

**LAPORAN PRAKTIKUM
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL 3
PENGENALAN DART**



**Disusun Oleh :
Arzario Irsyad Al Fatih
2211104032**

**Asisten Praktikum :
Ayu Susilowati
Noviana Rizki Anisa Putri**

**Dosen Pengampu :
Yudha Islami Sulistya**

**PROGRAM STUDI S1 SOFTWARE ENGINEERING
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024**

A. GUIDED

- Deklarasi VariabelCode

```
GUIDED > variable.dart > ...
Run | Debug
1 void main() {
2   // Penamaan variable menggunakan camelCase
3   String firstName = 'Praktikum';
4   var lastName = 'PPB';
5   final cuaca = 'cerah';
6
7   print('Hari ini $firstName $lastName');
8 }
```

PROBLEMS 1 OUTPUT TERMINAL PORTS DEBUG CONSOLE

as Kuliah\Semester 5\Praktikum\PPB Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenal
PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenalan_Dart\GUIDED\variable.dart"
Hari ini Praktikum PPB

Deskripsi Program

1. var adalah cara untuk mendeklarasikan variabel tanpa menentukan tipe datanya secara eksplisit. Dart akan secara otomatis menentukan tipe data berdasarkan nilai yang diberikan. Dalam contoh diatas adalah
2. Type annotation adalah cara untuk mendeklarasikan variabel dengan menentukan tipe datanya secara eksplisit. Pada contoh diatas yaitu String adalah tipe data yang secara eksplisit menyatakan bahwa variabel akan menyimpan teks.

Tipe	Deskripsi	Contoh
int	Integer (bilangan bulat)	1, -2, 0
double	Bilangan desimal	3.14, 10.53, -23.23
num	Bilangan bulat dan bilangan desimal	7, 3.14, -12.00
bool	Boolean	True, False
String	Text yang terdiri dari 0 atau beberapa karakter	'Faza', 'Hasna'
List	Sekuumpulan nilai	[0,1,2,3,4], ['a', 'b', 'c']
Map	Pasangan key-value	{"a": "aku", "b": 12}
dynamic	Tipe apapun	

3. final: Menyatakan bahwa nilai variabel tidak dapat diubah setelah inisialisasi.

- **Statement ControlCode**

If else

```

9      var open = 8;
10     var close = 17;
11     var now = 10;
12
13     if (now > open && now < close) {
14         print("Toko Buka");
15     } else {
16         print("Toko Tutup");
17     }
18 }

```

PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

as Kuliah\Semester 5\Praktikum\PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenal
 PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenalan_Dart\GUIDED\variable.dart"
 Toko Buka

Switchcase

```

GUIDED > variable.dart > ...
1      void main() {
20         // Switch Case
21         var day = 'b';
22
23         switch (day) {
24             case 'a':
25                 print('Nilai Sangat Bagus');
26                 break;
27             case 'b':
28                 print('Nilai Bagus');
29                 break; // Tambahkan break agar sesuai dengan pola
30             case 'c':
31                 print('Nilai Cukup');
32                 break;
33             default:
34                 print('Nilai Tidak Dikenali');
35         }
36     }

```

PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

as Kuliah\Semester 5\Praktikum\PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenal
 PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenalan_Dart\GUIDED\variable.dart"
 Nilai Bagus

Ternary

```

GUIDED > variable.dart > ...
1      void main() {
9         var open = 8;
10        var close = 17;
11        var now = 10;
12
13        // Ternary Operator
14        var tokoStatus = now > open && now < close ? "Toko Buka" : "Toko Tutup";
15        print(tokoStatus);
16    }

```

```
PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE
as Kuliah\Semester 5\Praktikum\PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengena
PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenalan_Dart\GUIDED\variable.dart"
Toko Buka
```

Deskripsi Program If else

Kode tersebut mengecek nilai variabel now yang menyatakan waktu saat ini.

1. Pertama, kode mengecek apakah now (waktu saat ini) berada di antara nilai variabel open (jam buka) dan close (jam tutup) menggunakan kondisi if (now \geq open && now $<$ close).
2. Jika kondisi tersebut benar, maka akan mencetak "Toko Buka".
3. Jika tidak, kode akan mengecek apakah now sama dengan 12 menggunakan kondisi else if (now == 12).
4. Jika now sama dengan 12, maka akan mencetak "Toko sedang istirahat".
5. Jika tidak ada kondisi di atas yang terpenuhi, maka akan mencetak "TokoTutup".

Deskripsi Program Switch Case

Kode ini menggunakan switch case untuk memeriksa nilai variabel `nilai`. Jika `nilai` adalah "a", "b", atau "c", masing-masing akan mencetak "Nilai sangat bagus", "Nilai bagus", atau "Nilai cukup". Jika tidak ada case yang sesuai, default akan mencetak "Nilai tidak tersedia".

Deskripsi Ternary Operator

Kode menggunakan operator ternary untuk menetapkan apakah toko buka atau tutup, lalu mencetak hasilnya.

- **Looping Code**

For Loop

```
GUIDED > loop.dart > ...
Run | Debug
1 void main() {
2   // For loop untuk mencetak angka 1 sampai 5
3   for (int i = 1; i <= 5; i++) {
4     print(i);
5   }
}

PROBLEMS 1 OUTPUT TERMINAL PORTS DEBUG CONSOLE
as Kuliah\Semester 5\Praktikum\PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Per
PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenalan_Dart\GUIDED\loop.dart"
1
2
3
4
5
```

While Loop

```
7 // While loop untuk mencetak angka 1 sampai 5
8 int i = 1;
9 while (i <= 5) {
10     print('Angka: $i');
11     i++; // Increment untuk menghindari loop tak berujung
12 }
13 }
14
```

PROBLEMS 1 OUTPUT **TERMINAL** PORTS DEBUG CONSOLE

as Kuliah\Semester 5\Praktikum\PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Peng
PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenalan_Dart\GUIDED\loop.dart"

Angka: 1
Angka: 2
Angka: 3
Angka: 4
Angka: 5

Deskripsi for loop

Cara kerja: Variabel `i` diinisialisasi dengan nilai 0. Kondisi `i <= 5` diperiksa, dan jika benar, nilai `i` dicetak lalu ditambahkan 1. Proses ini berulang hingga `i` lebih besar dari 5.

Output: Angka 0 hingga 5 dicetak satu per satu.

Deskripsi While loop

Cara kerja: Variabel `i` diinisialisasi dengan nilai 0. Kondisi `i <= 5` diperiksa, dan jika benar, program mencetak "print: \$i" dan menambah `i` sebesar 1. Proses ini terus berlanjut hingga `i` menjadi lebih besar dari 5.

Output: Mencetak "print: 0" hingga "print: 5".

- **List Code**

Fixed Length List

GUIDED > list.dart > main

Run | Debug

```
1 void main() {
2     List<int> fixesList = List.filled(3, 0);
3     fixesList[0] = 10;
4     fixesList[1] = 20;
5     fixesList[2] = 30;
6     print("Fixed Length List: $fixesList");
7 }
8
```

PROBLEMS 1 OUTPUT **TERMINAL** PORTS DEBUG CONSOLE

as Kuliah\Semester 5\Praktikum\PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03 Per
PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenalan_Dart\GUIDED\list.dart"

Fixed Length List: [10, 20, 30]

Growable List

```
GUIDED > list.dart > main
1  void main() {
9  List<int> growableList = []; // Growable List
10  growableList.add(10);
11  growableList.add(20);
12  growableList.add(30);
13  print("Growable List: $growableList");
14  growableList.add(50);
15  growableList.add(60);
16  growableList.add(70);
17  print(growableList);
18 }
```

PROBLEMS 40 OUTPUT TERMINAL PORTS DEBUG CONSOLE

```
as Kuliah\Semester 5\Praktikum\PPB_Arzarior_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pen
PPB_Arzarior_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenalan_Dart\GUIDED\list.dart"
Growable List: [10, 20, 30]
[10, 20, 30, 50, 60, 70]
```

Deskripsi List Fixed Length List

Inisialisasi: Membuat `fixedList` dengan panjang tetap 3 dan diisi dengan nilai 0.

Pengisian Nilai: Mengisi elemen-elemen list dengan nilai 10, 20, dan 30.

Output: Mencetak isi dari `fixedList`, yang sekarang menjadi [10, 20, 30].

Penghapusan Elemen: Mencoba menghapus elemen pada indeks 30.

Masalah: Karena `fixedList` hanya memiliki 3 elemen (indeks 0, 1, dan 2), menghapus elemen pada indeks 30 akan menyebabkan error (index out of range).

Fixed Length List: List dengan panjang tetap yang tidak bisa diubah setelah inisialisasi.

Error Handling: Kode ini akan menghasilkan error saat mencoba menghapus elemen di luar batas list. Sebaiknya menggunakan indeks yang valid untuk menghindari error.

Deskripsi Growable List

Inisialisasi: Membuat `growableList` yang dapat berkembang (growable) dengan awal kosong.

Menambahkan Elemen: Menambahkan elemen 10, 20, dan 30 ke dalam list.

Output: Mencetak isi dari `growableList`, yang sekarang adalah [10, 20, 30].

Menambahkan Elemen Tambahan: Menambahkan elemen 50, 60, dan 70 ke dalam list.

Output: Mencetak isi terbaru dari `growableList`, yang sekarang adalah [10, 20, 30, 50, 60, 70].

Growable List: List yang dapat bertambah panjang saat elemen baru ditambahkan.

Fleksibilitas: Memungkinkan untuk menambah elemen kapan saja tanpa batasan ukuran awal.

- **Fungsi Code**

```
GUIDED > fungsi.dart > ...
1 // Fungsi tanpa nilai kembalian
2 void cetakPesan(String pesan) {
3   | print(pesan); // Mencetak pesan
4 }
5
6 // Fungsi dengan nilai kembalian
7 int kaliAngka(int a, int b) {
8   | return a * b; // Mengembalikan hasil perkalian
9 }
10
11 Run | Debug
12 void main() {
13   | // Memanggil fungsi dan menyimpan hasilnya
14   | int hasil = kaliAngka(4, 5);
15
16   | // Menampilkan hasil
17   | print('Hasil kali: $hasil');
18
19   | // Memanggil fungsi tanpa nilai kembalian
20   | cetakPesan('Halo, Dart!');
21
22   | // Fungsi greet
23   | greet('Alice', 25);
24 }
25
26 void greet(String name, int age) {
27   | print('Hello $name, you are $age years old.');
```

PROBLEMS 40 OUTPUT TERMINAL PORTS DEBUG CONSOLE

```
as Kuliah\Semester 5\Praktikum\PPB_Arzarior_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Penge
PPB_Arzarior_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenalan_Dart\GUIDED\fungsi.dart"
Hasil kali: 20
Halo, Dart!
Hello Alice, you are 25 years old.
```

Deskripsi

0. Fungsi cetakPesan: Menerima parameter bertipe String dan mencetak pesan yang diterima.
1. Fungsi perkalian: Menerima dua parameter bertipe int, mengalikan keduanya, dan mengembalikan hasilnya.
2. Fungsi greet: Menerima dua parameter, name bertipe String dan age bertipe int, lalu mencetak sapaan yang menyertakan nama dan usia.
3. Fungsi mainfunction: Memanggil fungsi-fungsi yang telah didefinisikan sebelumnya. Menghitung hasil perkalian 5 dan 4 dengan fungsi perkalian. Mencetak pesan menggunakan fungsi cetakPesan. Menggreet pengguna dengan nama 'Alice' dan usia 25 menggunakan fungsi greet.

B. UNGUIDED

1. Soal 1

```
UNGUIDED > soal-01.dart > ...
1  import 'dart:io';
2
3  String cekNilai(int nilai) {
4      if (nilai > 70) {
5          return "Nilai A";
6      } else if (nilai > 40 && nilai <= 70) {
7          return "Nilai B";
8      } else if (nilai > 0 && nilai <= 40) {
9          return "Nilai C";
10     } else {
11         return "";
12     }
13 }
14
15 void main() {
16     stdout.write('Masukkan nilai: ');
17     int nilai = int.parse(stdin.readLineSync());
18     String hasil = cekNilai(nilai);
19     if (hasil.isNotEmpty) {
20         print('$nilai merupakan $hasil');
21     } else {
22         print('Nilai tidak valid');
23     }
24 }
25
```

Run | Debug

PROBLEMS 48 OUTPUT TERMINAL PORTS DEBUG CONSOLE

```
as Kuliah\Semester 5\Praktikum\PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenal
PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenal Dart\UNGUIDED\soal-01.dart"
Masukkan nilai: 75
75 merupakan Nilai A
PS C:\Users\toshiba\Documents\Tugas Kuliah\Semester 5\Praktikum\PPB_Arzario_Irsyad_Al_Fati
ugas Kuliah\Semester 5\Praktikum\PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengen
Masukkan nilai: 35
35 merupakan Nilai C
```

Penjelasan

Mengimpor Library: Mengimpor library `dart:io` untuk memungkinkan penggunaan fungsi input dan output.

Meminta Input: Meminta pengguna untuk memasukkan tinggi piramida.

Membaca Input: Menggunakan `stdin.readLineSync()` untuk membaca input, kemudian mengonversinya menjadi integer dengan `int.tryParse()`.

Memeriksa Validitas Input: Memeriksa apakah input valid. Jika nilai tinggi `null` atau kurang dari atau sama dengan 0, program mencetak pesan kesalahan dan berhenti.

Mencetak Piramida: Menggunakan nested loops untuk mencetak piramida:

- Loop Pertama: `for (int i = 1; i <= tinggi; i++)` mengatur tingkat piramida.
- Loop Kedua: `for (int j = 1; j <= tinggi - i; j++)` mencetak spasi untuk perataan.
- Loop Ketiga: `for (int k = 1; k <= (2 * i - 1); k++)` mencetak bintang berdasarkan tingkat.

2. Soal

```
UNGUIDED > soal-02.dart > ...
1  import 'dart:io';
2
3  void cetakPiramida(int tinggi) {
4      for (int i = 1; i <= tinggi; i++) {
5          // Menampilkan spasi
6          for (int j = 1; j <= tinggi - i; j++) {
7              stdout.write(' ');
8          }
9          // Menampilkan bintang
10         for (int k = 1; k <= 2 * i - 1; k++) {
11             stdout.write('*');
12         }
13         print('\n');
14     }
15 }
16
17 Run | Debug
18 void main() {
19     stdout.write('Masukkan tinggi piramida: ');
20     int tinggi = int.parse(stdin.readLineSync()!);
21     cetakPiramida(tinggi);
22 }
```

PROBLEMS 48 OUTPUT TERMINAL PORTS DEBUG CONSOLE

```
as Kuliah\Semester 5\Praktikum\PPB_Arzario_Irsyad_Al_Fatih_22111040
PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenalan Dart
Masukkan tinggi piramida: 5
*
***
*****
*****
*****
```

Deskripsi

Mengimpor library `dart:io` untuk operasi input dan output. Meminta pengguna memasukkan tinggi piramida dan mengonversi input menjadi integer. Memeriksa validitas input; jika tidak valid, mencetak pesan kesalahan dan berhenti. Menggunakan nested loops untuk mencetak piramida: loop pertama mengontrol tingkat, loop kedua mencetak spasi, dan loop ketiga mencetak bintang sesuai tingkat.

3. Soal

```
UNGUIDED > soal-03.dart > ...
1  import 'dart:io';
2
3  bool cekPrima(int bilangan) {
4      if (bilangan < 2) {
5          return false;
6      }
7      for (int i = 2; i <= bilangan ~/ 2; i++) {
8          if (bilangan % i == 0) {
9              return false;
10         }
11     }
12     return true;
13 }
14
15 void main() {
16     stdout.write('Masukkan bilangan: ');
17     int bilangan = int.parse(stdin.readLineSync());
18     if (cekPrima(bilangan)) {
19         print('$bilangan adalah bilangan prima');
20     } else {
21         print('$bilangan bukan bilangan prima');
22     }
23 }
24
```

Run | Debug

PROBLEMS 48 OUTPUT TERMINAL PORTS DEBUG CONSOLE

```
as Kuliah\Semester 5\Praktikum\PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE
PPB_Arzario_Irsyad_Al_Fatih_2211104032_SE_06_02\03_Pengenalan_Dart\UNGUI
Masukkan bilangan: 23
23 adalah bilangan prima
PS C:\Users\toshiba\Documents\Tugas Kuliah\Semester 5\Praktikum\PPB_Arza
ugas Kuliah\Semester 5\Praktikum\PPB_Arzario_Irsyad_Al_Fatih_2211104032_
Masukkan bilangan: 12
12 bukan bilangan prima
```

Deskripsi

Import Library: Mengimpor library `dart:io` untuk operasi input dan output.

Fungsi `cekPrima`: Memeriksa apakah `bilangan` yang dimasukkan adalah bilangan prima. Jika bilangan kurang dari 2, fungsi mengembalikan `false`. Fungsi menggunakan loop untuk memeriksa pembagi dari 2 hingga setengah dari bilangan tersebut. Jika ada pembagi yang cocok, fungsi mengembalikan `false`; jika tidak, mengembalikan `true`.

Fungsi `main`: Meminta pengguna untuk memasukkan bilangan, membaca input, dan memanggil fungsi `cekPrima`. Program mencetak hasil apakah bilangan tersebut adalah bilangan prima atau bukan.