

## Atividade Aula 01

07-02-2025

### Dart

1. Imprima "Hello, Dart!"
2. Variáveis e Tipos: Crie variáveis do tipo int, double, String, bool e imprima seus valores.
3. Operações Matemáticas: Peça dois números ao usuário e exiba a soma, subtração, multiplicação e divisão.
4. Conversão de Tipos: Converta um número double para int e vice-versa.
5. Estruturas Condicionais: Verifique se um número é par ou ímpar.
6. Switch Case: Crie um programa que leia um dia da semana e exiba uma mensagem correspondente.
7. Loop for: Exiba os números de 1 a 10.
8. Loop while: Some os números de 1 a 100 e exiba o resultado.
9. Lista Simples: Crie uma lista de frutas e imprima cada uma delas usando um loop.
10. Mapa (Dicionário): Crie um mapa com nomes e idades e exiba os valores.
11. Função Simples: Crie uma função que receba dois números e retorne a soma deles.
12. Função com Valor Padrão: Faça uma função que receba um nome e exiba uma saudação. Se nenhum nome for passado, use "Visitante".
13. Função Recursiva: Calcule o fatorial de um número.
14. Manipulação de Listas: Ordene uma lista de números.
15. Filtro em Listas: Filtre os números pares de uma lista usando .where().
16. Exceções: Crie um programa que tente dividir um número por zero e trate o erro.
17. Classe e Objeto: Crie uma classe Carro com propriedades marca, modelo e ano, e um método para exibir as informações.
18. Encapsulamento: Modifique a classe Carro para que modelo seja privado e crie métodos getter e setter.
19. Herança: Crie uma classe Animal com um método emitirSom(), e classes Cachorro e Gato que herdam Animal e implementam emitirSom().
20. Mixin: Crie um mixin chamado Volante e aplique-o a uma classe Carro.
21. Futuro e Assincronismo: Simule uma requisição assíncrona usando Future.delayed().
22. Stream: Crie uma Stream que emita números de 1 a 5 com intervalo de 1 segundo entre eles.
23. Manipulação de Arquivos: Leia e escreva em um arquivo usando dart:io.

24. Uso de Pacotes Externos: Instale o pacote http e faça uma requisição GET para uma API pública.
25. Programação Funcional: Use map, reduce e where para transformar uma lista de números.
26. Generics: Crie uma classe genérica Caixa<T> que armazena um valor de qualquer tipo.
27. Extensão de Classes: Crie uma extensão para a classe String que adiciona um método toCamelCase().
28. Operadores Avançados: Use o operador ?? (null-aware) para tratar valores nulos em um mapa de configurações.