



WHILE & FOR

AULA 3 PPDM

Roteiro.

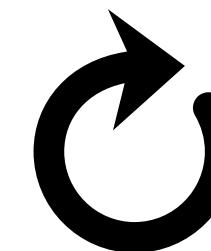
- O que aprendi ? 07:30 – 07:50.
- Exercício Aula Anterior. 07:50 – 08:20.
- While & Do While. 08:20 – 08:50.
- Quiz. 08:50 – 09:00.
- Descanso/Anotações. 09:15 – 09:25.
- Exercício While. 09:25 – 09:50.
- For & For Each. 09:50 – 10:20.
- Quiz. 10:20 – 10:40.
- Exercícios 10:40 – 11:30.



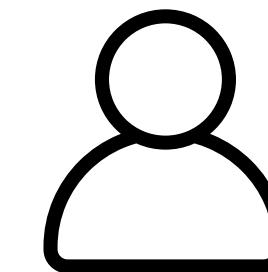
While



**REPETIR ATÉ QUE UMA
CONDIÇÃO SEJA
VERDADEIRA.**



**QUANDO NÃO SEI
QUANTAS VEZES DEVO
REPETIR O CÓDIGO.**



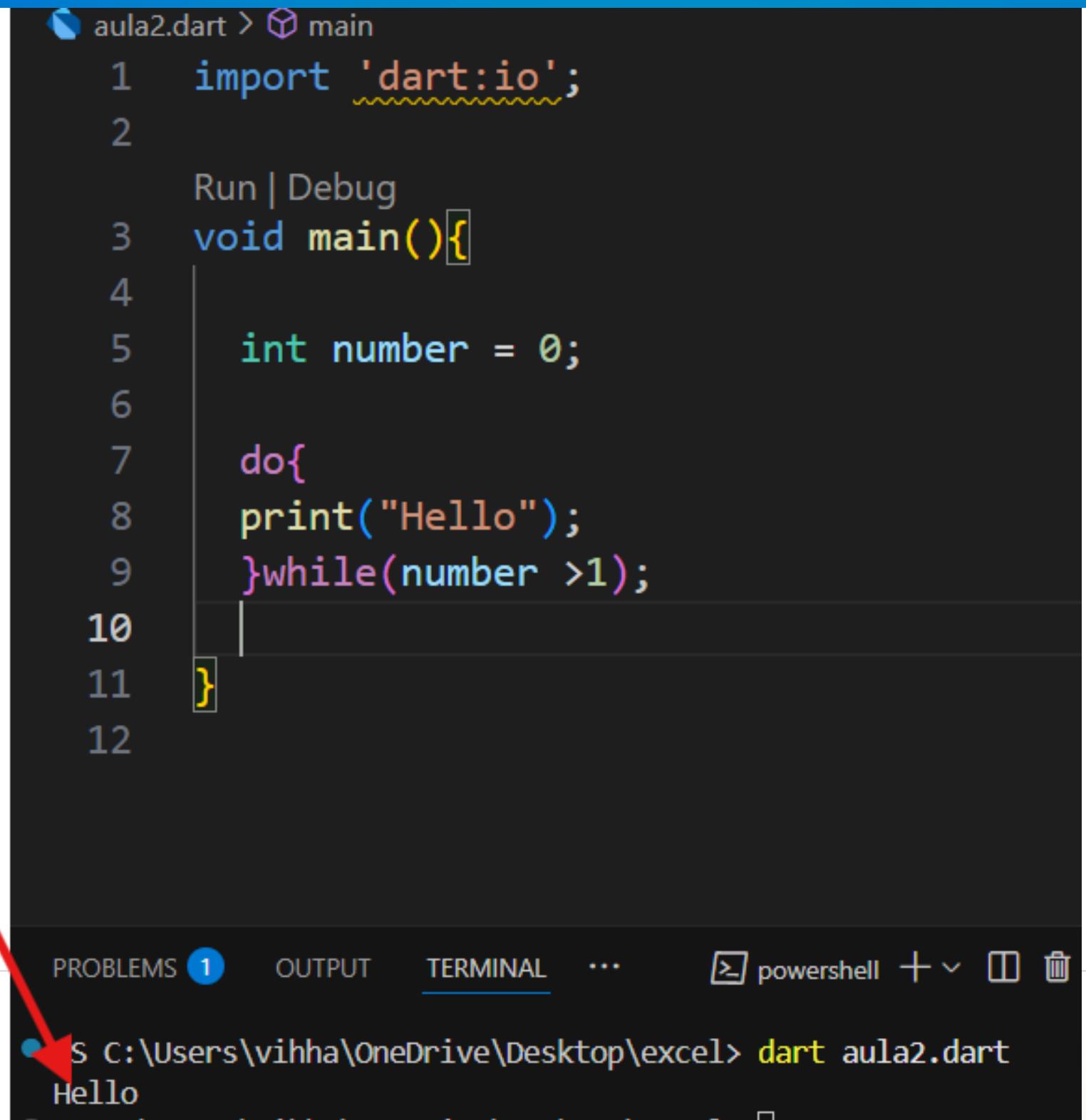
**QUANDO DEPENDE DA
RESPOSTA DO
USUÁRIO.**

Formas de fazer no dart.

*Testando a condição depois
da execução.
do while.*

*Testando a condição antes da
execução.
while.*

Exemplos.



The screenshot shows a code editor window with a dark theme. At the top, it displays the file path "aula2.dart > main". The code itself is as follows:

```
aula2.dart > main
1 import 'dart:io';
2
3 void main(){
4
5     int number = 0;
6
7     do{
8         print("Hello");
9     }while(number >1);
10
11 }
12
```

At the bottom of the editor, there is a navigation bar with tabs: PROBLEMS (1), OUTPUT, TERMINAL, ..., and a powershell icon. Below the editor, a terminal window shows the output of running the script:

```
s C:\Users\viha\OneDrive\Desktop\excel> dart aula2.dart
Hello
```

A red arrow points from the bottom-left towards the terminal output area.

Exemplos.

```
aula2.dart > main
1 import 'dart:io';
2
3     Run | Debug
4 void main() {
5     String? resposta;
6
7     do {
8         stdout.write("Você quer continuar? (sim/nao): ");
9         resposta = stdin.readLineSync();
10    } while (resposta != "nao");
11
12    print("Encerrando o programa...");
13 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

powershell + ▾

```
PS C:\Users\viha\OneDrive\Desktop\excel> dart aula2.dart
Você quer continuar? (sim/nao): nao
Encerrando o programa...
```

Exemplos.

The screenshot shows a code editor interface with a dark theme. At the top, there's a file navigation bar with icons for file operations like Open, Save, and Close. Below the bar, the current file is listed as 'aula2.dart > main'. The code itself is a Dart script:

```
1 import 'dart:io';
2
3 void main() {
4     int contador = 0;
5
6     while (contador < 5) {
7         print("Contador: $contador");
8         contador++;
9     }
10 }
```

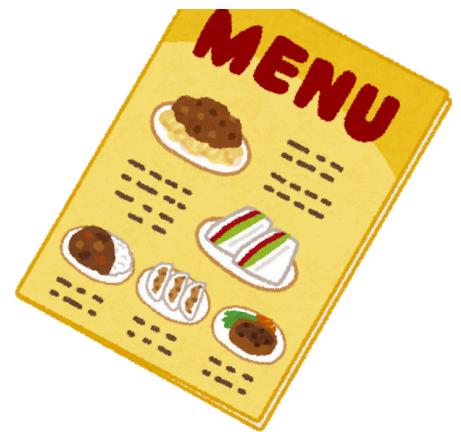
Line 7 has a yellow warning icon next to it. The code editor also includes a status bar at the bottom with tabs for PROBLEMS (1), OUTPUT, DEBUG CONSOLE, TERMINAL (which is underlined in blue), and PORTS. To the right of the tabs is a terminal window showing the output of running the script:

- PS C:\Users\viha\OneDrive\Desktop\excel> dart aula2.dart
- Você quer continuar? (sim/nao): nao
- Encerrando o programa...
- PS C:\Users\viha\OneDrive\Desktop\excel> dart aula2.dart

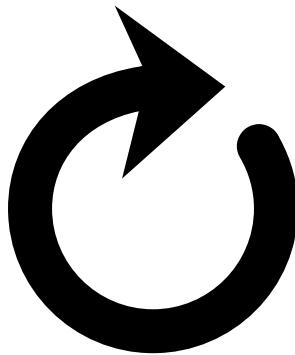
Below the terminal, the output of the script is displayed:

```
Contador: 0
Contador: 1
Contador: 2
Contador: 3
Contador: 4
```

Quando usar? do while

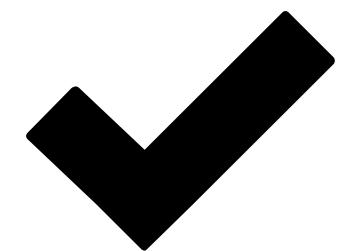


**MENUS INTERATIVOS
QUE DEVEM
APARECER PELO
MENOS UMA VEZ.**



**REPETIÇÕES QUE SÓ
PODEM SER
AVALIADAS APÓS UMA
PRIMEIRA EXECUÇÃO.**

Quando usar? while



**VALIDAÇÃO DE
ENTRADA ANTES DE
CONTINUAR**



**ESPERAR POR UMA
CONDIÇÃO
VERDADEIRA ANTES
DE INICIAR O BLOCO**



4GIFS.com

QUIZ TIME



Aplicações.

While.

Verifica a condição antes de executar o bloco de código.

- Se a condição for falsa logo no início, o código nunca será executado.
- Ideal quando não temos certeza se precisamos repetir algo.

Use quando você só quer que o código rode se a condição for verdadeira desde o início.

Exemplo: validação de entrada.

```
1 import 'dart:io'; //importa bibliotecas e funções da linguagem
2
3 Run | Debug
4 void main() {
5   print("Digite seu nome");
6   String? nome = stdin.readLineSync();
7
8   while(nome != "vini") {
9     print("Tente novamente");
10    print("Digite seu nome!!!");
11    nome = stdin.readLineSync();
12
13 }
14
15 print("Acesso liberado continuando!!!");
16
17
18
19
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
✖ : comando não encontrado
vinidev@pop-os:/media/vinidev/BC3EAC7F3EAC3474/DartParaAulas$ ^[[A
✖ : comando não encontrado
vinidev@pop-os:/media/vinidev/BC3EAC7F3EAC3474/DartParaAulas$ dart dowhileslide.dart
Menu de opções:
1 - Ver produtos
2 - Comprar
```

Do while.

Executa o bloco de código pelo menos uma vez antes de verificar a condição.

- Sempre executa uma vez (mesmo que a condição seja falsa depois).
- Ideal quando queremos garantir que algo seja exibido ou executado pelo menos uma vez.

Use quando você quer que algo seja exibido ou executado pelo menos uma vez, mesmo que a condição não continue válida depois.

Exemplo: Menu

```
1 import 'dart:io';
2
3 Run | Debug
4 void main() {
5   String opcao;
6
7   do {
8     print("Menu de opções:");
9     print("1 - Ver produtos");
10    print("2 - Comprar");
11    print("3 - Sair");
12    opcao = stdin.readLineSync()!;
13 } while (opcao != "3");
14
15 print("Programa encerrado.");
16 }
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
✖ : comando não encontrado
vinidev@pop-os:/media/vinidev/BC3EAC7F3EAC3474/DartParaAulas$ ^[[A
✖ : comando não encontrado
vinidev@pop-os:/media/vinidev/BC3EAC7F3EAC3474/DartParaAulas$ dart dowhileslide.dart
Menu de opções:
1 - Ver produtos
2 - Comprar
```

Exercicio

O MERCADINHO DO BAIRRO VAI USAR UM SISTEMA DE COMPRAS POR TELA TOUCH, E VOCÊ FOI CONTRATADO PARA FAZER ESSE PROGRAMA.

REGRAS DO SISTEMA:

1. O PROGRAMA COMEÇA MOSTRANDO OS ITENS DISPONÍVEIS.
2. DEPOIS ELE PEDE O CPF DO CLIENTE.
3. DEPOIS, O CLIENTE VAI DIGITAR:
 - O NOME DE UM ITEM
 - O VALOR DO ITEM
4. APÓS CADA ITEM, O PROGRAMA DEVE PERGUNTAR:
 - “DESEJA INCLUIR MAIS ITENS? (SIM/NÃO)”
5. ENQUANTO O CLIENTE DIGITAR “SIM”, O PROGRAMA DEVE CONTINUAR PEDINDO OS PRÓXIMOS ITENS.
6. QUANDO O CLIENTE DIGITAR “NÃO”, O PROGRAMA DEVE:
 - MOSTRAR TODOS OS ITENS COMPRADOS (EM FORMA DE TEXTO MESMO)
 - MOSTRAR O VALOR TOTAL DA COMPRA
7. DEPOIS, O CLIENTE DEVE ESCOLHER UMA DAS FORMAS DE PAGAMENTO:
 - 1 - À VISTA (COM 10% DE DESCONTO)
 - 2 - PARCELADO NO CARTÃO (COM 10% DE JUROS)
 - 3 - FIADO (COM 15% A MAIS PARA PAGAR NA PRÓXIMA COMPRA)
8. ENQUANTO O CLIENTE NÃO DIGITAR UMA OPÇÃO VÁLIDA (1, 2 OU 3), O PROGRAMA DEVE REPETIR A PERGUNTA.

FOR OFAMOSO “PARA”.

1

Quantidade definida.

Use FOR quando saber a quantidade de vezes que o programa deve repetir.

2

Variável “Observadora”.

Geralmente chamada de “i”. Serve para observar as mudanças dos dados na repetição.

3

Usar com Listas.

Muito usado “PARA” mostrar os dados de uma lista.

EXEMPLO

Quero que o usuário insira as 3 frutas preferidas dele.

The screenshot shows a code editor window with a dark theme. The file is named `forexemplo.dart`. The code is as follows:

```
forexemplo.dart > main
1 import 'dart:io';
2
3 void main() {
4     List frutas = [];
5
6     for (int i=0;i<3; i++) {
7         print("Digite sua fruta preferida");
8         String? fruta = stdin.readLineSync();
9         frutas.add(fruta);
10    }
11
12 print("Suas frutas preferidas são $frutas");
13
14
15
16
17 }
```

Below the code editor is a terminal window showing the execution of the script and its output:

```
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS
• vinidev@pop-os:/media/vinidev/BC3EAC7F3EAC3474/DartParaAulas$ dart forexemplo.dart
Digite sua fruta preferida
Melao
Digite sua fruta preferida
Abacaxi
Digite sua fruta preferida
Maracuja
Suas frutas preferidas são [Melao, Abacaxi, Maracuja]
○ vinidev@pop-os:/media/vinidev/BC3EAC7F3EAC3474/DartParaAulas$ 
```

FOR EACH

PARA CADA ITEM DE UMA LISTA FAÇA ALGO.....

um mosquito começa a me rodear
eu depois de tentar espantar ele pela
5º vez:



Usado para substituir o FOR em situações de iteração de dados de uma lista.

Para cara

FOR EACH

Quero que o programa mostre todos os gêneros de filmes disponíveis.

Ao invés disso.

```
foreach exemplo.dart > main
1 import 'dart:io';
2
3 Run | Debug
4 void main() {
5 List<String> generos_disponiveis = ["Ação","terror","aventura","romanc
6
7 for(String i in generos_disponiveis){
8   print("Genero disponivel $i");
9 }
10
11
12 }
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
• vinidev@pop-os:/media/vinidev/BC3EAC7F3EAC3474/DartParaAulas$ dart foreach exemplo.dart
Genero disponivel Ação
Genero disponivel terror
Genero disponivel aventura
Genero disponivel romance
○ vinidev@pop-os:/media/vinidev/BC3EAC7F3EAC3474/DartParaAulas$
```

Prefira isso.

```
foreach exemplo.dart > main
1 import 'dart:io';
2
3 Run | Debug
4 void main() {
5 List<String> generos_disponiveis = ["Ação","terror","aventura","romanc
6
7 // for(String i in generos_disponiveis){
8 //   print("Genero disponivel $i");
9 // }
10
11 generos_disponiveis.forEach((String i)=> print("Genero disponivel $i"))
12
13
14 }
15
16
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
• vinidev@pop-os:/media/vinidev/BC3EAC7F3EAC3474/DartParaAulas$ dart foreach exemplo.dart
Genero disponivel Ação
Genero disponivel terror
Genero disponivel aventura
Genero disponivel romance
○ vinidev@pop-os:/media/vinidev/BC3EAC7F3EAC3474/DartParaAulas$
```

MOMENTO

QUIZ



EXERCÍCIO 1.

Contexto:

Seu chefe está criando um sistema de simulação de pagamentos. O usuário digita o valor total da compra e a quantidade de parcelas que deseja.

Regras do sistema:

- O programa deve pedir:
 - O valor da compra
 - A quantidade de parcelas (ex: 3, 5, 10...)
- Depois, o sistema deve exibir um relatório com o valor de cada parcela.
- (Exemplo: “Parcela 1: R\$ 100.00, Parcela 2: R\$100.00 ,.....”)

EXERCÍCIO

2.

Contexto:

Você está ajudando a organizar uma lista de presença simples para uma sala de aula.

Regras do sistema:

- **O sistema tem uma lista pronta de nomes (use uma lista simples tipo: ["Ana", "Bruno", "Carlos"])**
- **Para cada nome da lista, o sistema deve perguntar:**
 - **“A pessoa Ana está presente? (sim/não)”**
- **No final, mostrar quantos alunos estão presentes.**

EXERCÍCIO

3.

Contexto:

Você está desenvolvendo um sistema para mostrar o cardápio de uma lanchonete.

Regras do sistema:

- **Existe uma lista fixa com os seguintes itens e preços:**
 - "Hambúrguer - R\\$ 12.00"
 - "Batata Frita - R\\$ 8.00"
 - "Refrigerante - R\\$ 5.00"
 - "Suco - R\\$ 6.00"
 - "Milkshake - R\\$ 10.00"
- **O sistema deve mostrar cada item do cardápio numerado, assim:**
 - 1 - Hambúrguer
 - 2 - Batata Frita
 - 3 - Refrigerante
 -
- **Todos produtos devem estar numerados de acordo com a ordem.**

OBRIGADO

P P D M

professor

Vinícius Almeida.