

PRAKTIKUM PEMROGRAMAN PERANGKAT BERGERAK

PERTEMUAN KE-3

PENGENALAN DART



Disusun Oleh:

Aorinka Anendya Chazanah / 2211104013

S1 SE-06-01

PROGRAM STUDI S1 SOFTWARE ENGINEERING

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

GUIDED

Apa itu Dart?

Dart adalah bahasa pemrograman yang dikembangkan oleh Google untuk pembuatan aplikasi multiplatform, termasuk mobile, desktop, dan web. Bahasa ini modern dan kaya fitur, serta memiliki banyak kesamaan dengan bahasa populer lainnya seperti Java, C#, JavaScript, Swift, dan Kotlin.

Contoh kode:

```
void main() {  
    print('Praktikum Perangkat Bergerak');  
}
```

Karakteristik Dart:

- **Statically typed**

```
var name = 'Praktikum PPB';  
String language = 'Dart';
```

- **Type inference**

```
Hello Praktikum PPB. Welcome to Dart!
```

- **String interpolation**

```
print('Hello $name. Welcome to $language!');
```

- **Multi-paradigm**

1. Mendukung paradigma Object-Oriented Programming (OOP), dapat membuat program terstruktur yang memiliki konsep object, classes, properties, constructor, dll.
2. Functional Programming (FP), dapat menerapkan fitur-fitur seperti higher-order function, recursion, first-class citizen, dll.

A. Variable

a. Variabel dengan var

var adalah cara untuk mendeklarasikan variabel tanpa menentukan tipe datanya secara eksplisit. Dart akan secara otomatis menentukan tipe data berdasarkan nilai yang diberikan.

Contoh Kode

```
void main() {  
    // VARIABLE  
    // Menggunakan var  
    var name = "Alice"; // Tipe data String  
    var age = 25; // Tipe data Integer  
    print("Nama: $name, Usia: $age");  
}
```

b. Type Annotation

Type annotation adalah cara untuk mendeklarasikan variabel dengan menentukan tipe datanya secara eksplisit.

Contoh Kode

```
void main() {  
    // Type annotation  
    String nama = "Bob"; // Tipe data String  
    int usia = 30; // Tipe data Integer  
    print("Nama: $nama, Usia: $usia");  
}
```

c. Multiple variable

Dart memungkinkan Anda untuk mendeklarasikan beberapa variabel sekaligus dengan tipe yang sama.

Contoh Kode

```
void main() {  
    // Multiple variable  
    String firstName, lastName; // Tipe data String  
    firstName = "Charlie";  
    lastName = "Brown";  
    print("Nama Lengkap: $firstName $lastName");  
}
```

B. Statement Control

a. IF-ELSE Statement

Untuk memproses suatu kondisi dalam program, kita dapat menggunakan ekspresi if. Jika hasil dari ekspresi tersebut bernilai true (benar), maka blok kode di dalam if akan dijalankan. Namun, jika kondisi tersebut tidak terpenuhi atau bernilai false (salah), kita dapat menggunakan ekspresi else untuk mengeksekusi blok kode alternatif.

Contoh Kode

```
void main() {  
    // IF ELSE  
    // if-else statement  
    var openHours = 8;  
    var closedHours = 21;  
    var now = 17;  
  
    if (now > openHours && now < closedHours) {  
        print("Hello, we're open");  
    } else {  
        print("Sorry, we've closed");  
    }  
}
```

b. Switch-case Statement

```
void main() {  
    var day = 3; // Misalkan 1 = Senin, 2 = Selasa, dst.  
    switch (day) {  
        case 1:  
            print("Senin");  
            break;  
        case 2:  
            print("Selasa");  
            break;  
        case 3:  
            print("Rabu");  
            break;  
        case 4:  
            print("Kamis");  
            break;  
        case 5:  
            print("Jumat");  
            break;  
        case 6:  
            print("Sabtu");  
            break;  
        case 7:  
            print("Minggu");  
            break;  
        default:  
            print("Hari tidak valid");  
    }  
}
```

C. Looping

a. For Loops

Gunakan for loop saat kondisinya tau persis seberapa banyak looping akan dilakukan, contohnya melakukan perulangan sebanyak 10 kali dengan iterasi sebanyak 1 tingkat atau 1 kali.

Contoh Kode

```
void main() {  
    // LOOPING  
    for (int i = 1; i <= 5; i++) {  
        print(i);  
    }  
}
```

b. While Loops

Gunakan while loop saat kondisinya tidak tahu kapan perulangan akan berhenti, contohnya sediakan input angka hingga user menginput tanda " ".

Contoh Kode

```
void main() {  
    // While loops  
    int i = 1; // Deklarasi variabel  
    while (i <= 5) {  
        print(i);  
        i++; // Tambahkan 1 ke i setelah setiap iterasi  
    }  
}
```

D. List

List adalah tipe data yang digunakan untuk merepresentasikan koleksi nilai yang terurut. List dapat berisi elemen-elemen dengan tipe data yang sama atau berbeda. List dapat diinisialisasi dengan nilai-nilai yang diberikan, atau dapat dibuat kosong dan kemudian diisi kemudian.

a. Fixed length list

```

void main() {
    // LIST
    // Membuat fixed-length list dengan panjang 3
    List<int> fixedList =
        List.filled(3, 0); // List dengan 3 elemen, diisi dengan 0

    // Mengubah elemen dalam list
    fixedList[0] = 10;
    fixedList[1] = 20;
    fixedList[2] = 30;

    print(fixedList); // Output: [10, 20, 30]
}

```

b. Growable List

Gunakan growable list apabila memiliki banyak object yang tidak menentu atau banyaknya object yang terus bertambah.

Contoh Kode

```

void main() {
    // Membuat growable list (panjangnya bisa berubah)
    List<int> growableList = [];

    // Menambahkan elemen ke dalam list
    growableList.add(10);
    growableList.add(20);
    growableList.add(30);

    print(growableList); // Output: [10, 20, 30]

    // Menambahkan lebih banyak elemen
    growableList.add(40);
    growableList.add(50);

    print(growableList); // Output: [10, 20, 30, 40, 50]

    // Menghapus elemen dari list
    growableList.remove(20);

    print(growableList); // Output: [10, 30, 40, 50]
}

```

E. Fungsi

Pada bahasa pemrograman yang mendukung Object Oriented Programming, fungsi atau prosedur memiliki peranan yang sangat penting. Untuk menghasilkan kualitas kode yang sangat baik, programmer bisa menggunakan beberapa prinsip pemrograman yang umum digunakan seperti SOLID, KISS, YAGNI, dsb. Semua prinsip tersebut menjunjung tinggi separation of concern yang artinya setiap kodingan memiliki tanggung jawabnya sendiri dan mengurangi sebanyak mungkin boilerplate code.

a. Mendefinisikan Fungsi

```
void function_name() {  
    // statements  
}
```

b. Memanggil Fungsi

```
void main() {  
    print(factorial(6));  
}
```

c. Mengembalikan Nilai

```
void main() {  
    //mengembalikan nilai  
    String sapaan(String nama) {  
        return "Halo, $nama!";  
    }  
}
```

d. Menambahkan Parameter

```
void main() {  
    //menambahkan parameter  
    void greet(String name, int age) {  
        print('Hello $name, you are $age years old.');    }  
}
```

UNGUIDED

1. Soal:

Buatlah sebuah fungsi dalam Dart yang menerima sebuah nilai dari user, lalu melakukan percabangan untuk memberikan output berdasarkan kondisi berikut:

Deskripsi :

- Jika nilai lebih besar dari 70, program harus mereturn "Nilai A".
- Jika nilai lebih besar dari 40 tetapi kurang atau sama dengan 70, program harus mereturn "Nilai B".
- Jika nilai lebih besar dari 0 tetapi kurang atau sama dengan 40, program harus mereturn "Nilai C".
- Jika nilai tidak memenuhi semua kondisi di atas, program harus mereturn teks kosong.

Sampel Input: 80 Sampel Output: 80 merupakan Nilai A Sampel Input: 5
Sampel Output: 50 merupakan Nilai B

Jawaban:

Source code

```
1  import 'dart:io';
2
3  void main() {
4    print('Masukkan nilai:');
5    String? input = stdin.readLineSync();
6
7    int nilai = int.tryParse(input ?? '') ?? -1;
8
9    String result = cekNilai(nilai);
10
11    if (result.isNotEmpty) {
12      print('$nilai merupakan $result');
13    } else {
14      print('Nilai tidak valid');
15    }
16  }
17
18  String cekNilai(int nilai) {
19    if (nilai > 70) {
20      return 'Nilai A';
21    } else if (nilai > 40 && nilai <= 70) {
22      return 'Nilai B';
23    } else if (nilai > 0 && nilai <= 40) {
24      return 'Nilai C';
25    } else {
26      return '';
27    }
28  }
29
```

Output

```
PS D:\Backup Folder E\KULIAH\SEMESTER 5\PPB_Aorinka-Anendya-Chazanah_2211104013_06-01\03_Pengenalan_Dart\UNGUIDED>
Masukkan nilai:
80
80 merupakan Nilai A
PS D:\Backup Folder E\KULIAH\SEMESTER 5\PPB_Aorinka-Anendya-Chazanah_2211104013_06-01\03_Pengenalan_Dart\UNGUIDED>
Masukkan nilai:
60
60 merupakan Nilai B
PS D:\Backup Folder E\KULIAH\SEMESTER 5\PPB_Aorinka-Anendya-Chazanah_2211104013_06-01\03_Pengenalan_Dart\UNGUIDED>
```


Penjelasan:

Kode di atas menggunakan algoritma percabangan. Pertama, program akan meminta pengguna untuk memasukkan input nilai yang nantinya akan ditampilkan pada console.

Setelah itu, nilai yang diinputkan kemudian diubah menjadi integer. Apabila konversi gagal, maka akan diberikan nilai default -1.

Kemudian nilai yang telah diinputkan dan dikonversi tersebut akan dievaluasi menggunakan fungsi. Jika nilai lebih dari 70, hasilnya adalah 'Nilai A, jika antara 41 dan 70, hasilnya adalah 'Nilai B', dan jika antara 1 dan 40, hasilnya adalah 'Nilai C'. Terakhir, apabila nilai tidak valid, fungsi akan mengembalikan string kosong.

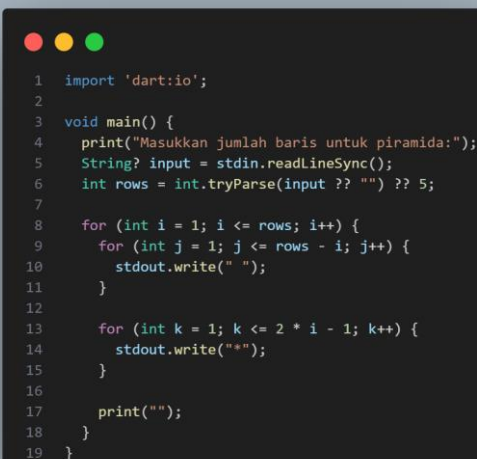
Output yang dihasilkan adalah sebuah string yang menunjukkan nilai yang dimasukkan dan kategori di atas, atau pesan yang menunjukkan bahwa nilai tersebut tidak valid.

2. Soal:

Buatlah sebuah program dalam Dart yang menampilkan piramida bintang dengan menggunakan for loop. Panjang piramida ditentukan oleh input dari user.

Jawaban:

Source code



```
1 import 'dart:io';
2
3 void main() {
4   print("Masukkan jumlah baris untuk piramida:");
5   String? input = stdin.readLineSync();
6   int rows = int.tryParse(input ?? "") ?? 5;
7
8   for (int i = 1; i <= rows; i++) {
9     for (int j = 1; j <= rows - i; j++) {
10      stdout.write(" ");
11    }
12
13    for (int k = 1; k <= 2 * i - 1; k++) {
14      stdout.write("*");
15    }
16
17    print("");
18  }
19 }
```

Output

```
PS D:\Backup Folder E\KULIAH\SEMESTER 5\PPB_Aorinka-  
Masukkan jumlah baris untuk piramida:  
4  
  *  
 ***  
*****  
*****
```

Penjelasan:

Program ini dibuat untuk menampilkan piramida bintang yang ukurannya ditentukan oleh input pengguna. Program ini menggunakan looping for.

Cara kerja program ini dimulai dengan fungsi `stdin.readLineSync()`. Fungsi ini digunakan meminta pengguna untuk memasukkan jumlah baris yang diinginkan. Input tersebut nantinya diubah menjadi tipe data integer menggunakan `int.TryParse()`. Jika input kosong atau tidak valid, maka nilai default yang digunakan adalah 5.

Setelah itu, program kemudian menggunakan dua perulangan for untuk mencetak piramida. Perulangan pertama digunakan untuk mencetak spasi agar bintang tersusun rapi membentuk piramida. Perulangan kedua mencetak bintang (*) yang membentuk baris piramida, di mana jumlah bintang di setiap baris bertambah secara bertahap.

Setelah setiap baris selesai dicetak, program berpindah ke baris baru menggunakan `print("")`. Output program ini nantinya akan berbentuk piramida yang terdiri dari bintang dengan jumlah baris sesuai input yang diberikan.

3. Soal

Buatlah program Dart yang meminta input berupa sebuah bilangan bulat dari user, kemudian program akan mengecek apakah bilangan tersebut merupakan bilangan prima atau bukan.

Sampel Input: 23

Sampel Output: bilangan prima

Sampel Input: 12 Sampel Output: bukan bilangan prima

Jawaban:
Source code

```
1 import 'dart:io';
2
3 void main() {
4   print('Masukkan sebuah bilangan bulat:');
5   String? input = stdin.readLineSync();
6
7   int angka = int.tryParse(input ?? '') ?? -1;
8
9   if (isPrime(angka)) {
10    print('$angka adalah bilangan prima.');
11  } else {
12    print('$angka bukan bilangan prima.');
13  }
14 }
15
16 bool isPrime(int angka) {
17   if (angka <= 1) {
18     return false;
19   }
20   for (int i = 2; i <= angka ~/ 2; i++) {
21     if (angka % i == 0) {
22       return false;
23     }
24   }
25   return true;
26 }
27
```

Output

```
PS D:\Backup Folder E\KULIAH\SEMESTER 5\PPB_Aorinka-Anendya-Chazanah_2211104013_06-01\03_Pengenalan_Dart\UNDGUIDED\soal_3> dart
soal3.dart
Masukkan sebuah bilangan bulat:
23
23 adalah bilangan prima.
PS D:\Backup Folder E\KULIAH\SEMESTER 5\PPB_Aorinka-Anendya-Chazanah_2211104013_06-01\03_Pengenalan_Dart\UNDGUIDED\soal_3> dart
soal3.dart
Masukkan sebuah bilangan bulat:
12
12 bukan bilangan prima.
PS D:\Backup Folder E\KULIAH\SEMESTER 5\PPB_Aorinka-Anendya-Chazanah_2211104013_06-01\03_Pengenalan_Dart\UNDGUIDED\soal_3> |
```

Penjelasan

Program ini dibuat untuk menerima input berupa bilangan bulat dari pengguna, lalu mengecek apakah bilangan tersebut adalah bilangan prima atau bukan.

Pertama, pengguna akan diminta untuk memasukkan sebuah bilangan bulat, setelah memasukkan angka, program akan memeriksa apakah bilangan tersebut prima atau tidak.

Program tersebut menggunakan loop untuk mengecek apakah ada pembagi selain 1 dan bilangan itu sendiri. Karena, sebuah bilangan dikatakan prima jika hanya bisa dibagi oleh 1 dan angka itu sendiri, sehingga program menggunakan loop. Jika ditemukan pembagi, maka bilangan tersebut bukan prima, dan program mencetak bahwa bilangan tersebut bukan bilangan prima. Jika tidak ditemukan pembagi lain, bilangan tersebut adalah bilangan

prima. Output yang dihasilkan program adalah pesan yang menyatakan apakah bilangan yang dimasukkan merupakan bilangan prima atau bukan.