

**LAPORAN TUGAS GUIDED
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL 3
PENGENALAN DART**



Disusun Oleh :

Ade Fatkhul Anam / 22111040451

SE-06-02

Asisten Praktikum :

Muhammad Faza Zulian Gesit Al Barru

Aisyah Hasna Aulia

Dosen Pengampu :

Yudha Islami Sulistya

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

GUIDED

A. Variable

Variabel dalam Dart adalah lokasi penyimpanan yang diberi nama, digunakan untuk menyimpan data yang dapat diubah selama eksekusi program. Variabel ini bisa menyimpan berbagai jenis data seperti angka, teks, objek, dan lainnya

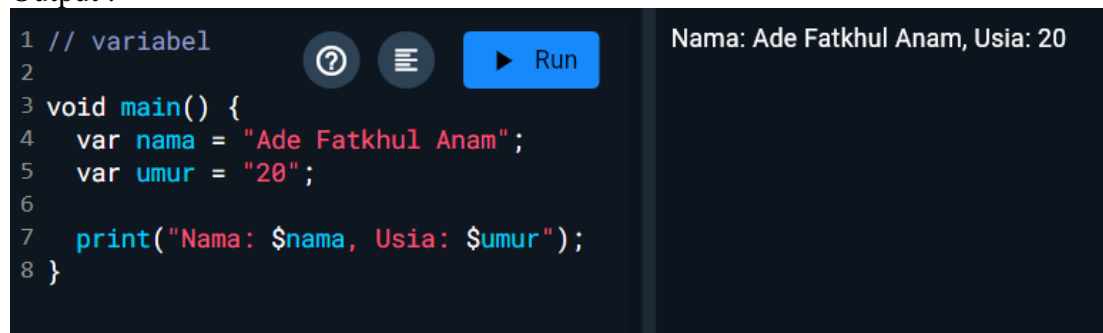
Source code:

```
// variabel

void main() {
  var nama = "Ade Fatkhul Anam";
  var umur = "20";

  print("Nama: $nama, Usia: $umur");
}
```

Output :

A screenshot of a code editor with a dark theme. On the left, the Dart source code is displayed:

```
1 // variabel
2
3 void main() {
4   var nama = "Ade Fatkhul Anam";
5   var umur = "20";
6
7   print("Nama: $nama, Usia: $umur");
8 }
```

 Above the code are icons for help, settings, and a blue 'Run' button. On the right, the output is shown: 'Nama: Ade Fatkhul Anam, Usia: 20'.

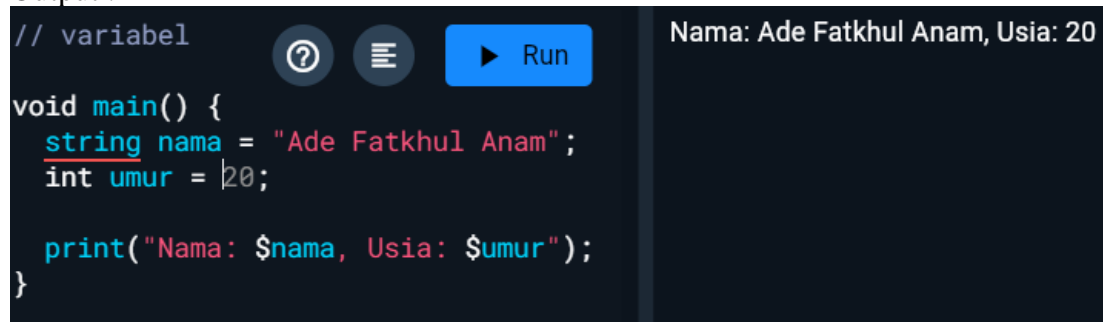
Source code:

```
// variabel dengan string dan integer

void main() {
  string nama = "Ade Fatkhul Anam";
  int umur = 20;

  print("Nama: $nama, Usia: $umur");
}
```

Output :

A screenshot of a code editor with a dark theme. On the left, the Dart source code is displayed:

```
// variabel

void main() {
  string nama = "Ade Fatkhul Anam";
  int umur = 20;

  print("Nama: $nama, Usia: $umur");
}
```

 Above the code are icons for help, settings, and a blue 'Run' button. On the right, the output is shown: 'Nama: Ade Fatkhul Anam, Usia: 20'.

Source code:

```
// Multiple variable

void main() {
  String firstName, lastName;
  firstName = "Ade";
  lastName = "Fatkhul Anam";
  print("Nama Lengkap: $firstName $lastName");
}
```

Output :

A screenshot of a Dart IDE interface. On the left, a code editor shows the following Dart code:

```
1 // Multiple variable
2
3 void main() {
4   String firstName, lastName;
5   firstName = "Ade";
6   lastName = "Fatkhul Anam";
7   print("Nama Lengkap: $firstName $lastName");
8 }
9
```

There are icons for help, settings, and a 'Run' button. On the right, the output console displays the text: 'Nama Lengkap: Ade Fatkhul Anam'.

B. Statement Control

Statement Control dalam Dart adalah pernyataan yang digunakan untuk mengontrol alur eksekusi program.

Source code:

```
// Statement Control
// IF-ELSE Statement

void main() {
  var openHours = 8;
  var closedHours = 21;
  var now = 17;
  if (now > openHours && now < closedHours) {
    print("Hello, kami telah buka");
  } else {
    print("Sorry, kami telah tutup");
  }
}
```

Output:

```
1 // Statement Control
2 // IF-ELSE Statement
3
4 void main() {
5     var openHours = 8;
6     var closedHours = 21;
7     var now = 17;
8     if (now > openHours && now < closedHours) {
9         print("Hello, kami telah buka");
10    } else {
11        print("Sorry, kami telah tutup");
12    }
13 }
14
```

Run

Hello, kami telah buka

Source code:

```
// Statement Control
// Switch-Case Statement

void main() {
var nilai = 'b';

    switch (nilai){
        case 'a':
            print('Nilai Sangat Bagus');
            break;
        case 'b':
            print("Nilai Bagus");
        case 'c':
            print('Nilai Cukup');
            break;
        default:
            print("Nilai tidak tersedia");
    }
}
```

Output :



```
1 // Statement Control
2 // Switch-Case Statement
3
4 void main() {
5     var nilai = 'b';
6
7     switch (nilai){
8         case 'a':
9             print('Nilai Sangat Bagus');
10            break;
11        case 'b':
12            print("Nilai Bagus");
13        case 'c':
14            print('Nilai Cukup');
15            break;
16        default:
17            print("Nilai tidak tersedia");
18    }
19
20 }
```

Nilai Bagus

C. Looping

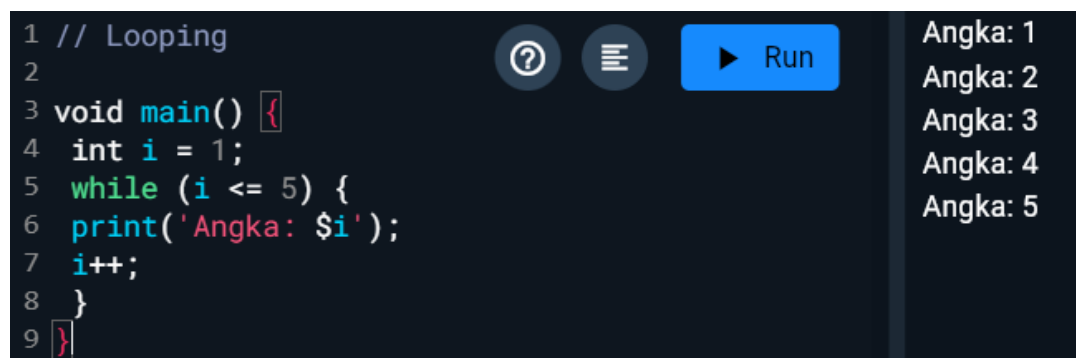
Looping dalam Dart adalah proses mengulang eksekusi blok kode tertentu beberapa kali hingga kondisi tertentu terpenuhi.

Source code:

```
// Looping

void main() {
  int i = 1;
  while (i <= 5) {
    print('Angka: $i');
    i++;
  }
}
```

Output:

A screenshot of a code editor with a dark theme. On the left, a Dart code snippet is shown with line numbers 1 through 9. The code uses a while loop to print the numbers 1 through 5. On the right side of the editor, the output of the code is displayed, showing the text 'Angka: 1' through 'Angka: 5' on separate lines. Above the output, there are three circular icons: a question mark, a hamburger menu, and a 'Run' button with a play icon.

```
1 // Looping
2
3 void main() {
4   int i = 1;
5   while (i <= 5) {
6     print('Angka: $i');
7     i++;
8   }
9 }
```

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

D. List

List adalah kumpulan objek yang diatur secara teratur. List ini mirip dengan array di bahasa pemrograman lain. List di Dart dapat menyimpan berbagai tipe data dan diakses menggunakan indeks yang dimulai dari 0. Ada dua jenis List di Dart:

1. **Growable List:** Panjangnya bisa berubah-ubah sesuai kebutuhan.
2. **Fixed Length List:** Panjangnya tetap dan tidak bisa diubah setelah dibuat.

Source code:

```
// list
// Growable List

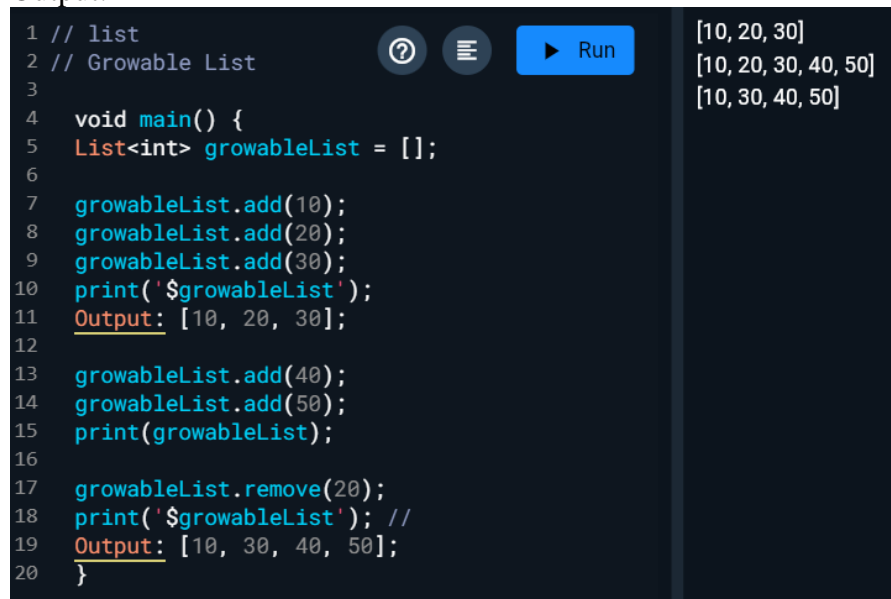
void main() {
  List<int> growableList = [];

  growableList.add(10);
  growableList.add(20);
  growableList.add(30);
  print('$growableList');
  Output: [10, 20, 30];

  growableList.add(40);
  growableList.add(50);
  print(growableList);

  growableList.remove(20);
  print('$growableList'); //
  Output: [10, 30, 40, 50];
}
```

Output:



The screenshot shows a code editor with the following Dart code:

```
1 // list
2 // Growable List
3
4 void main() {
5   List<int> growableList = [];
6
7   growableList.add(10);
8   growableList.add(20);
9   growableList.add(30);
10  print('$growableList');
11  Output: [10, 20, 30];
12
13  growableList.add(40);
14  growableList.add(50);
15  print(growableList);
16
17  growableList.remove(20);
18  print('$growableList'); //
19  Output: [10, 30, 40, 50];
20 }
```

On the right side, the output of the program is displayed in three lines:

```
[10, 20, 30]
[10, 20, 30, 40, 50]
[10, 30, 40, 50]
```

Source code:

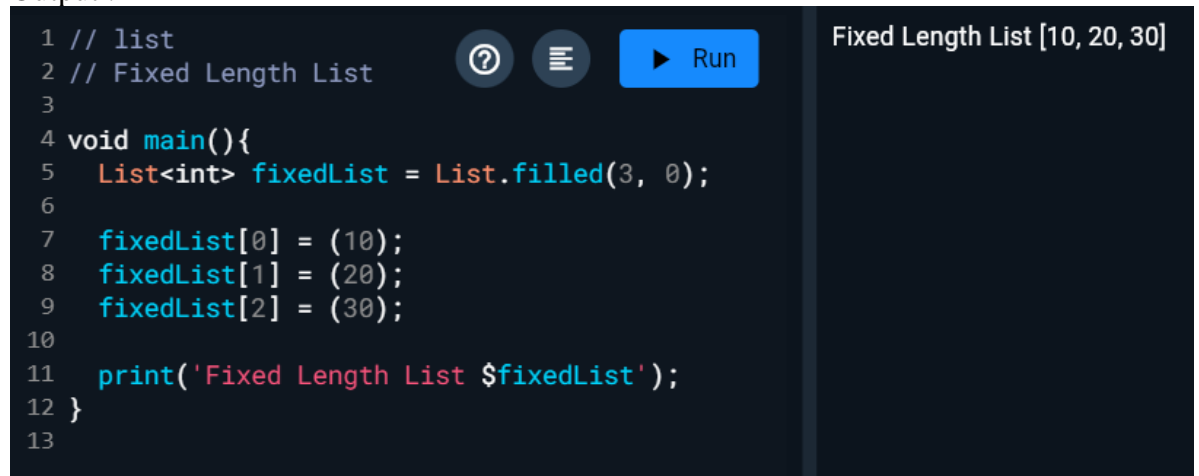
```
// list
// Fixed Length List

void main(){
    List<int> fixedList = List.filled(3, 0);

    fixedList[0] = (10);
    fixedList[1] = (20);
    fixedList[2] = (30);

    print('Fixed Length List $fixedList');
}
```

Output :

A screenshot of a Dart IDE interface. On the left, a code editor shows the source code for a Dart program. The code defines a main function that creates a List of size 3 filled with 0, then sets the elements to 10, 20, and 30, and finally prints the list. On the right, there is a 'Run' button and a panel displaying the output of the program, which is 'Fixed Length List [10, 20, 30]'.

```
1 // list
2 // Fixed Length List
3
4 void main(){
5     List<int> fixedList = List.filled(3, 0);
6
7     fixedList[0] = (10);
8     fixedList[1] = (20);
9     fixedList[2] = (30);
10
11     print('Fixed Length List $fixedList');
12 }
13
```

Fixed Length List [10, 20, 30]

E. Fungsi

Fungsi dalam Dart adalah blok kode yang dapat dipanggil dan dijalankan dari berbagai bagian program. Fungsi ini dapat mengambil parameter, mengembalikan nilai, atau melakukan tindakan tertentu tanpa mengembalikan apa pun. Source code:

```
// Fungsi

void cetakPesan(String pesan) {
    print(pesan);
}

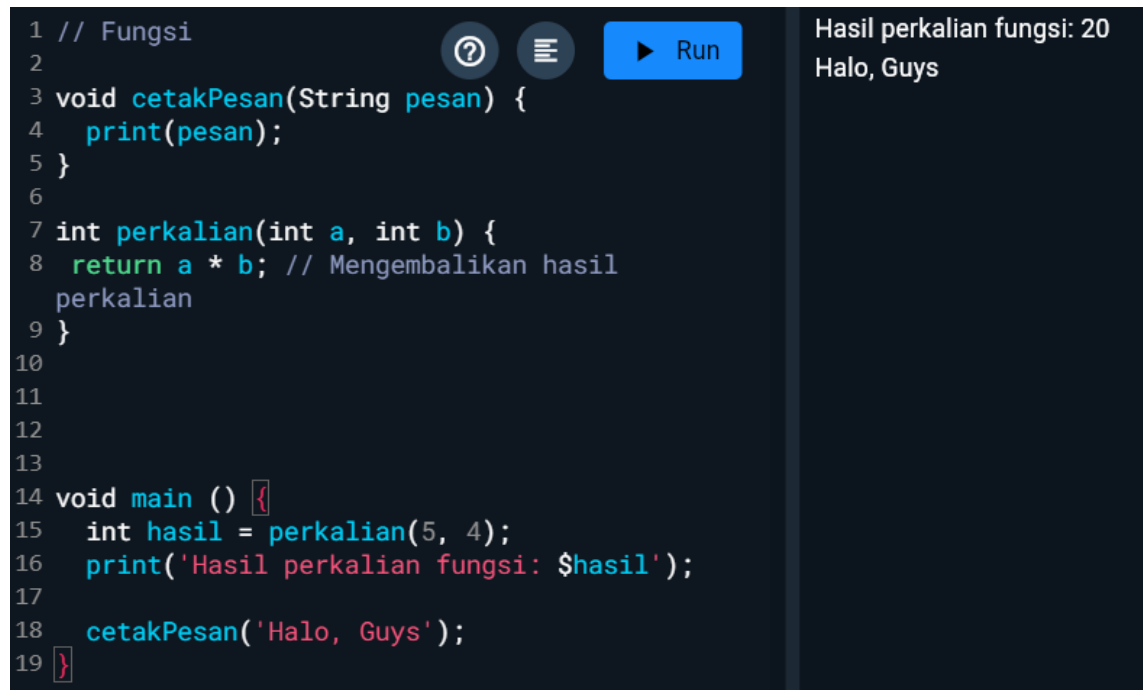
int perkalian(int a, int b) {
    return a * b; // Mengembalikan hasil perkalian
}
```

```
void main () {
    int hasil = perkalian(5, 4);
}
```



```
    print('Hasil perkalian fungsi: $hasil');  
    cetakPesan('Halo, Guys');  
}
```

Output :



```
1 // Fungsi  
2  
3 void cetakPesan(String pesan) {  
4     print(pesan);  
5 }  
6  
7 int perkalian(int a, int b) {  
8     return a * b; // Mengembalikan hasil  
   perkalian  
9 }  
10  
11  
12  
13  
14 void main () {  
15     int hasil = perkalian(5, 4);  
16     print('Hasil perkalian fungsi: $hasil');  
17  
18     cetakPesan('Halo, Guys');  
19 }
```

Hasil perkalian fungsi: 20
Halo, Guys

