

Desenvolvimento  
Mobile 1  
Aula 03

Prof. Me Daniel Vieira



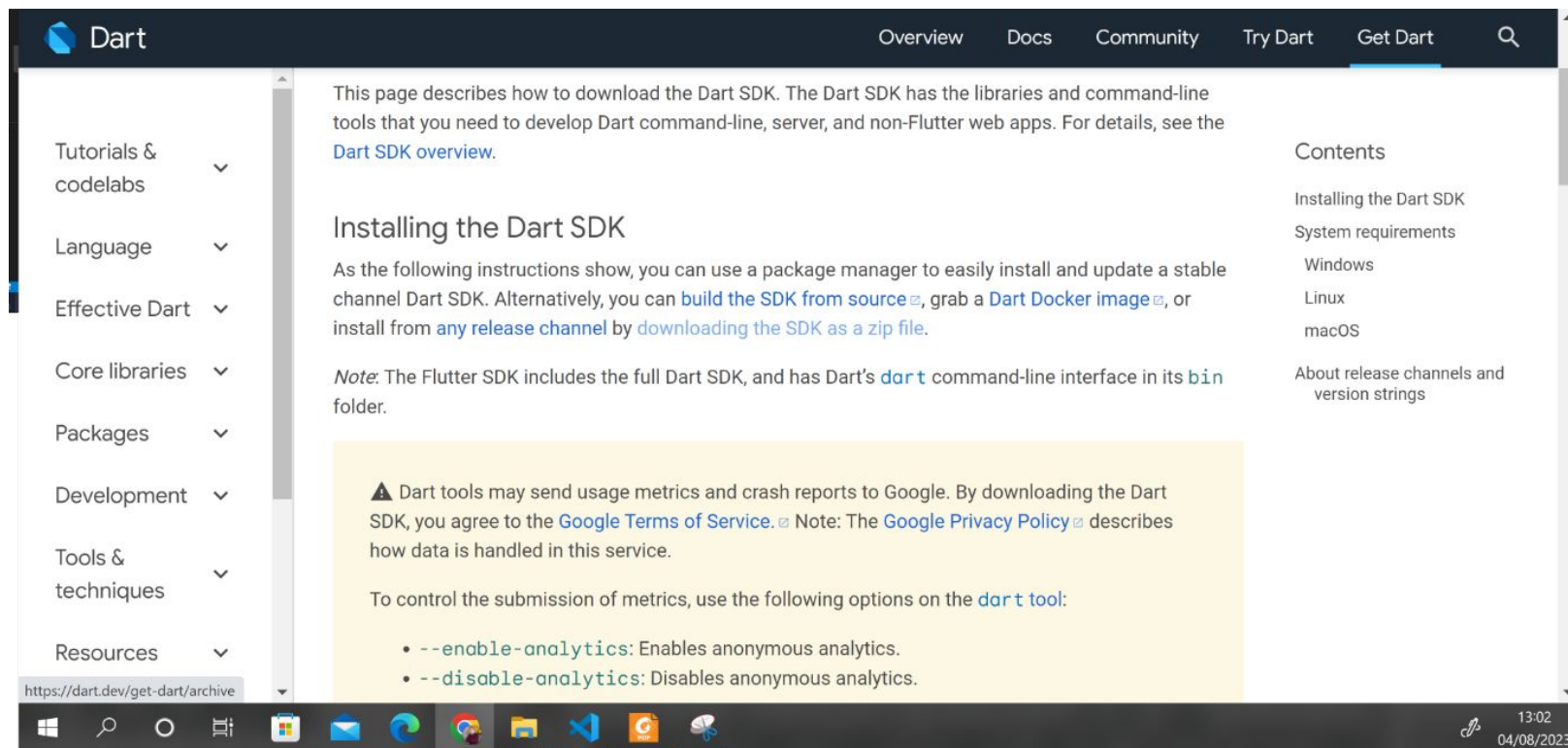
# Agenda

- 1- Configuração VSCODE
- 2-Variáveis Dart
- 3 - Estruturas condicionais
- 4 -Estrutura de repetição
- 5 - Arrays
- 6- Exercícios

# Configuração VSCODE - Linguagem Dart

É necessário baixar o SDK do Dart para utilizá-lo no VSCODE  
<https://dart.dev/>

# Configuração VSCODE - Linguagem Dart



The screenshot shows the Dart SDK documentation page. The left sidebar contains a navigation menu with the following items: Tutorials & codelabs, Language, Effective Dart, Core libraries, Packages, Development, Tools & techniques, and Resources. The main content area is titled 'Installing the Dart SDK' and includes a paragraph about using a package manager or building from source. A note mentions that the Flutter SDK includes the full Dart SDK. A yellow box contains a warning about Google analytics and provides command-line options to control analytics submission. The right sidebar shows a 'Contents' section with links to 'Installing the Dart SDK', 'System requirements', 'Windows', 'Linux', 'macOS', and 'About release channels and version strings'. The top navigation bar includes links for Overview, Docs, Community, Try Dart, and Get Dart. The bottom of the image shows a Windows taskbar with various application icons and a system clock indicating 13:02 on 04/08/2023.

**Dart** Overview Docs Community Try Dart Get Dart

This page describes how to download the Dart SDK. The Dart SDK has the libraries and command-line tools that you need to develop Dart command-line, server, and non-Flutter web apps. For details, see the [Dart SDK overview](#).

## Installing the Dart SDK

As the following instructions show, you can use a package manager to easily install and update a stable channel Dart SDK. Alternatively, you can [build the SDK from source](#), grab a [Dart Docker image](#), or install from [any release channel](#) by [downloading the SDK as a zip file](#).

*Note:* The Flutter SDK includes the full Dart SDK, and has Dart's `dart` command-line interface in its `bin` folder.

⚠ Dart tools may send usage metrics and crash reports to Google. By downloading the Dart SDK, you agree to the [Google Terms of Service](#). *Note:* The [Google Privacy Policy](#) describes how data is handled in this service.

To control the submission of metrics, use the following options on the `dart` tool:

- `--enable-analytics`: Enables anonymous analytics.
- `--disable-analytics`: Disables anonymous analytics.

**Contents**

- Installing the Dart SDK
- System requirements
- Windows
- Linux
- macOS
- About release channels and version strings

<https://dart.dev/get-dart/archive>

13:02  
04/08/2023

# Configuração VSCODE - Linguagem Dart

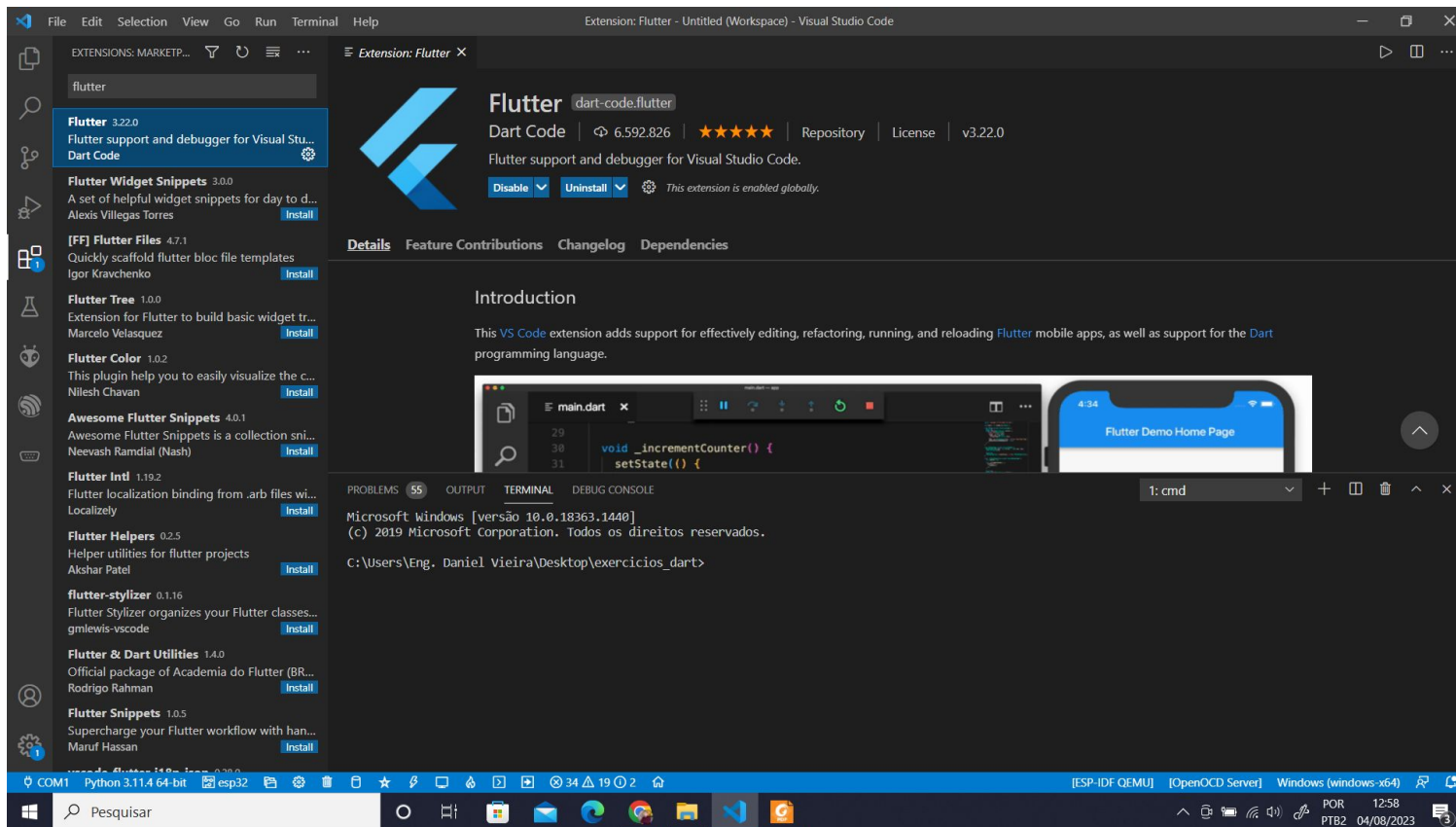
Descompactar o SDK , após o download e copiar a pasta para o disco c

Para utilizar a linguagem Dart é necessário configurar a variável de ambiente, assim como fizemos com

o Flutter

C:\dart-sdk\bin

# Configuração VS CODE



# Tipos de variáveis Dart

```
import "dart:io";  
void main()  
{  
  // Variável que armazena números inteiros  
  int idade = 28;  
  print("Idade: $idade");  
  // Variável que armazena números decimais  
  double raio = 10.25;  
  print("Raio: $raio");  
  // Variável que armazena caracteres e textos  
  String nome = "Daniel";  
  print("Ola $nome, seja bem vindo");  
  //Variavel que armazena verdadeiro ou false  
  bool ligado = true;
```

# Tipos de variáveis Dart

// Variável que guarda uma lista genérica

```
List numerosgenericos = [10,"Daniel", true,20];  
print(numerosgenericos);
```

// Variável que guarda uma lista de numeros inteiros

```
List<int>numerosinteiros = [10,20,30,40];  
print(numerosinteiros);
```

//Variável que guarda um dicionário com chave e valor em formato texto

```
Map<String, String> nome_sobrenome = {"Daniel": "Vieira", "Senai": "Roberto Mange"};
```

// Variável sem tipo pré definido, seu tipo é igual ao tipo do primeiro valor que recebe

```
var sobrenome = "Vieira";  
print(sobrenome);
```



# Exemplo código Dart

```
import 'dart:io';  
void main()  
{  
  print("Digite seu nome");  
  var nome = stdin.readLineSync();  
  print("Seu nome é $nome");  
}
```

```
import 'dart:io';  
  
void main() {  
  // Solicitando ao usuário que digite seu nome  
  print("Digite seu nome:");  
  String nome = stdin.readLineSync()!;  
  // Solicitando ao usuário que digite sua idade  
  print("Digite sua idade:");  
  String idadeString = stdin.readLineSync()!;  
  int idade = int.parse(idadeString);  
  
  // Exibindo as informações digitadas pelo  
  usuário  
  print("Seu nome é: $nome");  
  print("Sua idade é: $idade");  
}
```

# Comandos Dart

```
void main(){}
```

- void -> Vazio, não retorna nada;
- () → recebe argumentos de fora;
- {} -> Nosso código fica aqui dentro

```
print();
```

- função que imprime informações no console;
- () → recebe a informação;
- ; → necessário para terminar a frase.

# Comandos Dart

Para acessar informações de uma variável utilizamos o \$variável  
Necessário estar entre “ \$variavel”

Para receber uma informação digitada pelo usuário utilizamos a função  
`print("Digite seu nome");`  
`var nome = stdin.readLineSync();`

Para utilizar a função `stdin.readLineSync()` é necessário importar a biblioteca  
`dart:io`

Essa biblioteca permite utilizar o comando acima  
`import 'dart:io';`

# Comandos Dart

Existem casos que devemos converter a informação digitada pelo usuário como por exemplo para realizar cálculos.

Nesse caso utilizamos a função

```
int.parse(readLineSync()!);
```

```
float.parse(readLineSync()!);
```

# Estruturas condicionais

```
if(condicao == True)
{
}
else
{
}
```

```
if(idade >=18)
{
    print("Maior de idade");
}
else
{
    print("Menor de idade");
}
```

# Estruturas condicionais

```
switch(condição )
```

```
{
```

```
    case 1:
```

```
        break;
```

```
    case 2:
```

```
        break;
```

```
    case 3:
```

```
        break;
```

```
}
```

```
if(idade >=18)
```

```
{
```

```
    print("Maior de idade");
```

```
}
```

```
else
```

```
{
```

```
    print("Menor de idade");
```

```
}
```

# Estruturas de repetição - for

```
for(int i=0; i<10;i++){}
```

for → comando para iniciar o loop  
(int i=0; i<10;i++) → argumento de condição

int i=0 → iniciar objeto de comparação

i<10 → comparação para condição

i++ → ação pós-loop

```
for(int a=0;a>=6;a=a+2)
{
    print("Repetição $a");
}
```

# Estruturas de repetição - while

```
int i=0;  
while(i<10)  
{  
    print(i);  
    i=i+1;  
}
```



# Estruturas de repetição - do while

```
int i=0;  
do  
{  
    print(i);  
    i=i+1;  
}while(i<10);
```

# Comentários

Importantes para guiar quem vai ler seu código;

- `//` ou `/**/`
  - Comentários simples e cinzas, ideais para não poluir o código;
- `///` ou `****/`
  - Comentários de Documentação, verdes e podem poluir o código.

# Arrays

É um conjunto de variáveis onde podemos adicionar valores.

Ex:

```
void main()
{
    List a=["Daniel","Senai","Mobile"];
    List<int> num=[1,2,3,4,5,6];
    print(a);
    print(num);
}
```

# Exercícios

- 1) Crie um programa utilizando a linguagem Dart que receba informações digitadas pelo usuário:  
Nome, idade, curso
- 2) Crie um programa que receba a nota de dois alunos, calcule sua média e informe se o aluno está aprovado ou reprovado conforme a média.  
Se média for maior ou igual a 7 - Aprovado  
Maior ou igual a 4 e menor do que 7 - Exame  
Menor do que 4 Reprovado
- 3) Crie um programa que receba a idade de duas pessoas e print na tela qual é a 'pessoa mais velha
- 4) Crie um programa que receba o valor médio de 3 modelos de carro e indique qual é o carro mais caro e o mais barato.
- 5) Crie um programa que receba notas de 10 alunos e calcule sua média.

# Exercícios

6) Um estabelecimento está vendendo combustíveis com descontos variados. Para o etanol, se a quantidade comprada for maior ou igual 15 litros, o desconto será de 4% por litro. Caso contrário, será de 3% por litro. Para o diesel, se a quantidade comprada for maior ou igual a 15 litros, o desconto será de 5% por litro. Caso contrário, será de 3% por litro. O preço do litro de diesel é R\$ 2,00, o preço do litro de etanol é R\$ 1,70, o preço da gasolina é R\$ 4,50. Para gasolina maior ou igual a 20 L 3 % de desconto. Se não, sem desconto. Escreva um programa que leia a quantidade de litros vendidos e o tipo de combustível (E para etanol, D para diesel, G - Gasolina) e calcule o valor a ser pago pelo cliente. Tenha em mente algumas dicas:

- O valor do desconto será a multiplicação entre preço do litro, quantidade de litros e o valor do desconto.
- O valor a ser pago por um cliente será o resultado da multiplicação do preço do litro pela quantidade de litros menos o valor de desconto resultante do cálculo.

# Exercícios

- 7) Escreva um programa que calcule o preço a pagar pelo fornecimento de energia elétrica.
- Pergunte para o usuário a quantidade de KWh consumida, o tipo de instalação:
- R - Residência, I - Indústrias, C- Comércio.

Calcule o preço da energia com base na tabela a seguir

Tipo	Faixa (KWh)	Preço (R\$)
Residencial	Até 500	0,50
	Acima de 500	0,70
Comercial	Até 1000	0,65
	Acima de 1000	0,60
Industrial	Até 5000	0,55
	Acima de 5000	0,50

# Exercícios

- 8) Criar um programa que solicite ao usuário um número e calcule sua tabuada
- 9) Criar um programa que receba dois valores digitados pelo usuário e permita ao usuário escolher a operação :soma +, subtração -, multiplicação \* e divisão
- 10) Crie um programa que receba 20 valores de temperatura, calcule sua média, e imprima o maior e o menor valor de temperatura digitado pelo usuário.

# Obrigado!

Prof. Me Daniel Vieira

Email: [danielvieira2006@gmail.com](mailto:danielvieira2006@gmail.com)

Linkedin: Daniel Vieira

Instagram: Prof daniel.vieira95

