

JOBSHEET 2
“Pengantar Bahasa Pemrograman Dart”

Mata Kuliah [Pemrograman Mobile]

Dosen Pengampu: Ade Ismail, S.Kom., M.TI.



Nama	: Wardanadira Pinkan Dwiyuwanda
NIM	: 2241760116
Kelas	: SIB-3D
No Absen	: 26

PROGRAM STUDI SISTEM INFORMASI BISNIS
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2024

SOAL

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

```
1 void main() {  
2     for (int i = 0; i < 10; i++) {  
3         print('hello ${i + 2}');  
4     }  
5 }
```

Output yang diminta:

```
Console  
  
Nama saya adalah Fulan, sekarang berumur 18  
Nama saya adalah Fulan, sekarang berumur 17  
Nama saya adalah Fulan, sekarang berumur 16  
Nama saya adalah Fulan, sekarang berumur 15  
Nama saya adalah Fulan, sekarang berumur 14  
Nama saya adalah Fulan, sekarang berumur 13  
Nama saya adalah Fulan, sekarang berumur 12  
Nama saya adalah Fulan, sekarang berumur 11  
Nama saya adalah Fulan, sekarang berumur 10  
Nama saya adalah Fulan, sekarang berumur 9
```

Jawab:

KODE PROGRAM

```
void main() {  
    for (int i = 18; i >= 9; i--) {  
        print('Nama saya adalah Fulan, sekarang berumur $i');  
    }  
}
```

HASIL

```
Nama saya adalah Fulan, sekarang berumur 18  
Nama saya adalah Fulan, sekarang berumur 17  
Nama saya adalah Fulan, sekarang berumur 16  
Nama saya adalah Fulan, sekarang berumur 15  
Nama saya adalah Fulan, sekarang berumur 14  
Nama saya adalah Fulan, sekarang berumur 13  
Nama saya adalah Fulan, sekarang berumur 12  
Nama saya adalah Fulan, sekarang berumur 11  
Nama saya adalah Fulan, sekarang berumur 10  
Nama saya adalah Fulan, sekarang berumur 9
```

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

Jawab:

Memahami bahasa pemrograman Dart sangat penting sebelum menggunakan framework Flutter karena Dart merupakan bahasa dasar yang digunakan untuk membuat aplikasi Flutter. Dengan menguasai Dart, kita dapat lebih mudah memahami cara kerja Flutter, termasuk cara mengelola struktur data, logika pemrograman, dan konsep-konsep penting seperti asinkroni dan manajemen status. Selain itu, pemahaman yang kuat tentang Dart juga akan memfasilitasi proses debugging dan

optimalisasi aplikasi. ini memungkinkan kami memaksimalkan fitur Flutter. Jadi menguasai Dart adalah langkah awal yang penting untuk menjadi pengembang Flutter yang efektif dan efisien.

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

Jawab:

Pengantar Bahasa Pemrograman Dart

Bahasa dalam Dart adalah inti dari framework flutter dengan membutuhkan bahasa modern tinggi agar bisa memberi pengalaman baik pada pengembang. Bahasa Dart meliputi:

- Productive tooling (tool yang digunakan untuk menganalisis kode, plugin IDE, dan ekosistem paket besar)
- Garbage collection (digunakan untuk mengelola atau menangani objek yang tidak digunakan pada memori)
- Type annotations “optional” (digunakan untuk keamanan dan konsistensi mengontrol data dalam aplikasi)
- Statically Typed (digunakan untuk menganalisis types saat runtime dan menemukan bug dalam kode)
- Portability (bahasa dart yang dapat dikompilasi secara native ke kode Advanced RISC Machines (ARM) dan x86).

Kode Dart bisa dieksekusi dengan menggunakan macam-macam bahasa Dart yaitu:

- Runtime systems
- Dart core libraries
- Garbage collectors

Eksekusi Dart dibagi menjadi dua yaitu dengan menggunakan kompilasi Just-In-Time (JIT) dan kompilasi Ahead-Of-Time (AOT)

- Kompilasi JIT adalah tempat kode berupa Dart VM yang memuat dan mengkompilasi kode sumber ke kode mesin asli dengan menjalankan kode pada command line.
- Kompilasi AOT adalah sama seperti diatas. Cuma terdapat perbedaan pada sistem kerja Dart VM pada AOT. Dart VM menggunakan sistem runtime Dart.

DART OPERATORS

- `x == y` (metode untuk memanggil)
- `x.==(y)` (metode untuk melakukan perbandingan kesetaraan)

ARITHMETIC OPERATORS

- `+` (tambahan)
- `-` (pengurangan)
- `*` (perkalian)
- `/` (pembagian)
- `~/` (pembagian bilangan bulat) Di Dart, setiap pembagian sederhana dengan `/` menghasilkan nilai double. Untuk mendapatkan nilai bilangan bulat.
- `%` (operasi modulus (sisa bagi dari bilangan bulat))

- - (negasi (yang membalikkan suatu nilai))

SHORTCURT OPERATOR (+=, -=, *=, /=, dan ~/=)

INCREMENT DAN DECREMENT OPERATOR

- ++var atau var++ untuk menambah nilai variabel var sebesar 1
- --var atau var-- untuk mengurangi nilai variabel var sebesar 1

EQUALITY DAN RELATIONAL OPERATOR

- == untuk memeriksa apakah operan sama
- != untuk memeriksa apakah operan berbeda

Untuk melakukan pengujian relasional, maka gunakan operator sebagai berikut:

- > memeriksa apakah operan kiri lebih besar dari operan kanan
- < memeriksa apakah operan kiri lebih kecil dari operan kanan
- >= memeriksa apakah operan kiri lebih besar dari atau sama dengan operan kanan
- <= memeriksa apakah operan kiri kurang dari atau sama dengan operan kanan

LOGICAL OPERATOR

- !expression negasi atau kebalikan hasil ekspresi—yaitu, true menjadi false dan false menjadi true.
- || menerapkan operasi logika OR antara dua ekspresi.
- && menerapkan operasi logika AND antara dua ekspresi.