

Pertemuan 2
Pengantar Bahasa Pemrograman Dart



Dibuat Oleh:
Leon Shan Yoedha Adjie

POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
2025/2026



NAMA
NIM
KELAS

: Leon Shan Yoedha Adjie
: 2341720136
: 3E

Tugas Praktikum

1. Modifikasilah kode pada baris 3 di VS Code atau Editor Code favorit Anda berikut ini agar mendapatkan keluaran (*output*) sesuai yang diminta!

The image shows two screenshots of code editors. The top screenshot is from VS Code, showing a Dart file named `pertemuan2.dart` with the following code:

```
1 void main() {  
2   for (int i= 22; i > 1; i--) {  
3     print('My name is Yoedha, i am ${i-1} years old');  
4   }  
5 }
```

The bottom screenshot is from DartPad, showing the same code but with the name 'Yoedha' changed to 'Leon' in the print statement:

```
1 void main() {  
2   for (int i= 22; i > 1; i--) {  
3     print('My name is Leon, i am ${i-1} years old');  
4   }  
5 }
```

The output of the code in DartPad is shown on the right side of the editor:

```
My name is Leon, i am 21 years old  
My name is Leon, i am 20 years old  
My name is Leon, i am 19 years old  
My name is Leon, i am 18 years old  
My name is Leon, i am 17 years old  
My name is Leon, i am 16 years old  
My name is Leon, i am 15 years old  
My name is Leon, i am 14 years old  
My name is Leon, i am 13 years old  
My name is Leon, i am 12 years old  
My name is Leon, i am 11 years old  
My name is Leon, i am 10 years old  
My name is Leon, i am 9 years old  
My name is Leon, i am 8 years old  
My name is Leon, i am 7 years old  
My name is Leon, i am 6 years old  
My name is Leon, i am 5 years old  
My name is Leon, i am 4 years old  
My name is Leon, i am 3 years old  
My name is Leon, i am 2 years old  
My name is Leon, i am 1 years old
```



NAMA : Leon Shan Yoedha Adjie
NIM : 2341720136
KELAS : 3E

2. Mengapa sangat penting untuk memahami bahasa pemrograman Dart sebelum kita menggunakan framework Flutter ? Jelaskan!

Jawab:

Sangat penting untuk memahami bahasa pemrograman Dart sebelum menggunakan framework Flutter karena Dart adalah inti dan dasar fundamental dari Flutter. Dengan mempelajari Dart, seorang pengembang akan menguasai konsep-konsep inti seperti tipe data, operator, dan pemrograman berorientasi objek (OOP) yang digunakan secara ekstensif dalam pengembangan aplikasi Flutter. Penguasaan ini tidak hanya menjadi fondasi untuk dapat bekerja secara efektif, tetapi juga memberikan kenyamanan dan kepercayaan diri untuk bereksperimen lebih jauh, sehingga memungkinkan terciptanya aplikasi seluler yang luar biasa.

3. Rangkumlah materi dari codelab ini menjadi poin-poin penting yang dapat Anda gunakan untuk membantu proses pengembangan aplikasi mobile menggunakan framework Flutter.

Jawab:

- **Dart adalah Fondasi Utama Flutter:** Memahami Dart sangatlah esensial karena Dart merupakan bahasa inti yang digunakan oleh Flutter. Penguasaan Dart akan mempermudah pengembangan dan pemecahan masalah dalam proyek Flutter.
- **Konsep Dasar Pemrograman:** Codelab ini menekankan pentingnya memahami dasar-dasar Dart, termasuk:
 - **Tipe Data dan Operator:** Mengetahui tipe data bawaan dan berbagai operator (aritmatika, perbandingan, logika) adalah dasar untuk menulis logika aplikasi.
 - **Pemrograman Berorientasi Objek (OOP):** Flutter sangat bergantung pada konsep OOP. Memahami cara kerja fungsi dan metode dalam konteks OOP di Dart akan sangat membantu.
- **Fitur-Fitur Unggulan Dart:** Bahasa Dart memiliki beberapa fitur yang mendukung pengembangan yang produktif:
 - **Static Typing:** Membantu menemukan *error* lebih awal saat proses pengembangan, bukan saat aplikasi sudah berjalan.
 - **Garbage Collection:** Manajemen memori otomatis yang membuat developer tidak perlu pusing mengalokasikan dan membebaskan memori secara manual.
 - **Tooling yang Produktif:** Dart dilengkapi dengan berbagai alat bantu yang mempercepat proses pengembangan.
- **Mekanisme Kompilasi JIT dan AOT:** Dart menggunakan dua jenis kompilasi yang sangat berguna untuk Flutter:
 - **Just-In-Time (JIT):** Digunakan selama pengembangan untuk mengaktifkan fitur *Hot Reload*, yang memungkinkan melihat perubahan kode secara instan tanpa perlu me-restart aplikasi.
 - **Ahead-Of-Time (AOT):** Digunakan saat aplikasi akan dirilis (*release*) untuk memastikan performa yang cepat dan waktu startup yang singkat di perangkat pengguna.
- **Struktur Program Dasar:** Setiap program Dart dimulai dari fungsi `main()`. Memahami ini adalah langkah pertama untuk menulis dan menjalankan kode Dart.
- **Mulai Cepat dengan DartPad:** Codelab ini merekomendasikan penggunaan DartPad, sebuah editor kode online.

4. Buatlah penjelasan dan contoh eksekusi kode tentang perbedaan *Null Safety* dan *Late variabel* !

Jawab:



NAMA : Leon Shan Yoedha Adjie
NIM : 2341720136
KELAS : 3E

Null safety: Null Safety adalah fitur keamanan tipe data yang memungkinkan kompiler untuk mencegah error yang terkait dengan nilai null. Dengan null safety, variabel tidak bisa berisi nilai null kecuali secara eksplisit dinyatakan dapat menerima nilai null yang ditandai dengan simbol '?'.
Contoh:

```
6 // Null Safety
7
8 // Non-nullable: Harus memiliki nilai (tidak null)
9 String nama = "Yoedha";
10
11 // Nullable: Boleh berisi null (ditandai dengan ?)
12 String? tinggi = null;
13
14 // Mengakses nilai nullable dengan aman
15 print(nama.toUpperCase()); // YOEDHA (aman)
16 print(tinggi?.toUpperCase()); // null (aman)
17
18 // Menetapkan nilai ke nullable
19 tinggi = "181";
20 print(tinggi); // 181
21
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS

Connecting to VM Service at ws://127.0.0.1:58472/bpNDIvLJpDw=/ws
Connected to the VM Service.
YOEDHA
null
181
3



NAMA : Leon Shan Yoedha Adjie
NIM : 2341720136
KELAS : 3E

Late variabel: adalah fitur yang memungkinkan penundaan inisialisasi variabel non-nullable. Ini berguna ketika variabel tidak bisa diinisialisasi pada saat deklarasi tetapi dijamin akan diinisialisasi sebelum digunakan dan ditandai dengan *keyword* 'late'.

Contoh:

```
22 // Late Variabel
23 late String pesan;
24
25 // Fungsi sederhana
26 void setPesan() {
27     print("Menyiapkan pesan...");
28     pesan = "Halo Dunia!";
29 }
30
31 // Pesan belum diinisialisasi di sini
32 // print(pesan); // ERROR: LateInitializationError
33
34 // Inisialisasi pesan
35 setPesan();
36
37 // Sekarang aman untuk diakses
38 print(pesan); // Halo Dunia!
39
40 // Late dengan initializer - dijalankan saat pertama kali diakses
41 late String waktu = ambilWaktu();
42 print("Sebelum mengakses waktu");
43 print(waktu); // Initializer berjalan saat ini
44 print(waktu); // Initializer tidak berjalan lagi
45
46
47 String ambilWaktu() {
48     print("Mengambil waktu sekarang...");
49     return "14:30";
50 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS

Connecting to VM Service at ws://127.0.0.1:55870/P70S9GX9u1c=/ws
Connected to the VM Service.
Menyiapkan pesan...
Halo Dunia!
Sebelum mengakses waktu
Mengambil waktu sekarang...
2 14:30
Exited.