de Documentos.....	12
7. Animações e Transições	12
Referências	13

FUNDAMENTOS DE LÓGICA DE PROGRAMAÇÃO

Olá, estudante! Seja muito bem-vindo ao fascinante universo da programação! Se você chegou até aqui, é provável que já tenha ouvido falar sobre códigos, softwares, linguagens de programação e, especialmente, sobre lógica de programação. Talvez você esteja se perguntando: “Por onde eu começo?”. Não se preocupe, essa é uma dúvida comum, e estamos aqui para guiá-lo nos primeiros passos dessa jornada. Neste módulo, nosso foco será explorar os fundamentos da lógica de programação aplicada ao CSS.

O CSS (Cascading Style Sheets) é uma linguagem que dá vida e personalidade às páginas web, transformando estruturas estáticas em experiências visuais atrativas e dinâmicas. Com ele, você será capaz de controlar a aparência de um site: núcleos, fontes, posicionamento de elementos, animações, transições e muito mais. Ao entender a lógica por trás do CSS, você conseguirá planejar e organizar estilos de maneira eficiente, otimizando o design para que ele não seja apenas bonito, mas também funcional e acessível.

Nesta unidade, você conhecerá os fundamentos que fazem do CSS uma ferramenta essencial para qualquer programador web. Desde a compreensão das regras básicas de sintaxe até a aplicação de conceitos avançados como layouts responsivos e animações, você aprenderá a manipular o visual das páginas da web, tornando-o tão interessante e adaptável a diferentes dispositivos e tamanhos de tela.

FUNDAMENTOS DE LÓGICA DE PROGRAMAÇÃO

Dominar a lógica de programação no CSS não é apenas sobre aprender uma nova linguagem, mas sobre desenvolver um olhar crítico e uma abordagem estratégica para a estética digital. Este conjunto de conhecimentos permitirá que você enfrente os desafios do mercado de trabalho com segurança, compreendendo os princípios do design web e as melhores práticas para criar experiências visuais memoráveis. Além disso, você poderá contribuir de forma significativa para o avanço tecnológico, ajudando a construir uma web visualmente impactante e acessível para todos.

Vamos juntos explorar o universo de núcleos, formas e estilos que o CSS proporciona, aprendendo a construir layouts criativos e a dominar a arte de dar vida a sites e aplicações! A partir de agora, cada linha de código será um passo a mais rumo à criação de páginas web modernas, responsivas e interessantes. Prepare-se para embarcar nessa jornada, onde a lógica de programação e a criatividade se encontram para transformar o virtual em algo incrível e tangível!

Introdução

A lógica de programação no CSS (Cascading Style Sheets) é crucial para qualquer desenvolvedor web que deseje criar páginas visualmente atraentes e funcionais. CSS é uma linguagem que dá estilo e forma ao conteúdo estruturado pelo HTML, desenvolvendo textos e elementos simples em interfaces sofisticadas, dinâmicas e responsivas. No entanto, entender a lógica por trás do CSS vai muito além de escolher núcleos e fontes; é sobre aplicar uma abordagem lógica e organizada para controlar a aparência e a disposição de cada elemento de uma página web.

A importância da lógica de programação no CSS reside na capacidade de transformar um layout simples em algo visualmente impactante, mantendo sempre a coerência e a legibilidade do código. Saber como organizar e aplicar estilos de forma lógica permite que o desenvolvedor atualize um código CSS limpo, fácil de entender e de atualizar. Isso é fundamental, especialmente em projetos grandes, onde diferentes partes do CSS podem se sobrepor ou interagir de maneiras inesperadas.

Para quem está iniciando no mundo do desenvolvimento web, aprender a lógica de programação no CSS significa entender como as regras e propriedades de interação entre si. A compreensão das cascatas e da herança de estilos é essencial para controlar a forma como cada elemento aparece na página.

Para quem já possui conhecimento, o aprofundamento na lógica do CSS abre portas para a criação de layouts complexos, animações e efeitos que tornam uma navegação mais envolvente e profissional.

Além disso, a lógica do CSS é essencial para garantir que as páginas sejam responsivas e acessíveis em dispositivos de diferentes tamanhos e formatos. Compreender como construir um design flexível e adaptável não só melhora a experiência do usuário, mas também torna o site mais eficiente e acessível para todos.

Em suma, uma lógica de programação no CSS é a arte de aplicar regras estéticas de maneira meticulosa e estratégica, garantindo que o visual de um site não seja apenas bonito, mas também funcional e otimizado. É um componente indispensável para qualquer desenvolvedor que deseje criar uma web visualmente coesa e técnica, elevando a qualidade e a usabilidade de cada projeto digital.



FUNDAMENTOS DE LÓGICA DE PROGRAMAÇÃO - CSS

1. Seletores

Os seletores são a base da lógica do CSS, pois definem quais elementos HTML receberão as regras de estilo.

- Seletores de Tipo:
 - Ex: h1, p, div
 - Estilizam todos os elementos de um determinado tipo.
- Seletores de Classe:
 - Ex: .minha-classe
 - Usados para estilizar grupos de elementos que compartilham a mesma classe. Você pode aplicar a mesma classe a múltiplos elementos.
- Seletores de ID:
 - Ex: #meu-id
 - Usados para estilizar um único elemento, pois cada ID deve ser único em uma página.
- Seletores Atributo:
 - Ex: [type="text"]
 - Selecionam elementos com atributos específicos.
- Seletores Combinados:
 - Ex: div p (seleciona todos os parágrafos dentro de um div)
 - Combinam diferentes seletores para maior especificidade.

FUNDAMENTOS DE LÓGICA DE PROGRAMAÇÃO - CSS

2. Propriedades e Valores

Cada regra CSS consiste em uma propriedade e um valor.

- Propriedades:
 - Ex: color, font-size, margin
 - Definem qual aspecto visual será alterado.
- Valores:
 - Ex: red, 16px, 10px
 - Especificam como a propriedade será aplicada.

Exemplo:

```
p {  
    color: blue; /* Propriedade: color | Valor: blue */  
    font-size: 16px; /* Propriedade: font-size | Valor: 16px */  
}
```

3. Cascata e Especificidade

O CSS aplica regras em uma ordem específica, onde a cascata determina qual regra prevalece.

- Cascata:
 - As regras são aplicadas em uma ordem hierárquica: inline > ID > classe > tipo.
- Especificidade:
 - Cada tipo de seletor tem um valor de especificidade. Quando duas regras competem, a regra com a maior especificidade é aplicada.

FUNDAMENTOS DE LÓGICA DE PROGRAMAÇÃO - CSS

Cálculo de Especificidade:

- ID: 100
- Classe: 10
- Tipo: 1

Exemplo:

```
#meu-id {  
    color: green; /* Especificidade: 100 */  
}  
  
.minha-classe {  
    color: red; /* Especificidade: 10 */  
}  
  
div {  
    color: blue; /* Especificidade: 1 */  
}
```

Se um div tiver ambos ID e classe, o texto será verde.

4. Unidades e Medidas

O CSS oferece várias unidades de medida para definir tamanhos e espaçamentos.

- Unidades Absolutas:
 - Ex: px, cm, in
 - Tamanhos fixos que não mudam com a tela.
- Unidades Relativas:
 - Ex: %, em, rem, vw, vh
 - Adaptam-se ao tamanho do elemento pai ou da viewport, permitindo layouts responsivos.

FUNDAMENTOS DE LÓGICA DE PROGRAMAÇÃO - CSS

Exemplo:

```
.container {  
    width: 80%; /* Largura relativa */  
}  
  
.texto {  
    font-size: 1.5em; /* Largura relativa ao tamanho da fonte do pai */  
}
```

5. Modelo de Caixa (Box Model)

O modelo de caixa é um conceito central que define como os elementos são representados na tela.

- Componentes:
 - Conteúdo: A parte interna onde o texto e as imagens aparecem.
 - Padding: Espaço entre o conteúdo e a borda (interior).
 - Borda: A linha que envolve o padding.
 - Margin: Espaço fora da borda (exterior).

Exemplo:

```
.box {  
    width: 200px;  
    padding: 10px;  
    border: 5px solid black;  
    margin: 20px;  
}
```

Aqui, a largura total da .box será 200px (largura) + 20px (padding) + 10px (borda) + 20px (margin).

FUNDAMENTOS DE LÓGICA DE PROGRAMAÇÃO - CSS

6. Layouts e Fluxo de Documentos

CSS oferece ferramentas para criar layouts flexíveis e responsivos.

- Flexbox:
 - Um modelo de layout unidimensional que facilita o alinhamento e distribuição de espaço entre elementos.
- Grid:
 - Um modelo de layout bidimensional que permite criar layouts complexos com linhas e colunas.

Exemplo de Flexbox:

```
.container {  
    display: flex;  
    justify-content: space-between;  
}
```

7. Animações e Transições

O CSS permite criar animações e transições para melhorar a interatividade.

- Transições:
 - Permitem que propriedades mudem suavemente de um valor para outro ao longo do tempo.

Exemplo:

```
.button {  
    transition: background-color 0.5s;  
}  
  
.button:hover {  
    background-color: blue; /* A cor muda suavemente ao passar o mouse */  
}
```

FUNDAMENTOS DE LÓGICA DE PROGRAMAÇÃO - CSS

- Animações:
 - Permitem definir sequências de mudanças de estilo.

Exemplo:

```
@keyframes exemplo {  
    from { opacity: 0; }  
    to { opacity: 1; }  
}  
  
.elemento {  
    animation: exemplo 2s; /* A animação ocorre em 2 segundos */  
}
```

REFERÊNCIAS

W3C. Cascading Style Sheets, Level 2 Revision 1 (CSS 2.1) Specification. 2011. Disponível em: <https://www.w3.org/TR/CSS21/>. Acesso em: 24 out. 2024.

MURPHY, T. CSS: The Definitive Guide. 4. ed. Sebastopol: O'Reilly Media, 2017.

RAGAN, S. Learning CSS. 2. ed. Indianapolis: Wiley, 2020.

MCDONALD, J.; DICKEY, B. CSS for Dummies. 2. ed. Hoboken: Wiley Publishing, 2011.

FREEMAN, A.; ROBERTS, J. CSS: The Missing Manual. 3. ed. Sebastopol: O'Reilly Media, 2018.

LAYFETT, M. Responsive Web Design with HTML5 and CSS. 3. ed. Boston: Cengage Learning, 2018.