



INTRODUÇÃO A LINGUAGEM DE PROGRAMAÇÃO DART

Prof. Me. Hélio Esperidião

A linguagem

- Dart é uma linguagem de programação apresentada pelo Google, em 2011, com o objetivo de substituir o JavaScript como principal linguagem embutida nos navegadores.

Algumas características da linguagem são:



A sintaxe é C-like, portanto, se você programa em Java, C#, PHP ou Javascript, você não terá dificuldades em aprender a linguagem;



Segue o paradigma orientado a objetos;



Todos os objetos herdam da classe Object;



Fortemente tipada, mas não é necessário colocar um tipo, pois Dart consegue inferir os tipos;

Variáveis

Tipo	Descrição	Exemplo
int	Valores inteiros	1, 2, -1233123
double	Números de ponto flutuante	0.5, -333.234
String	Texto de comprimento arbitrário. Aspas simples ou duplas podem ser usadas para criar uma string.	Texto' ou "Text"
bool	Um booleano é um valor lógico que pode ser verdadeiro ou falso.	true ou false
List	Uma coleção de objetos.	[1, 2, 3]
Map	Um mapa é um objeto que associa chaves e valores.	{ 'key-1': 'value-1', 'key-2': 'value-2' }

Operadores relacionais

Operação	Descrição	Exemplo
==	Igual	k == 0
!=	Diferente	x != 3
>	Maior que	w > 7
<	Menor que	a < 22
>=	Maior ou Igual a	b >= 80
<=	Menor ou Igual a	c <= -3

Condicionais

```
double x = double.parse(txtN1Controler.text);
double y = double.parse(txtN2Controler.text);
double r = calcularMedia(x,y);
resultado = r.toString();
if (r < 6.0){
    showDialog(
        context: context,
        builder: (context) {
            return AlertDialog(
                content: Text("Reprovado"),
            );
        },
    );
} else {
    showDialog(
        context: context,
        builder: (context) {
            return AlertDialog(
                content: Text("Aprovado"),
            );
        },
    );
}
```

Repetições

```
for (var i = 0; i < 5; i++) {  
    showDialog( context: context, builder: (context) {  
        return AlertDialog( content: Text("$i"), );  
    },);  
}
```

Repetições

```
int i = 0;
while (i < 5) {
    showDialog( context: context, builder: (context) {
        return AlertDialog( content: Text("$i"), );
    },);

    i++;
}
```


Programação
orientada a
objetos

Objetos

Classes

Atributos

Objetos

- Um objeto é uma implementação de Tipos Abstratos de Dados (TAD's) que encapsula dados e operações. Todo objeto tem um estado, um comportamento e uma identidade.

Classes

- Classe é uma abstração que define um tipo de objeto e o que objetos desse determinado tipo tem dentro dela (atributos) e também que tipo de ação ele é capaz de executar (métodos).

Atributos

- São características que os objetos possuem. Ao definir um atributo sempre é passado nome e o que irá guardar, ou seja, qual tipo de dado que ele vai armazenar, exemplo: cor do olho, altura.

Construtores

- Método que inicializa o estado de um objeto.

Usuario.dart

```
class Usuario {  
  String nome;  
  double peso, altura;  
  
  Usuario(this.peso, this.altura, this.nome);  
}  
  
// Programa principal  
void main() {  
  
}
```

Possibilidade de nomear construtores

```
class Usuario {
    late String nome;
    late String email;
    late String Senha;

    Usuario(String nome, String email, String Senha) {
        this.nome = nome;
        this.email = email;
        this.Senha = Senha;
    }

    Usuario.semSenha(String nome, String email) {
        this.nome = nome;
        this.email = email;
    }

    Usuario() {
        nome = "";
        email = "";
        Senha = "";
    }
}
```

Encapsulamento

No Java e C# você usa a palavra reservada `private`

No Dart basta colocar um *underline* (`_`) no início do nome de um atributo, método ou classe para torná-lo privado.

Encapsulamento

```
class Usuario {  
    late String _nome;  
    late String _email;  
    late String _Senha;  
  
    String get Senha => _Senha;  
  
    set Senha(String value) {  
        _Senha = value;  
    }  
  
    String get email => _email;  
  
    set email(String value) {  
        _email = value;  
    }  
  
    String get nome => _nome;  
  
    set nome(String value) {  
        _nome = value;  
    }  
}
```

Métodos

```
import 'dart:math';

class Point {
  double x = 0;
  double y = 0;

  Point(this.x, this.y);

  double distanceTo(Point other) {
    var dx = x - other.x;
    var dy = y - other.y;
    return sqrt(dx * dx + dy * dy);
  }
}
```

Métodos, funções e procedimentos

```
double calcularMedia(double a, double b){  
    double resultado = (a + b)/2;  
    return resultado;  
}
```

Importe e instancie

```
import 'package:flutter/material.dart';  
import 'Usuario.dart';
```

```
Usuario usuario= new Usuario();  
Usuario u2 = new Usuario.semSenha("helio","helio@teste.com");
```