**Pertemuan 5**

**Pengantar Bahasa Pemrograman Dart**

**Bagian - 3**

A close up of a sign

Description automatically generated

Dibuat Oleh:

Leon Shan Yoedha Adjie

**POLITEKNIK NEGERI MALANG**

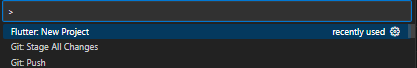
**JURUSAN TEKNOLOGI INFORMARSI**

**2025/2026**

1. **Praktikum 1: Membuat Project Flutter Baru**

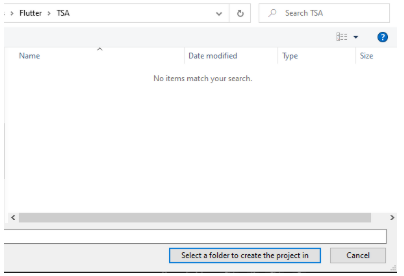
**Langkah 1:**

Buka VS Code, lalu tekan tombol **Ctrl + Shift + P** maka akan tampil *Command Palette*, lalu ketik **Flutter**. Pilih **New Application Project.**



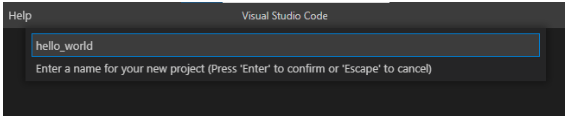
**Langkah 2:**

Kemudian buat folder **sesuai style** laporan praktikum yang Anda pilih. Disarankan pada folder dokumen atau desktop atau alamat folder lain yang tidak terlalu dalam atau panjang. Lalu pilih **Select a folder to create the project in**.



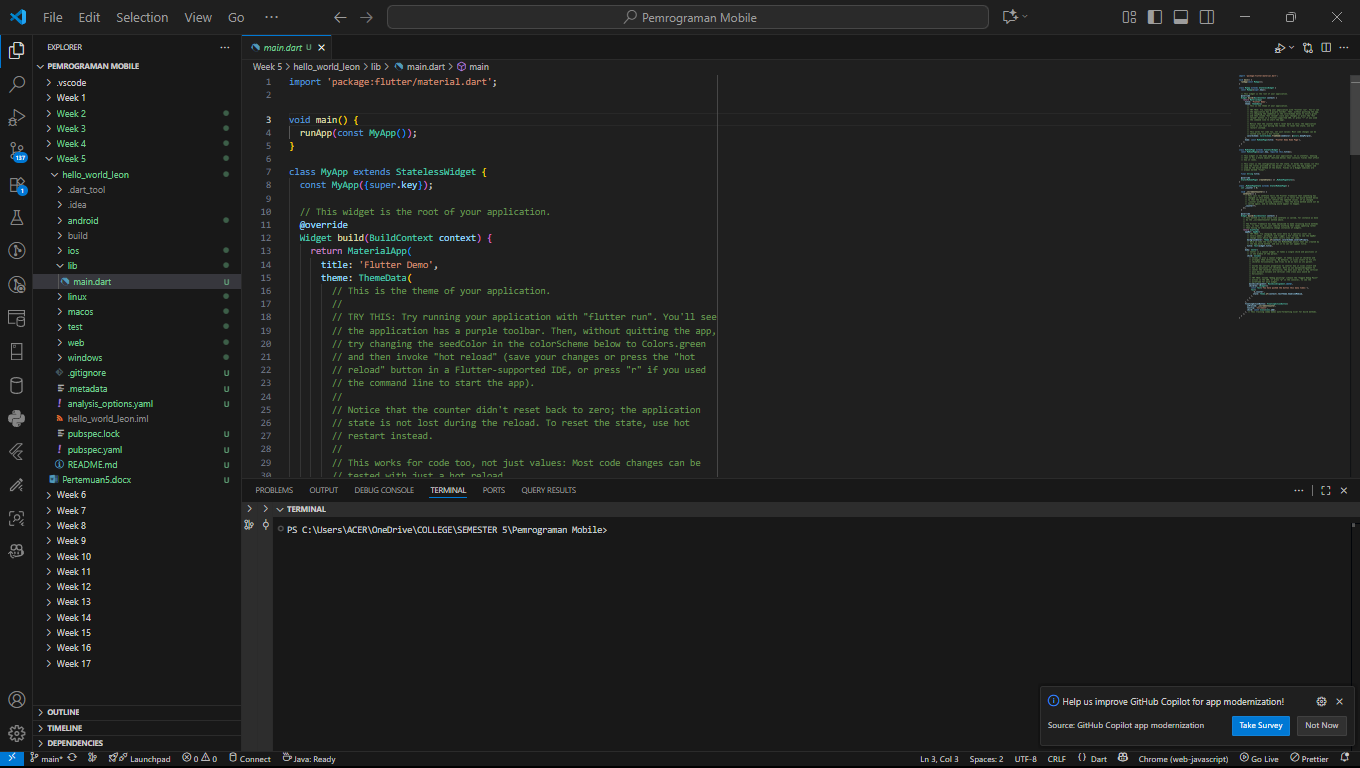
**Langkah 3:**

Buat nama project flutter **hello\_world** seperti berikut, lalu tekan **Enter**. Tunggu hingga proses pembuatan project baru selesai.



**Langkah 4:**

Jika sudah selesai proses pembuatan project baru, pastikan tampilan seperti berikut. Pesan akan tampil berupa "**Your Flutter Project is ready!**" artinya Anda telah berhasil membuat project Flutter baru.



1. **Praktikum 2: Menghubungkan Perangkat Android atau Emulator**

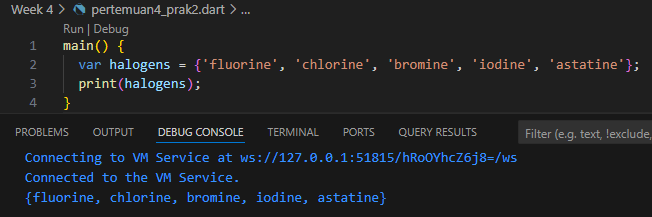
**Langkah 1:**

Ketik atau salin kode program berikut ke dalam fungsi main().



**Langkah 2:**

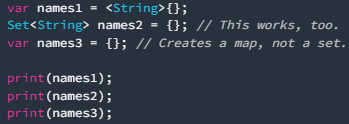
Silakan coba eksekusi (Run) kode pada langkah 1 tersebut. Apa yang terjadi? Jelaskan! Lalu perbaiki jika terjadi error.

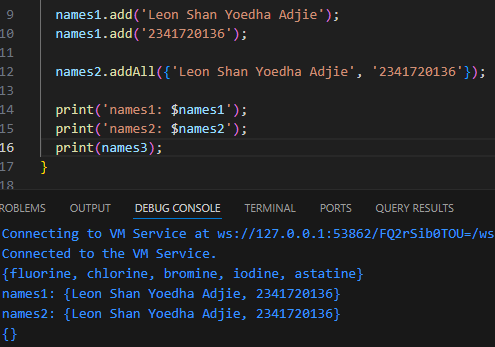


Fungsi var halogens diatas ialah membuat set berisi 5 elemen string kemudian di print.

**Langkah 3:**

Tambahkan kode program berikut, lalu coba eksekusi (Run) kode Anda.



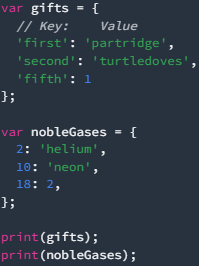
Apa yang terjadi ? Jika terjadi error, silakan perbaiki namun tetap menggunakan ketiga variabel tersebut. Tambahkan elemen nama dan NIM Anda pada kedua variabel Set tersebut dengan dua fungsi berbeda yaitu .add() dan .addAll(). Untuk variabel Map dihapus, nanti kita coba di praktikum selanjutnya. Dokumentasikan code dan hasil di console, lalu buat laporannya. 

Fungsi add() menambahkan value kepada set names1 dengan nama dan NIM dengan dua kali penambahan. Sedangkan addAll() menambahakan value kepada set names2 dengan value yang sama hanya dengan sekali penambahan.

1. **Praktikum 3: Eksperimen Tipe Data Maps**

**Langkah 1:**

Ketik atau salin kode program berikut ke dalam fungsi main().



**Langkah 2:**

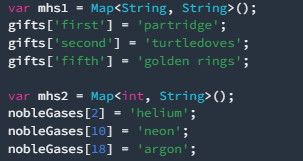
Silakan coba eksekusi (Run) kode pada langkah 1 tersebut. Apa yang terjadi? Jelaskan! Lalu perbaiki jika terjadi error.



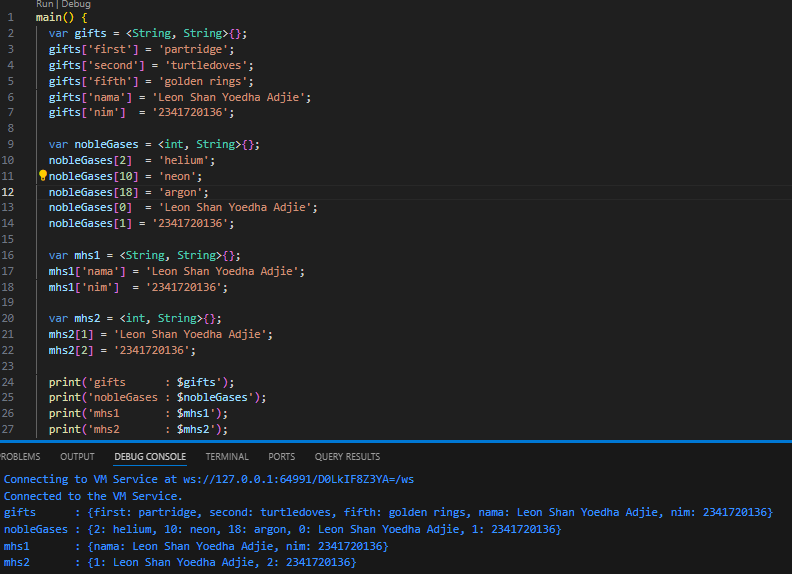
Pada kode tersebut terdapat fungsi untuk mendefinisikan sebuat data Map gifts dengan tiga key dan tiga value yang berbeda, sama halnya juga dengan nobleGases yang lalu di print.

**Langkah 3:**

Tambahkan kode program berikut di dalam *for-loop*, lalu coba eksekusi (Run) kode Anda.



Apa yang terjadi ? Jika terjadi error, silakan perbaiki namun tetap menggunakan Tambahkan elemen nama dan NIM Anda pada tiap variabel di atas (gifts, nobleGases, mhs1, dan mhs2). Dokumentasikan hasilnya dan buat laporannya!

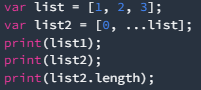


Pada setiap variabel memiliki key-value yang berbeda, jika key-value berupa int-string maka valuenya akan seperti variabel nobleGases dan jika string-string akan sama seperti variabel gifts.

1. **Praktikum 4: Eksperimen Tipe Data List: Spread dan Control-flow Operators**

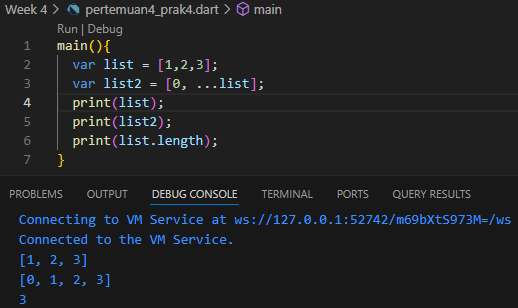
**Langkah 1:**

Ketik atau salin kode program berikut ke dalam fungsi main().



**Langkah 2:**

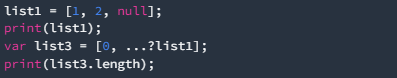
Silakan coba eksekusi (Run) kode pada langkah 1 tersebut. Apa yang terjadi? Jelaskan! Lalu perbaiki jika terjadi error.



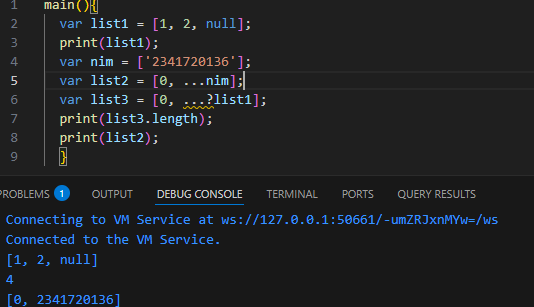
Pada kode tersebut terdapat fungsi untuk mendefinisikan sebuat list dengan value 1,2,3 dan list2 dengan value 0 dan dilanjutkan value dari list kemudian di print tiap list dan list2.length.

**Langkah 3:**

Tambahkan kode program berikut, lalu coba eksekusi (Run) kode Anda.



Apa yang terjadi ? Jika terjadi error, silakan perbaiki. Tambahkan variabel list berisi NIM Anda menggunakan Spread Operators. Dokumentasikan hasilnya dan buat laporannya!



* ... menyebarkan isi list.
* ...? aman bila sumber bisa null (jika null, tidak menambah apa pun).
* Nilai null tetap dihitung sebagai elemen dalam length.

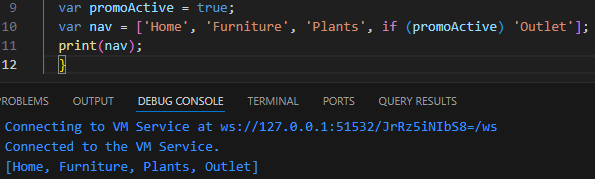
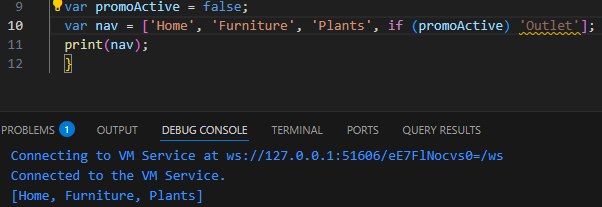
**Langkah 4:**

Tambahkan kode program berikut, lalu coba eksekusi (Run) kode Anda.



Apa yang terjadi ? Jika terjadi error, silakan perbaiki. Tunjukkan hasilnya jika

variabel promoActive ketika true dan false.

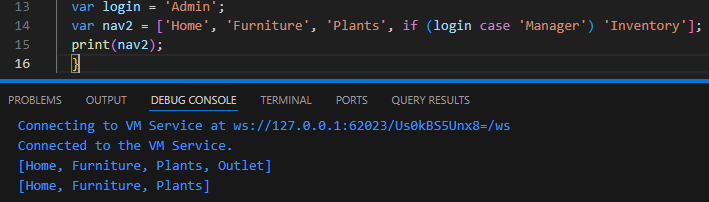
Jika var promoActive bernilai true maka akan print semua variabel pada variabel nav jika false salah satu nilai akan tidak tampil di print.

**Langkah 5:**

Tambahkan kode program berikut, lalu coba eksekusi (Run) kode Anda.



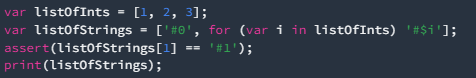
Apa yang terjadi ? Jika terjadi error, silakan perbaiki. Tunjukkan hasilnya jika variabel login mempunyai kondisi lain.



Jika login memiliki kondisi selain ’Manager’ , maka value inventory tidak akan muncul jika di print.

**Langkah 6:**

Tambahkan kode program berikut, lalu coba eksekusi (Run) kode Anda.



Apa yang terjadi ? Jika terjadi error, silakan perbaiki. Jelaskan manfaat **Collection For** dan dokumentasikan hasilnya.



Collection for adalah fitur sintaksis Dart yang memungkinkan kita menyisipkan loop langsung di dalam literal koleksi (List, Set, Map) untuk membangkitkan elemen secara deklaratif tanpa perlu membuat list sementara atau memanggil metode seperti map().toList().

1. **Praktikum 5: Eksperimen Tipe Data Records**

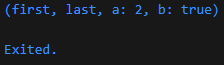
**Langkah 1:**

Ketik atau salin kode program berikut ke dalam fungsi main().



**Langkah 2:**

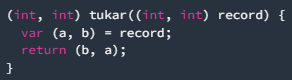
Silakan coba eksekusi (Run) kode pada langkah 1 tersebut. Apa yang terjadi? Jelaskan! Lalu perbaiki jika terjadi error.



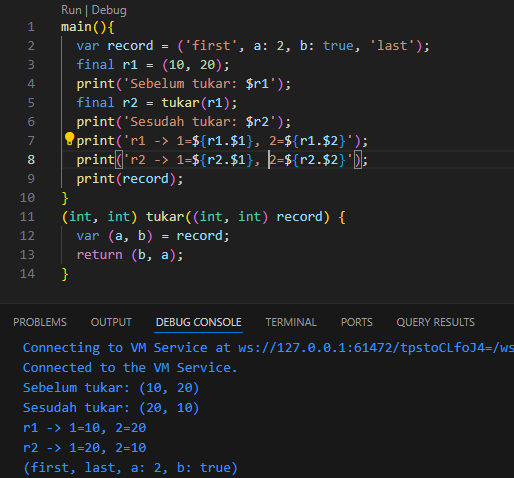
Pada kode tersebut terdapat variabel record dengan 4 value dimana value yang menggunakan field diprint terakhir yaitu a dan b

**Langkah 3:**

Tambahkan kode program berikut, lalu coba eksekusi (Run) kode Anda.



Apa yang terjadi ? Jika terjadi error, silakan perbaiki. Gunakan fungsi tukar() di dalam main() sehingga tampak jelas proses pertukaran value field di dalam Records.



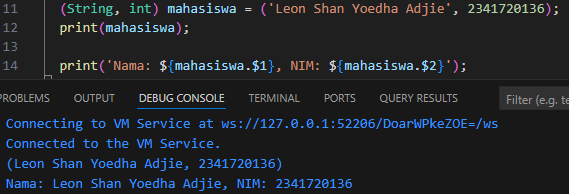
Records bersifat immutable; tukar() tidak mengubah r1, melainkan mengembalikan record baru.

**Langkah 4:**

Tambahkan kode program berikut, lalu coba eksekusi (Run) kode Anda.

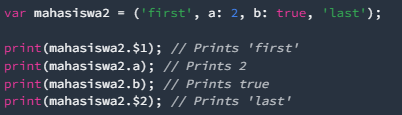


Apa yang terjadi ? Jika terjadi error, silakan perbaiki. Inisialisasi field nama dan NIM Anda pada variabel record mahasiswa di atas. Dokumentasikan hasilnya dan buat laporannya!

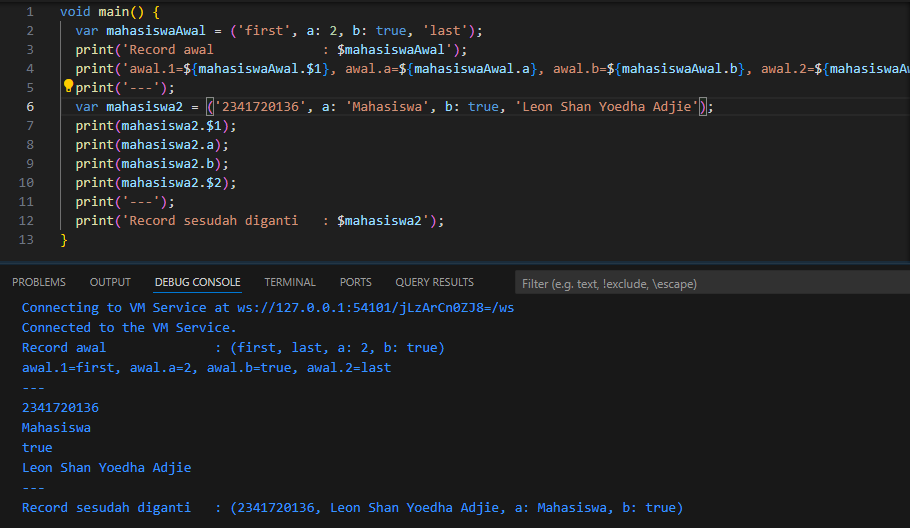


**Langkah 5:**

Tambahkan kode program berikut, lalu coba eksekusi (Run) kode Anda.

****

Apa yang terjadi ? Jika terjadi error, silakan perbaiki. Gantilah salah satu isi record dengan nama dan NIM Anda, lalu dokumentasikan hasilnya dan buat laporannya!



Urutan cetak record bersifat canonical: semua positional dulu, kemudian named terurut namanya.

1. **Tugas Praktikum**
2. Jelaskan yang dimaksud Functions dalam bahasa Dart!

*Jawab:*

Fungsi (function) adalah blok kode yang dapat dipanggil kembali untuk melakukan tugas tertentu. Fungsi punya nama (kecuali anonim), parameter (opsional), nilai kembalian (opsional), dan tipe.

1. Jelaskan jenis-jenis parameter di Functions beserta contoh sintaksnya!

*Jawab:*

* Required positional parameter (wajib dan berurutan)

void greet(String name, int times) {

for (var i = 0; i < times; i++) print('Hi $name');

}

* Optional positional parameter (diapit [ ]), bisa diberi default

void log(String message, [int? code, String prefix = '']) {

print('$prefix$message${code == null ? '' : ' ($code)'}');

}

* Named parameter (diapit { }), default optional; pakai required untuk mewajibkan

void createUser({required String name, int age = 18}) {

print('User $name, age $age');

}

1. Jelaskan maksud Functions sebagai first-class objects beserta contoh sintaknya!

*Jawab:*

Functions sebagai first-class objects Artinya fungsi diperlakukan seperti nilai:

* Bisa disimpan di variabel/struktur data
* Bisa diteruskan sebagai argumen
* Bisa dikembalikan dari fungsi lain

int kali2(int n) => n \* 2;

int Function(int) buatPenambah(int delta) {

return (x) => x + delta; // return fungsi (closure)

}

void main() {

final f = kali2; // simpan fungsi di variabel

print(f(3)); // 6

print(buatPenambah(5)(7)); // 12

}

1. Apa itu Anonymous Functions? Jelaskan dan berikan contohnya!

*Jawab:*

Anonymous Functions Fungsi tanpa nama. Biasanya dipakai sebagai callback singkat atau closure.

final angka = [1, 2, 3];

angka.forEach((n) => print(n \* n))

1. Jelaskan perbedaan Lexical scope dan Lexical closures! Berikan contohnya!

*Jawab:*

Perbedaan Lexical scope vs Lexical closures

* Lexical scope: Ruang lingkup variabel ditentukan oleh posisi kode (statik), bukan saat runtime. Fungsi bagian dalam dapat melihat variabel yang terdefinisi di blok/lingkup luar.

void outer() {

final x = 10;

void inner() {

print(x); // boleh

}

inner();

}

* Lexical closure: Fungsi yang “membawa” (meng-capture) variabel dari scope tempat dia didefinisikan, bahkan setelah scope itu selesai dieksekusi.

Function counter() {

var c = 0;

return () => ++c; // capture c

}

void main() {

final hitung = counter();

print(hitung()); // 1

print(hitung()); // 2

}

1. Jelaskan dengan contoh cara membuat return multiple value di Functions!

*Jawab:*

Return multiple value di Functions Cara yang idiomatik adalah menggunakan Records.

(int, int) hitung(int a, int b) => (a + b, a \* b);

void main() {

final (jumlah, produk) = hitung(3, 4);

print(jumlah); // 7

print(produk); // 12

}

Link Github:

<https://github.com/Holycious/Pemrograman-Mobile/tree/main/Week%204>