

**LAPORAN  
PRAKTIKUM PEMROGRAMAN MOBILE**

**JOBSHEET 2**



**Oleh :**

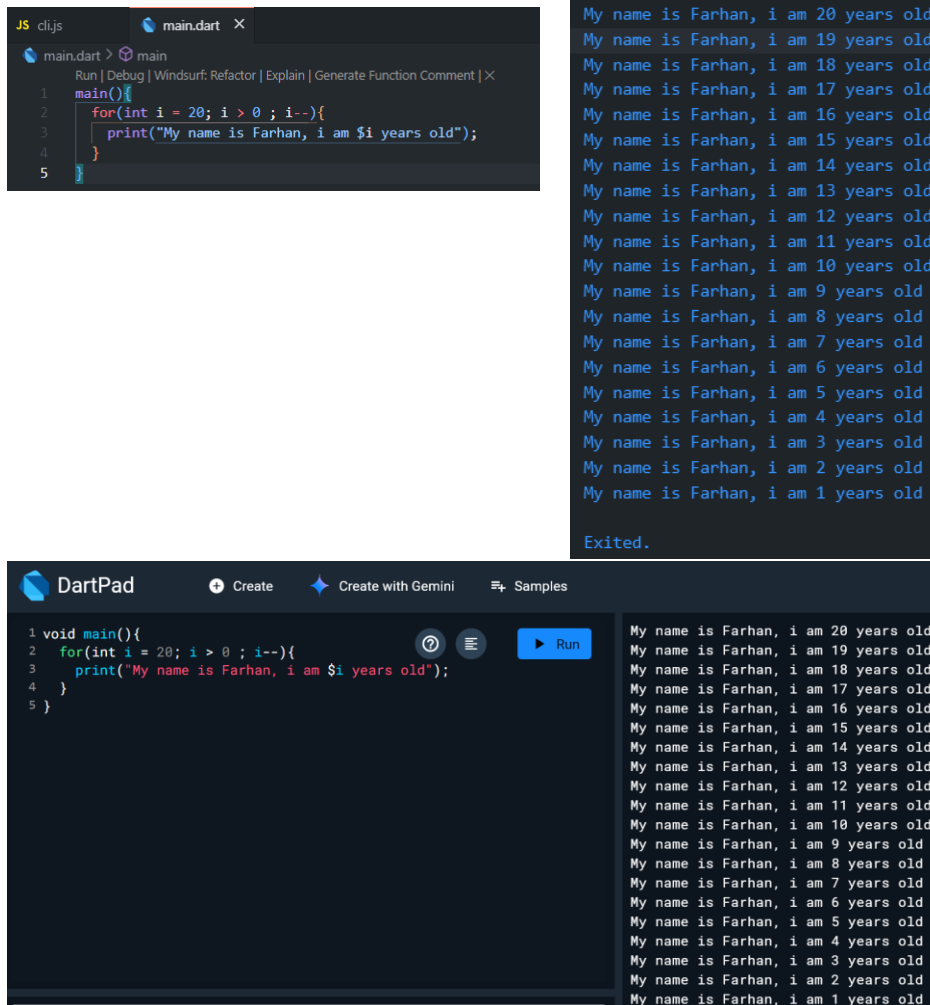
Farhan Mawaludin

2341720258

**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI MALANG  
2025/2026**

## Tugas Praktikum

1. Modifikasilah kode untuk menampilkan nama dan usia mulai dari usia anda sampai 1



The top screenshot shows a code editor with the following Dart code:

```
main.dart x
main.dart > main
Run | Debug | Windsurf: Refactor | Explain | Generate Function Comment | X
1 main() {
2   for(int i = 20; i > 0 ; i--){
3     print("My name is Farhan, i am $i years old");
4   }
5 }
```

The output of the code is:

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

The bottom screenshot shows the DartPad interface with the same code and output.

```
1 void main(){
2   for(int i = 20; i > 0 ; i--){
3     print("My name is Farhan, i am $i years old");
4   }
5 }
```

The output of the code is:

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

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

**Jawab :** Karna Bahasa dasar yang digunakan pada flutter adalah dart. Oleh karena itu sebaiknya memahami dart terlebih dahulu sebelum menggunakan framework flutter

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

**Jawab :**

**a. Dart = Bahasa utama Flutter**

- i. Semua kode Flutter ditulis pakai Dart.
- ii. Jadi sebelum bikin aplikasi Flutter, harus paham dasar-dasar Dart.

**b. Kenapa Dart dipilih untuk Flutter?**

- i. Cepat & efisien, ada fitur hot reload (ubah kode langsung terlihat di aplikasi).
- ii. Mudah dipahami, mirip bahasa pemrograman lain (Java, C, JavaScript).
- iii. Lintas platform, bisa dipakai untuk mobile, web, dan desktop.
- iv. Modern & aman, ada sistem pengecekan tipe data untuk mencegah error.

**c. Cara kerja Dart**

- i. JIT (Just-In-Time), dipakai saat ngoding, supaya cepat testing & debugging.
- ii. AOT (Ahead-Of-Time), dipakai saat aplikasi sudah jadi, supaya lebih cepat & ringan.

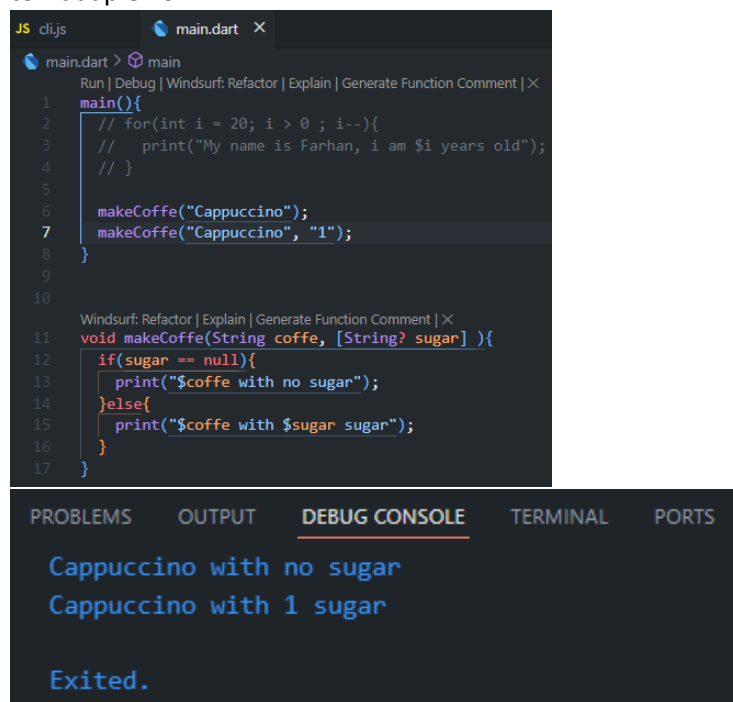
**d. Struktur dasar Dart**

- i. Semua data di Dart itu objek (tidak ada tipe primitif).
- ii. Dart mendukung OOP: class, object, method, inheritance (turunan), dll.
- iii. Ada operator umum (+, -, \*, /, ==, !=, >, <, &&, ||).

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

**Jawab :**

**Null Safety**, memastikan bahwa variabel tidak dapat berisi nilai null kecuali ditentukan secara eksplisit. Fitur ini membantu mencegah kesalahan null-dereferensi runtime dengan menangkapnya pada waktu kompilasi, membuat kode lebih kuat dan tidak terlalu rentan terhadap error.

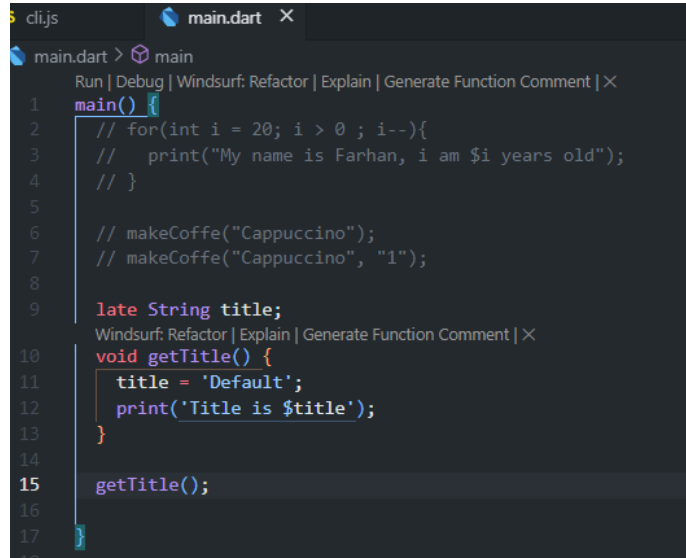


```
JS cli.js main.dart x
main.dart > main
Run | Debug | Windsurf: Refactor | Explain | Generate Function Comment | X
1 main(){
2   // for(int i = 20; i > 0 ; i--){
3   //   print("My name is Farhan, i am $i years old");
4   // }
5
6   makeCoffe("Cappuccino");
7   makeCoffe("Cappuccino", "1");
8 }
9
10
11 Windsurf: Refactor | Explain | Generate Function Comment | X
12 void makeCoffe(String coffe, [String? sugar ]){
13   if(sugar == null){
14     print("$coffe with no sugar");
15   }else{
16     print("$coffe with $sugar sugar");
17   }
18 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Cappuccino with no sugar  
Cappuccino with 1 sugar  
  
Exited.

**Late Variable**, digunakan untuk mendeklarasikan variabel yang akan diinisialisasi nanti dalam kode. Ini sangat berguna untuk variabel non-nullable yang tidak dapat diinisialisasi pada saat deklarasi tetapi dijamin akan diberi nilai sebelum diakses.



```
cli.js  main.dart X
main.dart > main
Run | Debug | Windsurf: Refactor | Explain | Generate Function Comment | X
1  main() {
2    // for(int i = 20; i > 0 ; i--){
3    //   print("My name is Farhan, i am $i years old");
4    // }
5
6    // makeCoffe("Cappuccino");
7    // makeCoffe("Cappuccino", "1");
8
9    late String title;
10   Windsurf: Refactor | Explain | Generate Function Comment | X
11   void getTitle() {
12     title = 'Default';
13     print('Title is $title');
14   }
15   getTitle();
16
17 }
```