

Material Didático: Introdução à Lógica de Programação

Conceitos Fundamentais

Antes de iniciarmos o desenvolvimento de qualquer aplicação, seja para web ou dispositivos móveis, é essencial dominar a **lógica de programação**. Isso significa entender a sequência lógica de instruções que levará à solução de um problema ou à execução de uma tarefa. A lógica de programação é o ponto de partida para a criação de sistemas robustos, pois possibilita que você organize o pensamento e implemente soluções eficientes.

Lógica de Programação

- **Lógica:** Colocar ordem no pensamento, ou seja, organizar ideias para desenvolver uma solução.
- **Lógica de Programação:** Técnica que define a sequência lógica de instruções até atingir a solução de um problema.
- **Sequência Lógica:** Conjunto de passos necessários para alcançar um objetivo.

Estrutura de Desenvolvimento de Aplicações

Dados na Informática

Os **dados** são as informações que serão processadas pelo aplicativo. Podem ser números, texto, datas, entre outros. Exemplo: o nome do usuário, sua senha ou data de nascimento. Manipular dados corretamente é essencial para o bom funcionamento de um sistema.

Estrutura

Toda aplicação possui uma estrutura que define como os dados serão processados e armazenados. Para isso, o sistema precisa seguir um fluxo:

- **Entrada de Dados:** As informações que o usuário insere no sistema, como um login ou senha.
- **Processamento de Dados:** O sistema verifica e processa essas informações, por exemplo, conferindo se a senha está correta.
- **Saída de Dados:** O resultado da operação, como permitir o acesso ao usuário ou exibir uma mensagem de erro.

Armazenamento

As informações precisam ser guardadas em algum lugar para serem recuperadas posteriormente. Em programação, esses locais de armazenamento são chamados de **variáveis** e **constantes**.

Variáveis e Constantes

Variáveis

As **variáveis** são locais de armazenamento temporário, onde os dados podem ser alterados durante a execução do programa. Cada variável tem um nome único, conhecido como **identificador**, que facilita o acesso a esses dados.

Exemplo:

```
javascript
```

```
let dataNascimento = "01/01/2000";
```

Aqui, a variável `dataNascimento` armazena a data de nascimento do usuário.

Constantes

As **constantes** são valores fixos que não podem ser alterados depois de definidos. Elas são úteis para armazenar dados que não devem ser modificados durante a execução do programa.

Exemplo:

```
javascript
```

```
const PI = 3.14159;
```

A constante `PI` armazena o valor de Pi, que não mudará durante a execução do programa.

Identificadores de Variáveis

Ao nomear uma variável, algumas regras devem ser seguidas:

1. O nome deve ser claro e descrever o que a variável armazena.
2. Deve começar com uma letra ou underscore (`_`), nunca com um número.
3. Não pode conter espaços ou caracteres especiais.

Exemplo de nome correto:

```
javascript
```

```
let alturaPessoa;
```

Tipos de Variáveis

As variáveis podem armazenar diferentes tipos de dados:

- **Inteiros:** Números sem casas decimais.
- **Reais:** Números com casas decimais.
- **Lógicos:** Valores booleanos (verdadeiro ou falso).
- **Caracteres:** Letras ou símbolos.

Exemplo:

```
javascript
```

```
let idade = 30; // inteiro  
let peso = 70.5; // real  
let nome = "Ana"; // caractere  
let ativo = true; // lógico
```

Estruturas de Controle

Ao desenvolver um programa, frequentemente é necessário tomar decisões. Isso é feito com **estruturas de controle**, como a condicional **SE**.

Exemplo de estrutura condicional:

javascript

```
if (senha == "1234") {  
    console.log("Acesso permitido");  
} else {  
    console.log("Senha incorreta");  
}
```

Neste exemplo, o programa verifica se a senha está correta e, dependendo da resposta, exibe uma mensagem.

Exemplos de Estruturas de Controle

Exemplo 1:

javascript

```
let idade = 18;  
if (idade >= 18) {  
    console.log("Maior de idade");  
} else {  
    console.log("Menor de idade");  
}
```

Exemplo 2:

javascript

```
let saldo = 500;  
if (saldo >= 100) {  
    console.log("Compra permitida");  
} else {  
    console.log("Saldo insuficiente");  
}
```

Tipos de Dados em Variáveis

Definindo tipos de dados:

1. **PESO**: real (números com casas decimais).
2. **IDADE**: inteiro (números inteiros).
3. **ALTURA**: real (números com casas decimais).
4. **NOME**: caractere (texto).
5. **V ou F**: lógico (verdadeiro ou falso).
6. **NACIONALIDADE**: caractere (texto).

Regras para Definir Variáveis

- **Exemplo de nome correto:**
 - peso
 - alturaPessoa
- **Exemplo de nome incorreto:**
 - Idade* (caractere inválido)
 - 1NOME (não pode começar com número)
 - Conta corrente (não pode conter espaço)

Exemplo de como renomear variáveis corretamente:

1. idade: Correto.
2. conta_corrente: Correto (espaço removido e substituído por underscore).
3. peso: Correto.
4. animal: Correto.

Resumo Final

Este material cobre os principais conceitos sobre lógica de programação e variáveis, essenciais para entender como construir e analisar sistemas. A prática desses fundamentos é a chave para dominar a programação e criar soluções eficientes em qualquer linguagem.