



**SENAI** *Serviço Nacional  
de Aprendizagem  
Industrial*



# LÓGICA DE PROGRAMAÇÃO COM DART

Prof. Mauro Andrade  
Curso Técnico de Desenvolvimento de  
Sistemas

## POR QUE USAR DART?

- Linguagem moderna e fácil
- Desenvolvida pelo Google
- Usada no Flutter (apps mobile)
- Sintaxe limpa e intuitiva
- Ótima para iniciantes



# HELLO WORLD!

```
void main() {  
    print("Olá, Mundo!");  
}
```

- void main() → Função principal sem retorno
- print() → Mostra na tela
- ; → Ponto e vírgula (obrigatório)



# VARIÁVEIS

Tipos de Dados:

- String → Texto
- int → Números inteiros
- double → Números decimais
- bool → Verdadeiro ou Falso

Exemplo:

```
String nome = "Maria";
```

```
int idade = 25;
```

```
double altura = 1.65;
```

```
bool estaChovendo = true;
```



# OPERADORES MATEMÁTICOS

```
int a = 10;
```

```
int b = 3;
```

```
print(a + b); // 13 → Soma
```

```
print(a - b); // 7 → Subtração
```

```
print(a * b); // 30 → Multiplicação
```

```
print(a / b); // 3.33 → Divisão
```

```
print(a % b); // 1 → Resto
```

Ordem: Parênteses → Multi/Div → Soma/Sub



## JUNTANDO TEXTO

```
String nome = "Ana";  
int idade = 20;
```

```
print("Meu nome é " + nome); // Concatenação  
print("Tenho $idade anos"); // Interpolação  
print("Nascida em ${2024 - idade}"); // Cálculo  
dentro do texto
```



# Dart

Prefira interpolação: mais limpo!

# INTERAGINDO COM O USUÁRIO

```
import 'dart:io';

void main() {
  print("Qual seu nome?");
  String nome = stdin.readLineSync()!;

  print("Olá, $nome! ");
}
```



- import → Traz funcionalidades extras
- stdin.readLineSync()! → Lê entrada do usuário
- ! → Confirma que não é nulo



## CONDIÇÕES = IF/ELSE

```
void main() {  
  int idade = 18;  
  if (idade >= 18) {  
    print("Pode dirigir!");  
  } else {  
    print("Ainda não pode dirigir! ");  
  }  
}
```

- > maior que
- < menor que
- >= maior ou igual
- <= menor ou igual
- == igual a



## Exercício - Condições

```
void main() {  
    double nota = 7.5;  
    if (nota >= 7) {  
        print("Aprovado! ");  
    } else if (nota >= 5) {  
        print("Recuperação! ");  
    } else {  
        print("Reprovado! ");  
    }  
}
```

Desafio: Adicione “Eh ninja!”  
para notas acima de 9



## Loops - Repetindo Ações

```
// For loop - Número definido de vezes  
for (int i = 1; i <= 5; i++) {  
    print("Número $i");  
}
```

```
// While - Enquanto condição for verdadeira  
int contador = 1;  
while (contador <= 5) {  
    print("Contando: $contador");  
    contador++;  
}
```



## Exercício - Loops

### CONTAGEM REGRESSIVA

```
for (int i = 10; i >= 1; i--) {  
    print("$i...");  
}  
print("Foguete lançado! ");
```

2 desafios:

- Conte de 2 em 2 até 20
- Liste os números pares até 10



# Listas - Coleções de Dados

LISTAS = GRUPO DE VALORES

```
list<String> frutas = ["maçã", "banana", "laranja"];
```

```
list<int> numeros = [1, 2, 3, 4, 5];
```

// Acessando elementos

```
print(frutas[0]); // "maçã" - índice começa em 0!
```

// Adicionando

```
frutas.add("uva");
```

// Tamanho

```
print(frutas.length); // 4
```



# Dart

## Exercício - Listas

```
List<String> compras = [];
```

```
// Adicione 3 itens
```

```
compras.add("pão");
```

```
compras.add("leite");
```

```
compras.add("ovos");
```

```
// Mostre a lista
```

```
for (int i = 0; i < compras.length; i++) {
```

```
    print("${i + 1}. ${compras[i]}");
```

```
}
```



# Dart

# Funções

FUNÇÕES = BLOCO DE CÓDIGO REUTILIZÁVEL

```
// Definindo função
void saudacao(String nome) {
    print("Olá, $nome! 😊");
}

// Usando função
void main() {
    saudacao("Maria");
    saudacao("João");
}
```

**Vantagens:**  
Reutilização +  
Organização



# Exercício - Funções

## CALCULADORA SIMPLES

```
double somar(double a, double b) {  
    return a + b;  
}  
  
void main() {  
    double resultado = somar(5, 3);  
    print("Resultado: $resultado"); // 8.0  
}}
```

**Desafio:** Crie  
funções para:

- Subtrair
- Multiplicar
- Dividir





# Projeto Final - Jogo de Adivinhação

Você deve criar um programa em Dart que implemente um jogo de adivinhação onde o jogador tenta descobrir um número secreto gerado aleatoriamente pelo computador entre 0 e 10

- O programa deve pedir ao jogador para digitar um palpite
- Deve informar se o palpite é maior, menor ou igual ao número secreto
- Deve contar o número de tentativas realizadas
- O jogo para quando jogador acertar o número
- Quando acertar, deve mostrar uma mensagem de parabéns com o número de tentativas

## RECURSOS SUGERIDOS:

```
// Biblioteca para gerar números aleatórios
import 'dart:math';

// Biblioteca para entrada de dados
import 'dart:io';

void main() {
  // DICA: Use Random() para criar o gerador de números aleatórios
  // DICA: nextInt(10) gera números de 0 a 9, some 1 para ter 1 a 10
}
```



# Dart

# Projeto Final - Jogo de Adivinhação

```
import 'dart:io';
import 'dart:math';

void main() {
  final random = Random();
  int numeroSecreto = random.nextInt(10) + 1;
  int tentativas = 0;

  print("Adivinhe o número de 1 a 10!");

  while (true) {
    tentativas++;
    print("Tentativa $tentativas: ");
    int palpito = int.parse(stdin.readLineSync()!);

    if (palpito == numeroSecreto) {
      print("🎉 Acertou em $tentativas tentativas!");
      break;
    } else if (palpito < numeroSecreto) {
      print("📈 Tente um número maior!");
    } else {
      print("📉 Tente um número menor!");
    }
  }
}
```

