

LAPORAN
Jobsheet-2 : PENGANTAR BAHASA PEMROGRAMAN DART
TUGAS PEMROGRAMAN MOBILE

Dibimbing oleh: Bapak Ade Ismail, S.Kom., M.TI.



Disusun oleh:
ANARADI OCTA LAVECHIA
2241760007 / SIB-3D

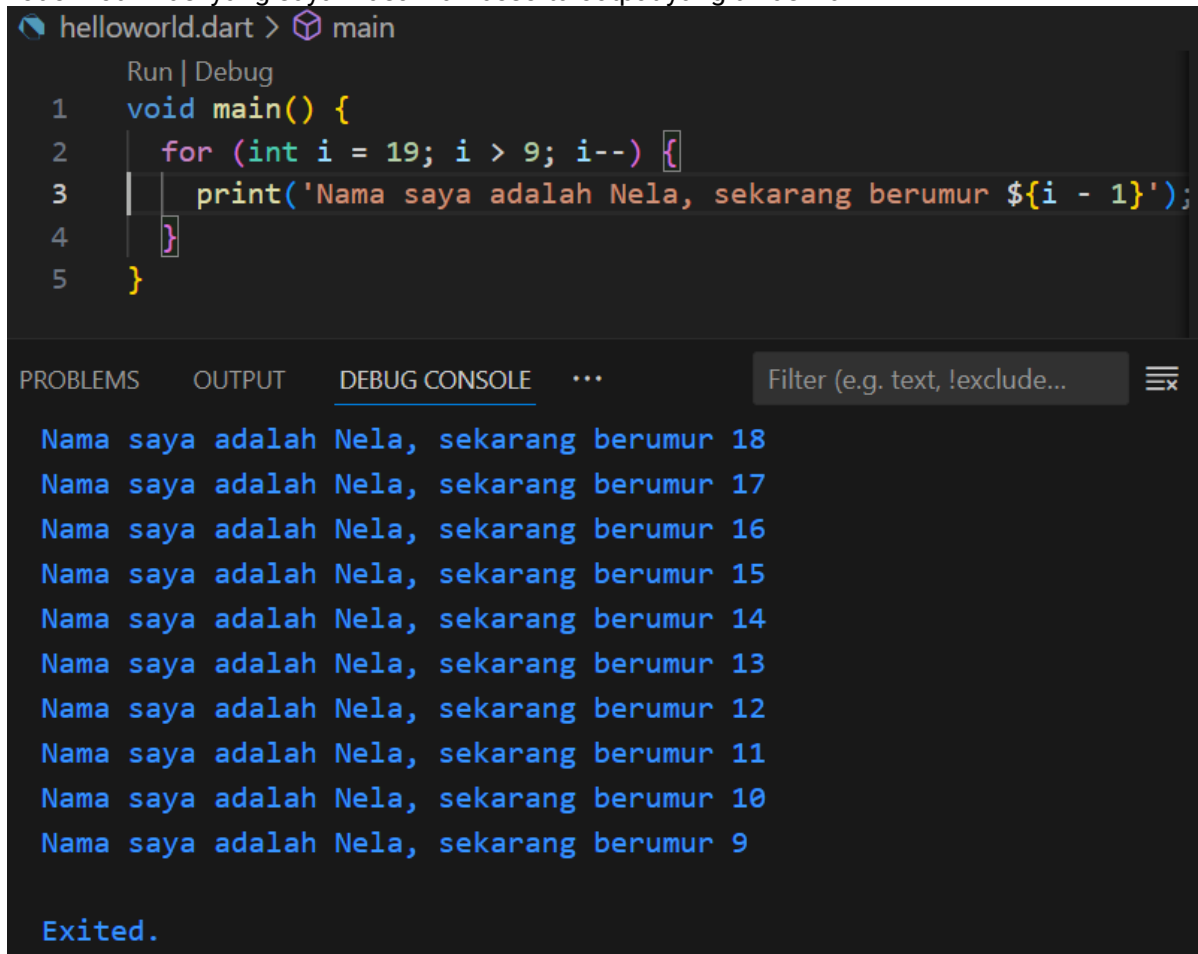
JURUSAN TEKNOLOGI INFORMASI
PRODI D-IV SISTEM INFORMASI BISNIS
POLITEKNIK NEGERI MALANG
2024

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 }
```

Kode modifikasi yang saya masukkan beserta output yang dihasilkan:



The screenshot shows the VS Code editor with a file named `helloworld.dart`. The code is as follows:

```
1 void main() {  
2   for (int i = 19; i > 9; i--) {  
3     print('Nama saya adalah Nela, sekarang berumur ${i - 1}');  
4   }  
5 }
```

The output window shows the following results:

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

Soal 2

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

Pentingnya untuk memahami bahasa pemrograman Dart sebelum menggunakan Flutter berdasarkan search dan membaca dari beberapa artikel di Google yaitu dikarenakan Dart adalah bahasa resmi yang dirancang untuk digunakan membangun aplikasi Flutter. Jika sudah memahami Dart, maka dapat menulis kode dengan lebih efisien, memanfaatkan fitur-fitur bahasa seperti asynchronous programming, dan memahami struktur aplikasi Flutter. Hal ini memungkinkan pengguna untuk memecahkan masalah, mengoptimalkan performa, dan mengembangkan aplikasi yang lebih baik secara keseluruhan.

Soal 3

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

Di codelab ini mempelajari tentang:

a. Getting started with Dart

Pengembangan framework Flutter melibatkan pengetahuan/fitur mendalam dengan bahasa Dart; Kode aplikasi, kode plugin, dan manajemen dependensi semuanya menggunakan bahasa Dart beserta fitur-fiturnya. Dart bertujuan untuk menggabungkan kelebihan-kelebihan dari sebagian besar bahasa tingkat tinggi dengan fitur-fitur bahasa pemrograman terkini, antara lain:

- Productive tooling untuk menganalisis kode, plugin IDE, dan ekosistem paket yang besar.
- Garbage collection untuk mengelola atau menangani dealokasi memori (terutama memori yang ditempati oleh objek yang tidak lagi digunakan).
- Type annotations (opsional) untuk keamanan dan konsistensi dalam mengontrol semua data dalam aplikasi.
- Statically typed (opsional) untuk menganalisis types saat runtime. Fitur ini penting untuk menemukan bug selama kompilasi kode.
- Portability: bahasa Dart tidak hanya untuk web (yang dapat diterjemahkan ke JavaScript) tetapi juga dapat dikompilasi secara native ke kode Advanced RISC Machines (ARM) dan x86.

b. The evolution of Dart

Dart diluncurkan pada tahun 2011 dan mulai berkembang sejak saat itu. Dart merilis versi stabilnya pada tahun 2013, dengan perubahan besar termasuk dalam rilis Dart 2.0 menjelang akhir 2018

c. How Dart works

Cara mengeksekusi Dart dapat dilakukan dengan dua cara, seperti yang dijelaskan berikut ini Dart virtual machines (VMs) dan JavaScript compilations. Eksekusi kode Dart dapat beroperasi dalam dua mode yaitu:

- Kompilasi Just In-Time (JIT) adalah tempat kode sumber dikompilasi sesuai kebutuhan—Just in time. Dart VM memuat dan mengkompilasi kode sumber ke kode mesin asli (native).
- Kompilasi Ahead-Of-Time (AOT) adalah dimana Dart VM dan kode yang dikompilasi sebelumnya, VM bekerja lebih seperti sistem runtime Dart, yang menyediakan garbage collector dan metode-metode native dari Dart software development kit (SDK) pada aplikasi.

d. Introducing the structure of the Dart language

Dart menyediakan sebagian besar operator standar untuk memanipulasi variabel; built-in types adalah yang paling umum ditemukan dalam bahasa pemrograman tingkat tinggi. Control flow dan function sangat mirip dengan bahasa pemrograman lainnya.

Dart dirancang untuk object-oriented (OOP) yang merupakan konsep objek yang menyimpan kedua data (disebut fields) dan kode (disebut methods). Objek-objek ini dibuat dari cetak biru yang disebut class yang mendefinisikan field dan method yang akan dimiliki oleh sebuah objek.

Di Dart, operator tidak lebih dari method yang didefinisikan dalam class dengan sintaks khusus. Dart hadir dengan banyak operator typical yang bekerja seperti banyak

bahasa pemrograman lainnya untuk perhitungan arithmetic, increment dan decrement, equality dan relational serta logical operators.

e. Hands-on with Dart

Cara termudah untuk memulai pemrograman Dart adalah dengan menggunakan kakas DartPad, yang dapat Anda akses di tautan <https://dartpad.dev/?sample=hello-world> pada web tersebut dapat melakukan modifikasi kode dan running program. Jika memilih untuk menjalankan kode tersebut secara lokal di PC/Laptop, simpan kode ke file Dart dan eksekusi dengan Dart di terminal—misalnya, simpan ke file bernama `hello_world.dart` dan kemudian run perintah `dart hello_world.dart`. Ini akan mengeksekusi fungsi main dari kode Dart tersebut.