

## Dasar Dart Lanjutan



### Praktikum 3

Tujuan :

- Mahasiswa memahami dasar-dasar pemrograman Dart lanjutan.
- Mahasiswa mampu menggunakan operator dan struktur kontrol Dart.

Output :

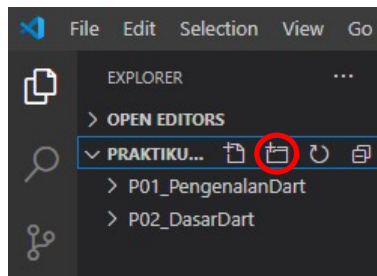
- Aplikasi Dart berbasis Console.

Alat dan Bahan:

- Laptop/PC Core i3 atau lebih tinggi
- Windows 7 (64bit OS) atau lebih tinggi
- Flutter SDK
- Visual Studio Code (VSCode)
- Flutter dan Dart Plugin sudah terpasang di VSCode
- Gunakan DartPad (<https://dartpad.dev/>) jika belum menginstal Flutter SDK

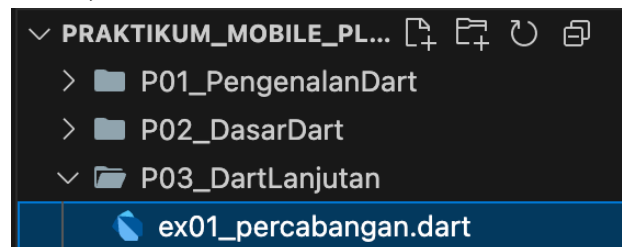
Berikut adalah langkah-langkah pengerjaan praktikum

1. Bukalah VSCode.
2. Buatlah sebuah folder baru dengan nama P03\_DartLanjutan . Klik icon **New Folder** yang diberi lingkaran merah.



Masukkan nama folder dengan nama “**P03\_DartLanjutan**”, tekan enter untuk membuat folder.

3. Buatlah sebuah file baru dengan nama `ex01_percabangan.dart`, simpan dalam folder `P03_DartLanjutan`. Klik icon **New File** yang diberi lingkaran merah. Pastikan folder yang aktif adalah folder `P03_DartLanjutan`, Masukkan nama file dengan nama “**ex01\_percabangan.dart**”, tekan enter untuk membuat file.



4. Masukkan kode berikut pada `ex01_percabangan.dart`:

```
import 'dart:io';

void main() {
  stdout.write('Masukkan angka: ');
  int? angka = int.tryParse(stdin.readLineSync()!);

  print('Nilai yang anda inputkan adalah ${angka ?? '-'}');
  angka = angka ?? 0;

  print(angka >= 75 ? "Lulus" : (angka >= 60 ? "Remedial" : "Tidak Lulus"));

  String status;
  if (angka >= 75) {
    status = "Lulus";
  } else if (angka >= 60) {
    status = "Remedial";
  } else {
    status = "Tidak Lulus";
  }
  print('Anda ${status}');

  status = '';
}
```

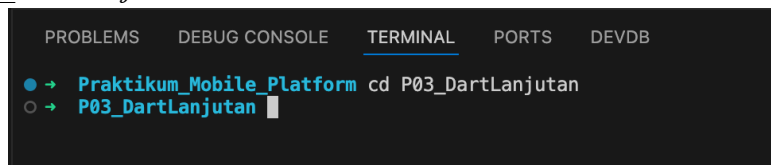
```

switch (angka ~/ 10) {
  case 10:
  case 9:
  case 8:
  case 7:
    status = "Lulus";
    break;
  case 6:
    status = "Remedial";
    break;
  default:
    status = "Tidak Lulus";
}

print("Nilai: $angka, Status: ${status}");
}

```

Simpan kode program. Bukalah terminal dengan cara klik menu Terminal → New Terminal pada VSCode, pindahlah kedalam folder P03\_DartLanjutan dengan memasukkan perintah “`cd P03_DartLanjutan`” di terminal.

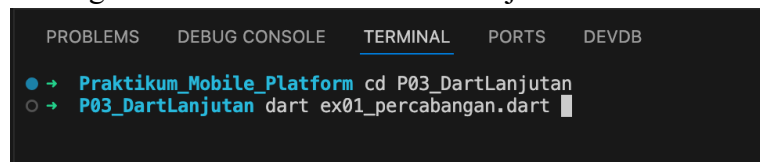


```

PROBLEMS  DEBUG CONSOLE  TERMINAL  PORTS  DEVDB
● → Praktik_Mobile_Platform cd P03_DartLanjutan
○ → P03_DartLanjutan

```

Masukkan perintah untuk menjalankan program di terminal ketik perintah “`dart ex01_percabangan.dart`” tekan **enter** untuk menjalankan:

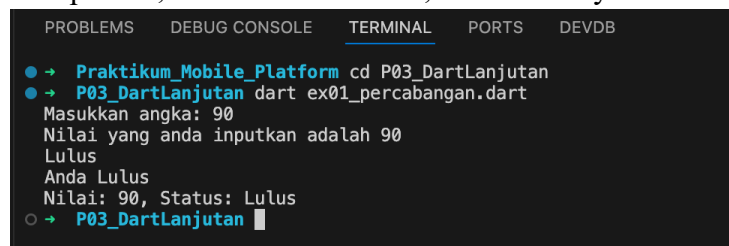


```

PROBLEMS  DEBUG CONSOLE  TERMINAL  PORTS  DEVDB
● → Praktik_Mobile_Platform cd P03_DartLanjutan
○ → P03_DartLanjutan dart ex01_percabangan.dart

```

Saat muncul inputkan, **masukkan nilai 90**, maka hasilnya adalah seperti berikut:



```

PROBLEMS  DEBUG CONSOLE  TERMINAL  PORTS  DEVDB
● → Praktik_Mobile_Platform cd P03_DartLanjutan
● → P03_DartLanjutan dart ex01_percabangan.dart
Masukkan angka: 90
Nilai yang anda inputkan adalah 90
Lulus
Anda Lulus
Nilai: 90, Status: Lulus
○ → P03_DartLanjutan

```

5. Buatlah sebuah file baru dengan nama `ex2_perulangan_list.dart`, simpan dalam folder `P03_DartLanjutan` dan masukkan kode berikut:

```

void main() {
  List<String> nama = ["Ali", "Budi", "Citra", "Dewi"];

  for (int i = 0; i < nama.length; i++) {
    if (nama[i] == "Budi") continue;
    print(nama[i]);
  }
}

```

```

    }
    print('\n');

    nama.forEach((nm) {
      print(nm);
    });
    print('\n');

    int i = 0;
    while (i < 2) {
      print(nama[i]);
      i++;
    }
    print('\n');

    i = 0;
    do {
      print(nama[i]);
      i++;
    } while (i <= 2);
    print('\n');
  }
}

```

6. Simpan kode program. Masukkan perintah untuk menjalankan program di terminal ketik perintah “dart ex2\_perulangan\_list.dart” tekan **enter** untuk menjalankan, dan hasilnya adalah seperti berikut:

```

P03_DartLanjutan dart ex02_perulangan_list.dart
Ali
Budi
Citra
Dewi

Ali
Budi
Citra
Dewi

Ali
Budi

Ali
Budi
Citra

```

7. Buatlah sebuah file baru dengan nama ex3\_perulangan\_map.dart, simpan dalam folder P03\_DartLanjutan dan masukkan kode berikut:

```

void main() {
  Map<String, int> umur = {'Andi': 25, 'Budi': 30, 'Citra': 28};

  umur.forEach((key, value) {
    print('$key berumur $value tahun');
  });
  print('\n');
}

```

```

for (var entry in umur.entries) {
  print('${entry.key} berumur ${entry.value} tahun');
}
print('\n');

for (var key in umur.keys) {
  print('$key berumur ${umur[key]} tahun');
}
}

```

8. Simpan kode program. Masukkan perintah untuk menjalankan program di terminal ketika perintah “dart ex3\_perulangan\_map.dart” tekan **enter** untuk menjalankan, dan hasilnya adalah seperti berikut:

```

PROBLEMS  DEBUG CONSOLE  TERMINAL  PORTS  DEVDB
● → P03_DartLanjutan dart ex03_perulangan_map.dart
Andi berumur 25 tahun
Budi berumur 30 tahun
Citra berumur 28 tahun

Andi berumur 25 tahun
Budi berumur 30 tahun
Citra berumur 28 tahun

Andi berumur 25 tahun
Budi berumur 30 tahun
Citra berumur 28 tahun
○ → P03_DartLanjutan █

```

9. Buatlah sebuah file baru dengan nama ex4\_input\_data.dart, simpan dalam folder P03\_DartLanjutan dan masukkan kode berikut:

```

import 'dart:io';

void main() {
  int? umur;
  while (umur == null || umur <= 0) {
    stdout.write('Masukkan umur Anda: ');
    String? inputUmur = stdin.readLineSync();
    umur = int.tryParse(inputUmur ?? '');
    if (umur == null || umur <= 0) {
      print('Input tidak valid, coba lagi.');
```

10. Simpan kode program. Masukkan perintah untuk menjalankan program di terminal ketika perintah “dart ex4\_input\_data.dart” tekan **enter** untuk menjalankan, dan hasilnya adalah seperti berikut:

```

PROBLEMS  DEBUG CONSOLE  TERMINAL  PORTS  DEVDB
● → P03_DartLanjutan dart ex04_input_data.dart
Masukkan umur Anda: 20
Umur Anda: 20 tahun

```

11. Buatlah sebuah file baru dengan nama `ex5_input_data_array.dart`, simpan dalam folder `P03_DartLanjutan` dan masukkan kode berikut:

```
import 'dart:io';

void main() {
  // Meminta input dari pengguna
  stdout.write("Masukkan Nama: ");
  String? nama = stdin.readLineSync();

  stdout.write("Masukkan NIM: ");
  String? nim = stdin.readLineSync();

  stdout.write("Masukkan Jurusan: ");
  String? jurusan = stdin.readLineSync();

  if (nama == '' || nim == '' || jurusan == '') {
    print('Data tidak lengkap, silahkan ulangi!');
    exit(1);
  }

  // Menyimpan input ke dalam Map
  Map<String, dynamic> mahasiswa = {
    'nama': nama,
    'nim': nim,
    'jurusan': jurusan,
  };

  if (mahasiswa.isEmpty) {
    print('Data mahasiswa tidak tersedia.');
```

```
    return;
  }

  print('\n');
  print(
    'Data Mahasiswa: Nama - ${mahasiswa['nama']}, NIM - ${mahasiswa['nim']}, Jurusan - ${mahasiswa['jurusan']}'
  );
}
```

```

print('\nData Mahasiswa:');
mahasiswa.forEach((key, value) {
    print('$key: $value');
});
}

```

12. Simpan kode program. Masukkan perintah untuk menjalankan program di terminal ketik perintah “dart ex5\_input\_data\_array.dart” tekan **enter** untuk menjalankan, dan hasilnya adalah seperti berikut:

```

PROBLEMS  DEBUG CONSOLE  TERMINAL  PORTS  DEVDB
● → P03_DartLanjutan dart ex05_input_data_array.dart
Masukkan Nama: Ikhsan
Masukkan NIM: 1234
Masukkan Jurusan: KSI

Data Mahasiswa: Nama - Ikhsan, NIM - 1234, Jurusan - KSI

Data Mahasiswa:
nama: Ikhsan
nim: 1234
jurusan: KSI
○ → P03_DartLanjutan

```

Tugas Praktikum :

### Latihan 1 – Menampilkan Data

Buatlah sebuah file baru dengan nama tp01\_logika\_tampil\_data.dart, simpan dalam folder P03\_DartLanjutan dan masukkan kode berikut:

```

void main() {
    List<Map<String, dynamic>> mahasiswa = [
        {'nama': 'Andi', 'nim': '2021001', 'jurusan': 'Informatika'},
        {'nama': 'Budi', 'nim': '2021002', 'jurusan': null},
        {'nama': null, 'nim': '2021003', 'jurusan': 'Teknik Komputer'},
    ];
}

```

- Buat blok jika data mahasiswa bernilai kosong ([]),
- Tampilkan Data Mahasiswa dengan Foreach dan for,
- Buat validasi jika ada value yang bernilai null.
- Simpan dan jalankan program.

## Latihan 2 – Membuat Program Login Sederhana

Buatlah sebuah program login dengan ketentuan sebagai berikut:

- Tetapkan username dengan nilai 'admin' dan password dengan nilai '1234'.
- Buat fitur input untuk menerima username dan password dari pengguna.
- Jika input sesuai dengan nilai yang telah ditetapkan, tampilkan pesan "Login berhasil!".
- Jika input tidak sesuai, pengguna harus mengulangi proses login.
- Batasi jumlah percobaan login maksimal 3 kali. Jika pengguna gagal dalam tiga percobaan, tampilkan pesan "Kesempatan habis. Login gagal!".
- Tambahkan validasi agar username dan password tidak boleh kosong atau hanya berisi spasi.

**\*\* 😊 Selamat Mengerjakan 😊 \*\*\***