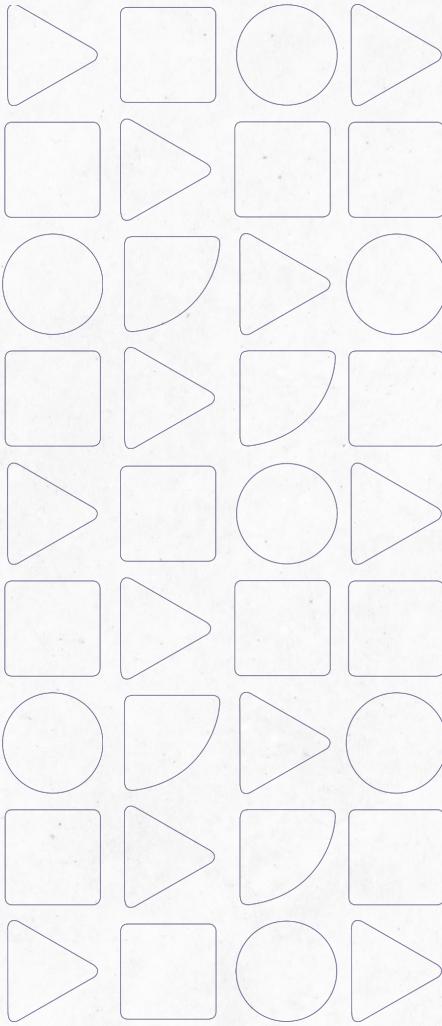


Introdução à Lógica de Programação

Disciplina: Programação Mobile



Conteúdos:

Introdução à lógica de programação.

Habilidade(s):

- Construir sistemas e aplicações *mobile*;
- Utilizar ambientes de desenvolvimento *mobile*;
- Criar, analisar e avaliar estruturas de aplicações.

Bloco 1

Introduzindo a linguagem de programação.

Vamos refletir

Em uma pequena cidade, há um único barbeiro. O barbeiro segue a seguinte regra: ele faz a barba de todos os homens da cidade que não fazem a própria barba, mas não faz a barba de homens que fazem a própria barba.

Tente responder:

O barbeiro faz a própria barba?



**Antes de desenvolver as ferramentas para
desenvolver aplicativos, é necessário ter o
conhecimento básico sobre lógica de programação!**

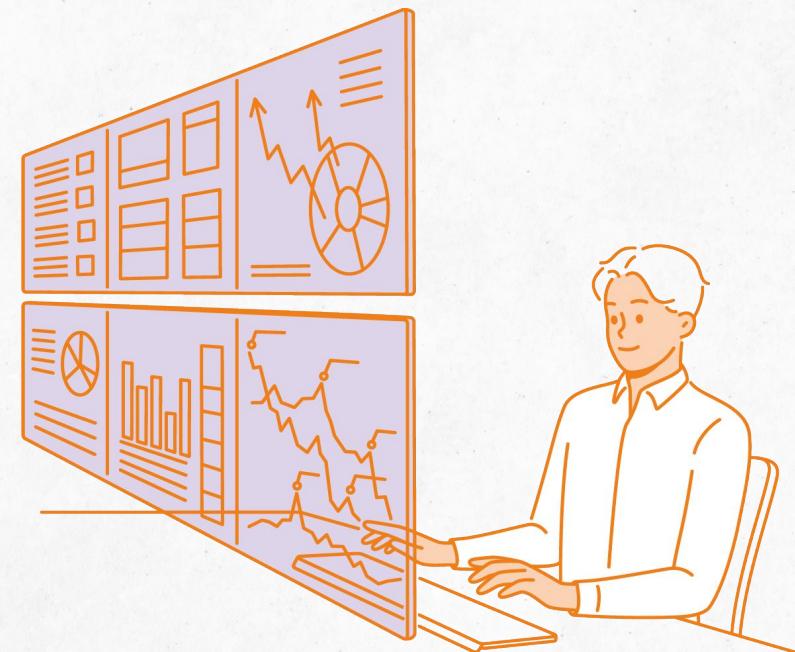


O que é lógica de programação?

É uma forma de organizar instruções de maneira sequencial e lógica para resolver um problema.



É como uma **receita de bolo**: para confeitar, é necessário seguir alguns passos, como misturar a farinha com o açúcar, adicionar ovos, pré-aquecer o forno etc.



Entretanto, é necessário diferenciar...

Você sabe declarar a diferença entre esses três tópicos?

Lógica

Lógica de programação

Sequência lógica

Gabarito

E aí, acertou?

Lógica

É a ciência que lida com **princípios do raciocínio** e da inferência, focando na validade dos argumentos.

Lógica de programação

É o processo de desenvolver algoritmos ou instruções lógicas que um computador pode **entender e executar** corretamente.

Sequência lógica

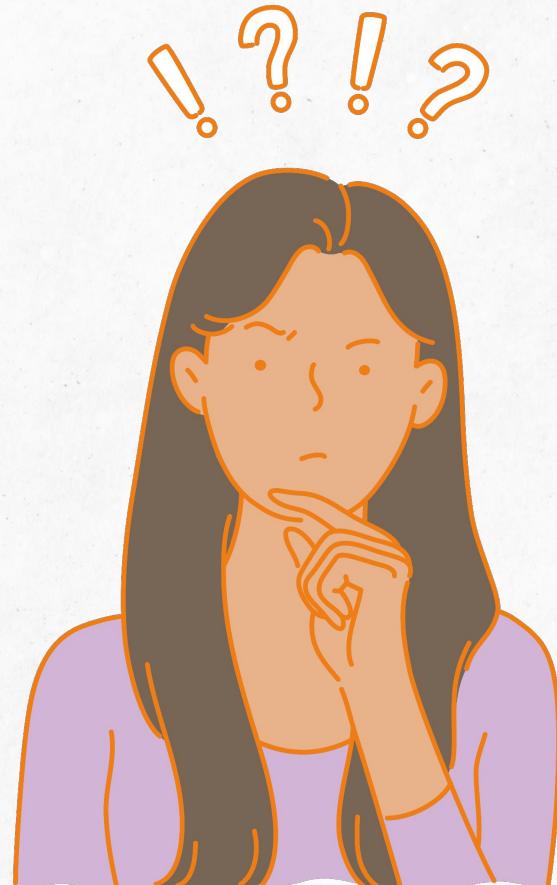
Refere-se à **ordem** em que as instruções ou os comandos são organizados em um programa.

Bloco 2

Explorando os dados.

Vamos refletir um pouco

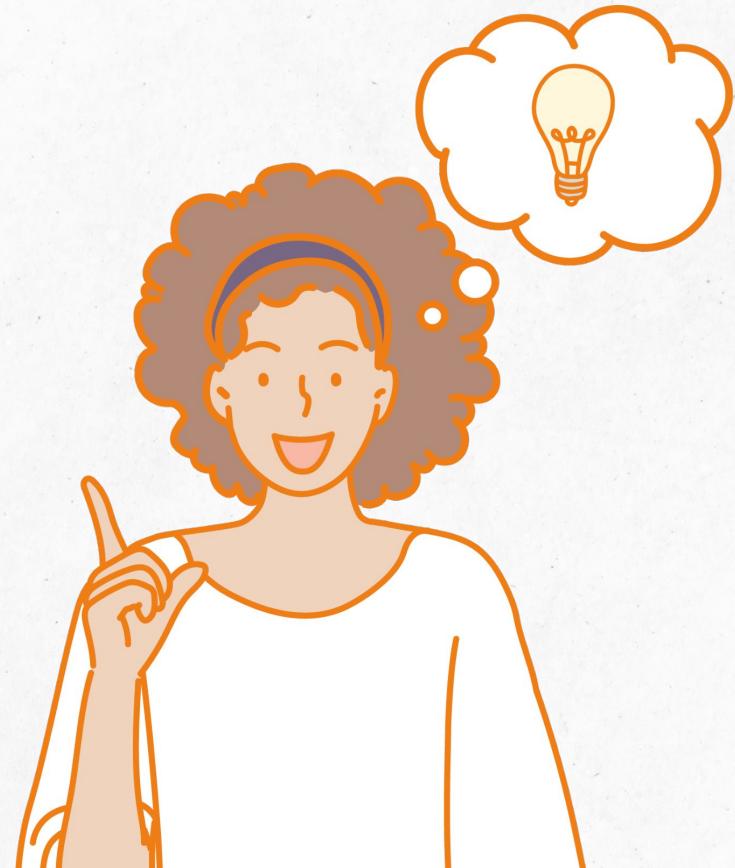
Imagine que você está desenvolvendo um aplicativo pela primeira vez. Quais são os passos iniciais mais indicados?



Vamos refletir um pouco

Imagine que você está desenvolvendo um aplicativo pela primeira vez. Quais são os passos iniciais mais indicados?

Isso mesmo, os dados! Vamos conhecê-los?



Hora de entender os dados

Os dados são informações que um programa precisa para funcionar corretamente.
Isso pode incluir **números, texto, datas, imagens**, entre outros.

Estruturas

São as instruções necessárias que o aplicativo deverá realizar. Isso inclui dados de entrada, de processamento e de saída.

Armazenamento

Corresponde à forma como os dados (informações) serão guardados.

Entrada: as informações que o usuário declara no aplicativo.

Processamento: o que o aplicativo deverá executar.

Saída: o que o aplicativo deverá apresentar ao usuário.

Vamos observar como isso funciona por meio de um exemplo

Imagine que você está desenvolvendo um aplicativo de uma biblioteca virtual, na qual os usuários podem pesquisar, reservar e ler livros.



Dados

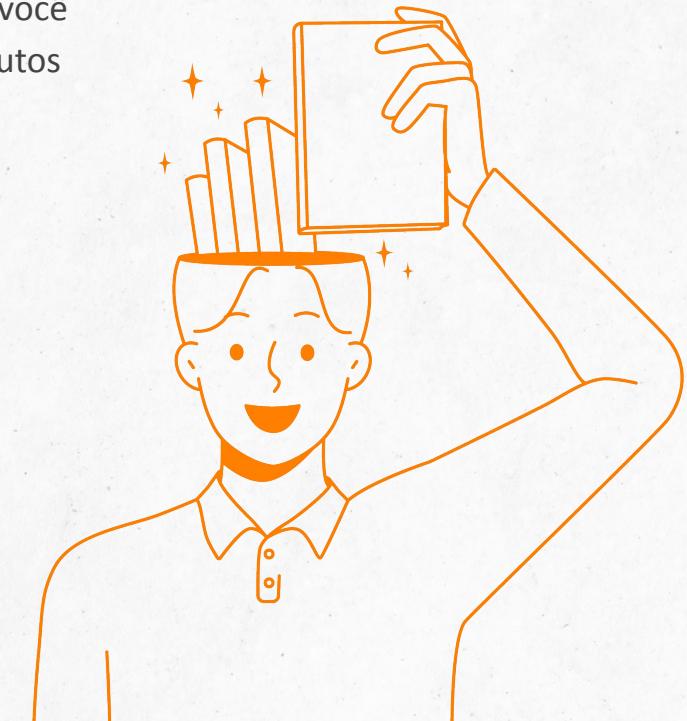
Você pode incluir informações como o **título do livro, o autor, a sinopse, a data de publicação, o número de páginas etc.**

Essas são as informações que são armazenadas e manipuladas pelo aplicativo.



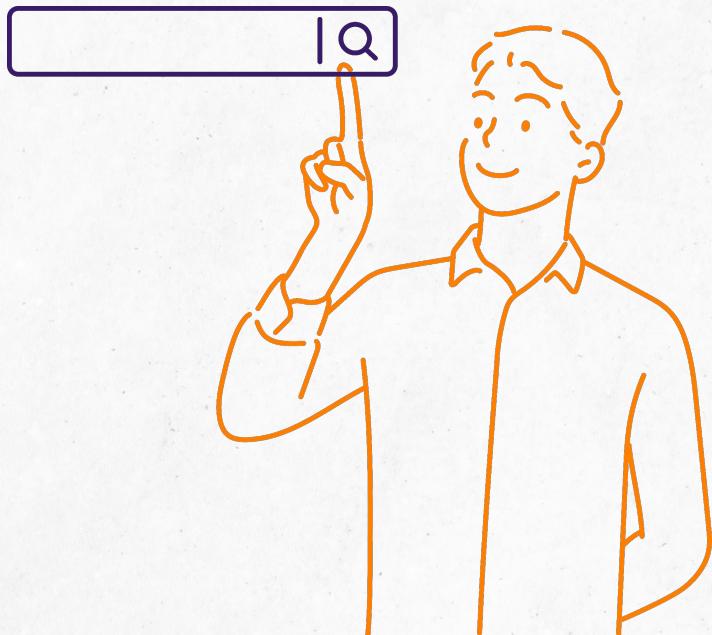
Estrutura de dados

Determina como essas informações são organizadas. Por exemplo, você pode ter uma estrutura de dados chamada "livro" que possui atributos como "título", "autor", "sinopse" e "data de publicação".



Entrada de dados

O usuário pode **digitar o título de um livro na barra de pesquisa para realizar uma busca**. Nesse caso, o título digitado é a entrada de dados fornecida pelo usuário.



Processamento de dados

O aplicativo, nesse caso, **receberia o título digitado pelo usuário**, buscara na base de dados o livro correspondente e **retornaria os resultados da pesquisa**.



Saída de dados

Pode ser uma **lista de livros** encontrados com base na pesquisa do usuário.
Geralmente, esses resultados são exibidos na interface do aplicativo para
que o usuário possa escolher uma obra.



Armazenamento de dados

No caso da biblioteca virtual, os dados dos livros podem ser armazenados em um banco de dados, no qual cada livro é representado por uma entrada na tabela de livros. Cada atributo do livro (título, autor, sinopse etc.) ocupará uma posição na memória do dispositivo e será identificado no código do aplicativo.



Bloco 3

Vamos praticar?

Vamos analisar?

Analisem as situações abaixo referentes a um aplicativo móvel de entrega de comida.
Identifiquem **dados, estrutura de dados, entrada de dados, processamento de dados, saída de dados e armazenamento de dados**. Preparados?



Vamos analisar?

Ao abrir o aplicativo, você escolhe um lanche.



O aplicativo apresenta uma lista de restaurantes disponíveis de acordo com a localização informada.



Após um novo acesso, o aplicativo salva o seu histórico de pedidos.



Depois de adicionar a sua localização, o aplicativo acessa o seu banco de dados.



No aplicativo, há categorias organizadas por tipo de restaurante, como “chinês” ou “comida caseira”.



Navegando no aplicativo, você tem acesso às características dos restaurantes, como cardápio, endereço e tempo de entrega.



E aí, acertaram?

Ao abrir o aplicativo, você escolhe um lanche.

O aplicativo apresenta uma lista de restaurantes disponíveis de acordo com a localização informada.

Após um novo acesso, o aplicativo salva o seu histórico de pedidos.

Depois de adicionar a sua localização, o aplicativo acessa o seu banco de dados.

No aplicativo, há categorias organizadas por tipo de restaurante, como “chinês” ou “comida caseira”.

Navegando no aplicativo, você tem acesso às características dos restaurantes, como cardápio, endereço e tempo de entrega.



Entrada de dados.



Saída de dados.



Armazenamento de dados.



Processamento de dados.



Estrutura de dados.



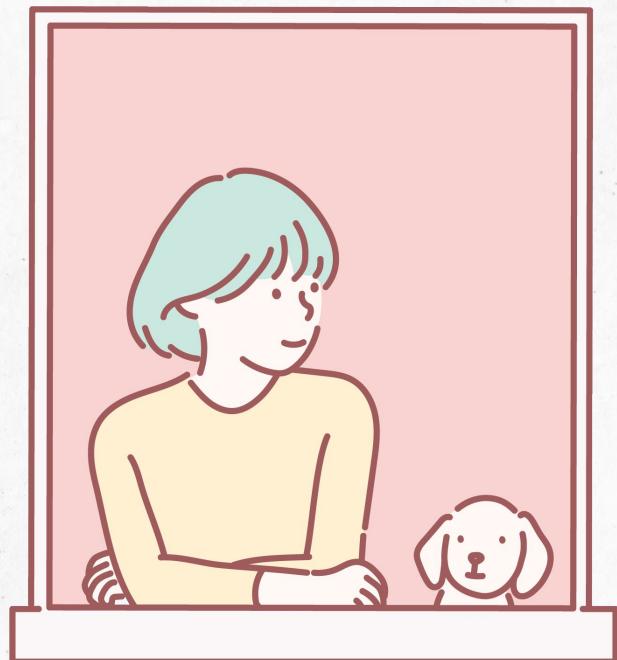
Dados.

Vamos exercitar os nossos conhecimentos?

Dividam-se em grupos e analisem o caso a seguir:

O PetMood é um aplicativo revolucionário que permite que você identifique o **humor do seu animal de estimação**. Usando a câmera do seu *smartphone*, o aplicativo analisa as expressões faciais do seu animal e interpreta as suas emoções.

Peguem uma folha de papel e realizem a **estrutura do desenvolvimento de dados** desse aplicativo. Nela, deverá conter dados, estrutura de dados, entrada de dados, processamento de dados, saída de dados e armazenamento de dados.



Vamos nos reunir?

Formem um círculo para discutir.



Bloco 4

Hora de conhecer os dados de tipo variável e constante.

Agora que você conheceu um pouco mais sobre dados, chegou a hora de explorar a estrutura. Para isso, é necessário compreender a identificação!

Confira a seguir.



Dados variáveis

É um espaço na memória que pode ser alterado a qualquer momento.

Em uma caixa chamada “**idade**”, você pode colocar diferentes valores ao longo do tempo. Nesse caso, a caixa está apta para colocar o valor 10, depois substituí-lo por 15, e assim por diante.



Dados constantes

São semelhantes a uma variável, mas o seu valor não pode ser alterado depois de definido.

Se uma caixa tem uma constante chamada "PI", ela sempre terá o valor 3.14159. Portanto, essa constante não pode ser modificada, pois permanecerá com o mesmo valor.

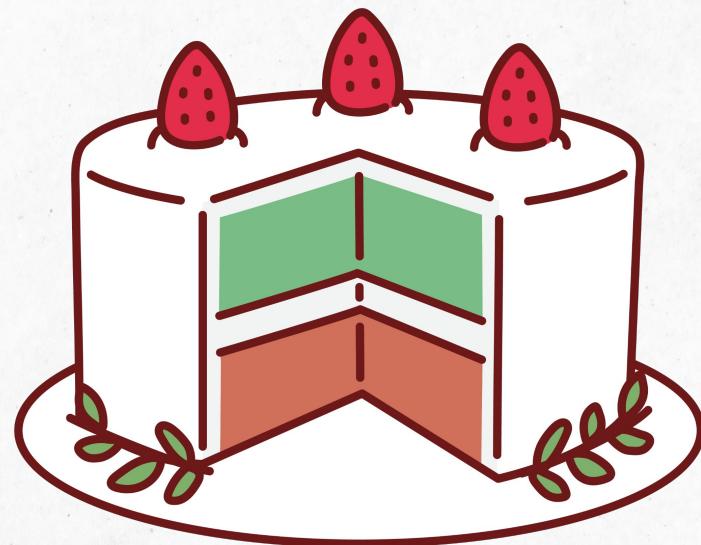


Explorando exemplos

Vamos analisar como esses dados seriam organizados em uma **receita de bolo**:

Os dados **variáveis** podem se referir à **quantidade de chocolate** que você pode adicionar a depender da sua preferência.

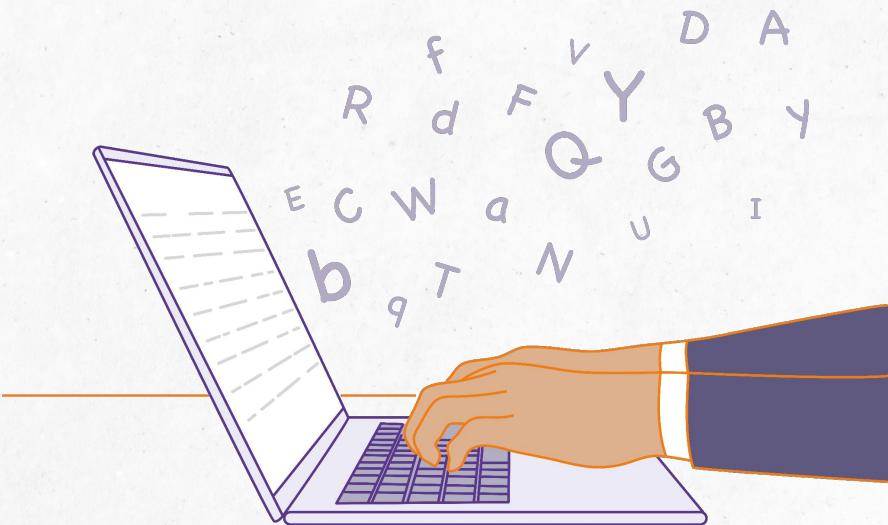
Entretanto, para criar um bolo, é necessário utilizar **duas xícaras de trigo**. Portanto, esse dado é **constante**.



Criando dados variáveis

Os dados variáveis precisam de identificadores e podem assumir apenas um único valor por vez. Para isso, é necessário respeitar algumas **regras e convenções**.

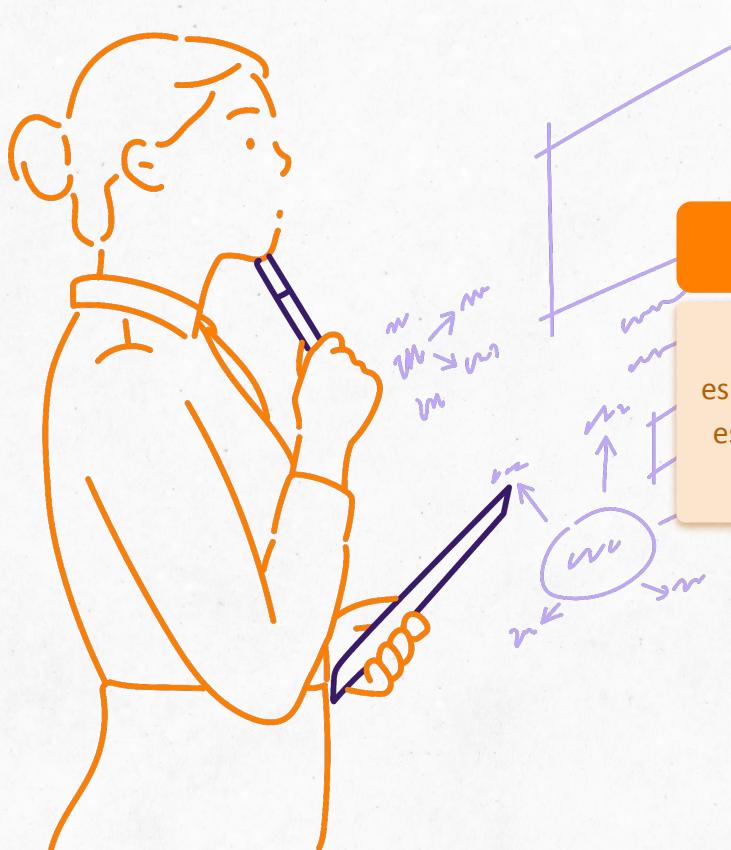
Esse processo se assemelha à criação de uma pasta no computador: para tal, o usuário não pode utilizar alguns símbolos ou determinados caracteres.



Regras para utilizar um identificador

Confira o que é necessário:

Regra	Exemplo
É importante usar nomes (identificadores) que defina claramente o que será armazenado.	idadePessoa (25)
Pode iniciar por letras ou <i>underline</i> (_).	_enderecoresidencial
Não pode iniciar com um número.	123tiposanguineo
Não pode ter sinais de pontuação.	...numerocpf
Não deve conter espaço em branco.	data nascimento
Não criar variáveis com nomes muito longos.	historicodeensinomedio



Regra CamelCase

Ela tem esse nome devido ao fato de que as palavras são unidas sem espaços e a primeira letra de cada palavra subsequente (exceto a primeira) é escrita em maiúscula, o que cria um padrão visual semelhante às costas de um **camelo**.



Como escrever com a CamelCase?

1

Escreva a primeira palavra em minúsculas. A primeira palavra do nome começa com letra minúscula, sem espaços. Por exemplo, "nome";

2

Inicie cada palavra subsequente com letra maiúscula. Para as palavras seguintes, a primeira letra é maiúscula. Por exemplo, "DeUsuario";

3

Sem espaços ou caracteres especiais. Não são permitidos espaços, acentos ou caracteres especiais (como vírgulas ou pontos) dentro do nome;

4

Mantenha a consistência. É importante manter a mesma convenção ao longo do código. Se você começar um nome com CamelCase, mantenha essa convenção.

Bloco 5

Conhecendo os tipos de dado variável.

Identifique corretamente

Classifiquem, de forma oral, as nomenclaturas dos dados a seguir, indicando se eles estão **corretos ou incorretos**.

?

numero1;

?

valor_3;

?

num2;

?

total#123;

?

1numero;

?

contador10;

?

@valor;

?

variável-abc.

Gabarito

E aí, acertaram?



numero1;



num2;



1numero (começa com um dígito);



@valor (contém um caractere especial);



valor_3;



total#123 (contém um símbolo não permitido);



contador10;



variável-abc (contém um hífen).

Explorando os dados variáveis

Ao criar um dado que seja variável, é necessário informar **qual tipo de dado será armazenado**.
A seguir, confira os aspectos básicos nos quais eles podem ser divididos.



Inteiros

Armazena números inteiros.

idade = 25

numeroDeAlunos = 30

temperatura = -10



Real

Armazena números com classe decimal.

peso = 65.5

altura = 1.75

preco = 9.99



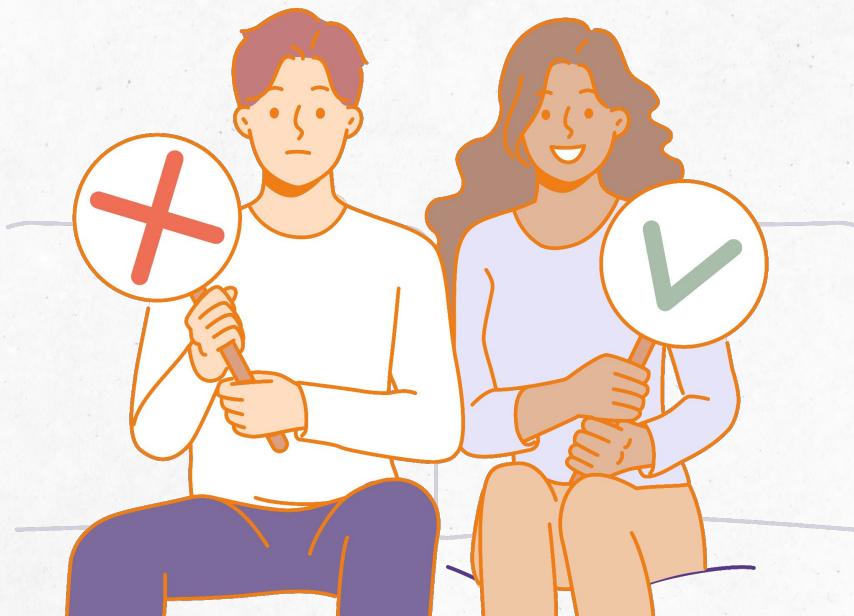
Lógico

Armazena informações como verdadeiro ou falso.

estaChovendo = verdadeiro

luzAcesa = falso

alunoAprovado = verdadeiro



Caracter

Armazena letras ou caracteres especiais.

```
letra = 'A'
```

```
caracterEspecial = '$'
```

```
simbolo = '@'
```



Vamos exercitar?

Classifiquem os dados abaixo de acordo com os **tipos de dados variáveis**.

Dado	Tipo
Peso	
Idade	
Altura	
Nome	
V ou F	
Nacionalidade	

Gabarito

E aí, acertaram?

Dado	Tipo
Peso	Real
Idade	Inteiro
Altura	Real
Nome	Caracter
V ou F	Lógico
Nacionalidade	Caracter

Bloco 6

Conhecendo as estruturas condicionais.

Estrutura condicional

Durante o desenvolvimento de um programa, é necessário estar atento à **estrutura condicional**. Isso se refere à **tomada de decisão** que um sistema precisa determinar.

Elá é semelhante a uma bifurcação em uma estrada, na qual você pode escolher um caminho dependendo de certas condições.



Estrutura *if-else*

É a estrutura condicional mais comum.

Nesse caso, é como se você fizesse uma pergunta para o programa e ele respondesse de acordo com a resposta que você recebe.

Imagine que você está criando um programa que verifica se uma pessoa é maior de idade.

Se o dado de entrada for maior que 18 = maior de idade.

Se o dado de entrada for menor que 18 = menor de idade.



Explorando exemplos

Vamos analisar as decisões que um computador executa?

Veja as frases a seguir:

Levantar-se da cadeira.

Caminhar até a porta.

Abrir a porta.

Em situações adicionais, você poderia ver a sua mãe parada no corredor ou notar que o dia está chuvoso. **Como isso afetaria a história?**

Os três quês

Vamos formar uma roda para que possamos debater sobre os três quês?

O que me anima?	O que me preocupa?	O que eu preciso explorar?
O que animou você sobre o assunto abordado na aula?	Eita! Qual tema da aula te preocupa?	O que você sente que precisa explorar acerca do assunto?



Referências Bibliográficas

PROZ EDUCAÇÃO. *Apostila de Programação Mobile*. 2023.