

LAPORAN CODE LAB 2
PEMROGRAMAN MOBILE



Oleh:

ALBANI RAJATA MALIK

2241760080/06

SIB 3E

PROGAM STUDI D-IV SISTEM INFORMASI BISNIS
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG

Soal 1

Kode Program

```
void main() {  
    for (int i = 18; i > 8; i--) {  
        print('Nama saya adalah Albani, sekarang berumur $i');  
    }  
}
```

Hasil

```
Connecting to VM Service at ws://127.0.0.1:20178/YLo120sL3JM=/ws  
Nama saya adalah Albani, sekarang berumur 18  
Nama saya adalah Albani, sekarang berumur 17  
Nama saya adalah Albani, sekarang berumur 16  
Nama saya adalah Albani, sekarang berumur 15  
Nama saya adalah Albani, sekarang berumur 14  
Nama saya adalah Albani, sekarang berumur 13  
Nama saya adalah Albani, sekarang berumur 12  
Nama saya adalah Albani, sekarang berumur 11  
Nama saya adalah Albani, sekarang berumur 10  
Nama saya adalah Albani, sekarang berumur 9  
  
Exited.
```

Soal 2

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

Karena semua pengembangan framework Flutter melibatkan pengetahuan/fitur mendalam dengan bahasa Dart; Kode aplikasi, kode plugin, dan manajemen dependensi semuanya menggunakan bahasa Dart beserta fitur-fiturnya. Jadi dart merupakan kunci untuk menguasai Flutter secara menyeluruh karena Flutter menggunakan bahasa dart untuk sebagai bahasa dasar dalam membangun frameworknya.

Soal 3

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

Dart adalah inti dari framework Flutter. Dimana Dart memiliki fitur-fitur bahasa pemrograman terkini seperti : productive tooling, garbage collection, type annotation, statically typed, dan portability. Dart diluncurkan pada tahun 2011, dan dari sejak Dart telah merilis versi stabilnya pada tahun 2013 dengan perubahan besar termasuk dalam rilis Dart 2.0 menjelang akhir 2018. Seperti kebanyakan bahasa modern, Dart dirancang untuk object-

oriented (OO). Sesuai prinsip OO memastikan bahwa Dart memiliki fitur encapsulation, inheritance, composition, abstraction, dan polymorphism.

Dart Operator : Di Dart, operator tidak lebih dari method yang didefinisikan dalam class dengan sintaks khusus. Jadi, ketika Anda menggunakan operator seperti `x == y`, seolah-olah Anda sedang memanggil `x.==(y)` metode untuk melakukan perbandingan kesetaraan.

Arithmetic operator : Sama seperti operator aritmatik lainnya, contoh operator aritmatik pada dart adalah : '+' untuk tambahan, '-' untuk pengurangan, '*' untuk perkalian, '/' untuk pembagian, '~/' untuk pembagian bilangan bulat, dll.

Increment and decrement operator : Operator penambahan dan pengurangan juga merupakan operator umum dan diimplementasikan pada angka, sebagai berikut:

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

Equality and relational operators : Persamaan operator Dart dijelaskan sebagai berikut:

- `'=='` 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 operators : Operator logika di Dart adalah operator yang diterapkan pada operan bool; bisa berupa variabel, ekspresi, atau kondisi.. Operator logika yang disediakan adalah sebagai berikut:

- `!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.

Main function Seperti kebanyakan bahasa modern, Dart menggunakan function dan method sebagai cara untuk memisahkan kode. Sebuah function atau method adalah potongan kode yang menerima beberapa data, eksekusi kode, dan kemudian mengembalikan beberapa data (return).