

**LAPORAN
PRAKTIKUM PEMROGRAMAN MOBILE**

JOBSHEET 4



Oleh :

Farhan Mawaludin

2341720258

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2025/2026**

Praktikum 1 : Eksperimen Tipe Data List

Langkah 1

```
void main() {  
    var list = [1, 2, 3];  
    assert(list.length == 3);  
    assert(list[1] == 2);  
    print(list.length);  
    print(list[1]);  
  
    list[1] = 1;  
    assert(list[1] == 1);  
    print(list[1]);  
}
```

Pada langkah ini membuat list yang panjangnya berjumlah 3, kemudian dilakukan pengecekan apakah panjangnya == 3 dan list index ke-1 == 2. Kemudian print panjang list dan list pada index ke-1. Setelah itu ganti nilai pada index ke-1 dengan 1 dan cek apakah index ke-1 == 1 lalu print list index ke-1. Maka outputnya :

```
Connecting to VM Service  
Connected to the VM Service  
3  
2  
1
```

Langkah 2

Ubah kode pada langkah 1 menjadi variabel final yang mempunyai index = 5 dengan default value = null. Isilah nama dan NIM Anda pada elemen index ke-1 dan ke-2

```
void main() {  
    final List<dynamic> list = List.filled(5, null);  
  
    list[1] = "Farhan Mawaludin";  
    list[2] = "2341720258";  
  
    // print isi list  
    print("Isi List: $list");  
    print("Index 1 (Nama): ${list[1]}");  
    print("Index 2 (NIM): ${list[2]}");  
}
```

Tidak terjadi error karena list dideklarasikan sebagai final, sehingga variabelnya tidak bisa diganti dengan list baru tetapi isi elemennya tetap dapat diubah. Selain itu, tipe data dynamic? membuat list fleksibel untuk menampung berbagai jenis data, sehingga nama (String) dan NIM (String atau int) bisa ditulis dalam satu list tanpa masalah, selama pengisian dilakukan pada index yang valid. Jadi fungsi final disini supaya variabel list tidak dapat di deklarasikan kembali.

Praktikum 2 : Eksperimen Tipe Data Set

Langkah 1

```
void main(){  
    var halogens = {'fluorine', 'chlorine', 'bromine', '  
    print(halogens);  
}
```

Menampilkan Nilai pada variabel halogens dengan tipe data set

Langkah 2

```
void main() {  
    var halogens = {'fluorine', 'chlorine', 'bromine', '  
    print(halogens);  
  
    var names1 = <String>{};  
    Set<String> names2 = {}; // This works, too.  
  
    names1.add('Farhan Mawaludin');  
    names1.add('2341720258');  
    names2.addAll(['Farhan Mawaludin', '2341720258']);  
  
    print(names1);  
    print(names2);  
}
```

Tidak terjadi eror, add digunakan untuk menambahkan datanya satu per satu ke dalam set. Sedangkan addAll bisa langsung lebih dari 1 dan dalam berbentuk list

Praktikum 3 : Eksperimen Tipe Data Maps

Langkah 1

```
void main() {  
    var gift = {  
        'first': 'partridge',  
        'second': 'turtledoves',  
        'fifth': 1  
    };  
  
    var nobleGases = {  
        2: 'helium',  
        10: 'neon',  
        18: 2,  
    };  
  
    print(gift);  
    print(nobleGases);  
}
```

```
Connecting to VM Service at ws://127.0.0.1:56917/QIcKioMFimY=/ws  
Connected to the VM Service.  
{first: partridge, second: turtledoves, fifth: 1}  
{2: helium, 10: neon, 18: 2}
```

Tidak terjado error, disini menampilkan key dan value dari variabel gift dan nobleGases yang bertipe map.

Langkah 2

```
void main() {
    var gifts = {'first': 'partridge', 'second': 'turtledoves', 'fifth':
    'golden rings'};

    var nobleGases = {2: 'helium', 10: 'neon', 18: 'argon'};

    var mhs1 = Map<String, String>();    The value of the local variable
    gifts['first'] = 'partridge';
    gifts['second'] = 'turtledoves';
    gifts['fifth'] = 'golden rings';

    var mhs2 = Map<int, String>();    The value of the local variable 'mh
    nobleGases[2] = 'helium';
    nobleGases[10] = 'neon';
    nobleGases[18] = 'argon';

    print(gifts);
    print(nobleGases);
}
```

DEBUG CONSOLE ... Filter (e.g. text, !exclude, \escape)

Connecting to VM Service at ws://127.0.0.1:64739/gYmI8_aLiSg=/ws
Connected to the VM Service.
{first: partridge, second: turtledoves, fifth: golden rings}
{2: helium, 10: neon, 18: argon}

Exited.

Tidak terjadi error, pada gifts untuk key 'fifth' diubah menjadi golden rings dan pada nobleGases untuk key '18' diubah menjadi argon.

Langkah 3

Menambahkan elemen nama dan NIM pada setiap variabel

```
gifts.addAll({'nama': 'Farhan Mawaludin', 'nim': '2341720258'});
nobleGases.addAll({'20': 'Farhan Mawaludin', '21': '2341720258'});
mhs1.addAll({'nama': 'Farhan Mawaludin', 'nim': '2341720258'});
mhs2.addAll({'20': 'Farhan Mawaludin', '21': '2341720258'});

print(gifts);
print(nobleGases);
print(mhs1);
print(mhs2);
```

Connected to the VM Service.
{first: partridge, second: turtledoves, fifth: golden rings, nama: Farhan Mawaludin, nim: 2341720258}
{2: helium, 10: neon, 18: argon, 20: Farhan Mawaludin, 21: 2341720258}
{nama: Farhan Mawaludin, nim: 2341720258}
{20: Farhan Mawaludin, 21: 2341720258}

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

Langkah 1

```
Run | Debug | Windsurf: Refactor | Explain | Generate Function Con
void main() {
  var list = [1, 2, 3];
  var list2 = [0, ...list];
  print(list1);    Undefined name 'list1'.
  print(list2);
  print(list2.length);
}
```

Connected to the VM Service.

praktikum4.dart:4:9: Error: Undefined name 'list1'.
print(list1);
 ^^^^^

Terjadi error karena memang tidak ada variabel list1 .Kode perbaikan

```
void main() {
  var list = [1, 2, 3];
  var list2 = [0, ...list];
  print(list);
  print(list2);
  print(list2.length);
}
```

Connected to the VM Service.

[1, 2, 3]
[0, 1, 2, 3]
4

Menampilkan variabel list , list2 dan panjang list2

Langkah 2

```
list1 = [1, 2, null];    Undefined
print(list1);    Undefined name 'l
var list3 = [0, ...?list1];    Und
print(list3.length);
}
```

Setelah menambahkan ini terjadi error karena variabel list1 belum di deklarasikan. Kode perbaikan dan menambahkan nama dan nim menggunakan spread operator

```
var list1 = [1, 2, null];
var nim = [2341720258];
var name = ["Farhan Mawaludin"];
print(list1);
var list3 = [0, ...?list1, ...nim, ...name];    The receiver can't
print(list3);
print(list3.length);
```

[1, 2, null]
[0, 1, 2, null, 2341720258, Farhan Mawaludin]
6

Langkah 3

```
var nav = ['Home', 'Furniture', 'Plants', if (promoActive) 'Outlet'];  
print(nav);
```

Terjadi error karena promoActivenya belum di deklarasikan.

Kode perbaikan:

```
var promoActive = true;  
var nav = ['Home', 'Furniture', 'Plants', if (promoActive) 'Outlet'];  
print(nav);
```

Ketika True

```
6  
[Home, Furniture, Plants, Outlet]
```

Ketika false

```
6  
[Home, Furniture, Plants]
```

Langkah 4

```
var nav2 = [  
  'Home',  
  'Furniture',  
  'Plants',  
  if (login case 'Manager') 'Inventory',  
];  
print(nav2);
```

Terjadi error karena tidak ada variabel login.

Kode perbaikan:

```
var login = 'Manager';  
var nav2 = [  
  'Home',  
  'Furniture',  
  'Plants',  
  if (login case 'Manager') 'Inventory',  
];  
print(nav2);  
[Home, Furniture, Plants, Inventory]
```

Jika login memiliki kondisi lain, login = 'Admin' maka hasilnya :

```
var login = 'Admin';  
var nav2 = [  
  'Home',  
  'Furniture',  
  'Plants',  
  if (login case 'Manager') 'Inventory',  
];  
print(nav2);  
[Home, Furniture, Plants]
```

Langkah 5

```
var listOfInts = [1, 2, 3];
var listOfStrings = ['#0', for (var i in listOfInts) '#$i'];
assert(listOfStrings[1] == '#1');
print(listOfStrings);
```

```
[#0, #1, #2, #3]
```

Tidak terjadi error, mempermudah membuat list baru dari list lain tanpa harus bikin loop terpisah.

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

Langkah 1

```
void main() {
  var record = ('first', a: 2, b: true, 'last');
  print(record);
}
```

```
Connecting to VM Service at ws://127.0.0.1:6006
Connected to the VM Service.
(first, last, a: 2, b: true)
```

Menampilkan elemen pada variabel record

Langkah 2

```
void main() {
  var record = ('first', a: 2, b: true, 'last');
  print(record);

  var angka = (2,3);
  print(tukar(angka));
}
```

```
Windsurf: Refactor | Explain | Generate Function Comment | X
(int, int) tukar((int, int) record) {
  var (a, b) = record;
  return (b, a);
}
```

```
(first, last, a: 2, b: true)
(3, 2)
```

Membuat fungsi tukar, jadi menukar (2,3) menjadi (3,2)

Langkah 3

```
(String, int) mahasiswa;
print(mahasiswa); // The non-null assertion is required
```

Terjadi eror karena sudah di inisialisasi tapi nilainya tidak ada.
Kode perbaikan:

```
(String, int) mahasiswa = ('Farhan Mawaludin', 2341720258);
print(mahasiswa);
(Farhan Mawaludin, 2341720258)
```

Langkah 4

```
var mahasiswa2 = {'first', a: 2, b: true, 'last'};

print(mahasiswa2.$1); // Prints 'first'
print(mahasiswa2.a); // Prints 2
print(mahasiswa2.b); // Prints true
print(mahasiswa2.$2); // Prints 'last'
```

```
first
2
true
last
```

Tidak terjadi error karena cara pemanggilannya yang sudah sesuai dengan tipe data records
Lalu Gantilah salah satu isi record dengan nama dan NIM Anda.

```
var mahasiswa2 = {'Farhan Mawaludin', a: 2341720258, b: true, 'last'};

print(mahasiswa2.$1); // Prints 'first'
print(mahasiswa2.a); // Prints 2
print(mahasiswa2.b); // Prints true
print(mahasiswa2.$2); // Prints 'last'
}
```

```
Farhan Mawaludin
2341720258
true
last
```

Tugas Praktikum

1. Jelaskan yang dimaksud Functions dalam bahasa Dart!
Jawab : Function adalah sebuah blok kode yang dibuat untuk melakukan tugas tertentu, selain itu jika fungsinya ingin digunakan lagi maka kira tidak perlu membuat lagi.
2. Jelaskan jenis-jenis parameter di functions beserta contoh sintaksnya
Jawab :

- **Positional parameter (wajib, urutannya penting)**

```
void tambah(int a, int b) {
  print(a + b);
}
```

- **Optional positional parameter**

```
void sapa(String nama, [String? pesan]) {
  print("Halo $nama ${pesan ?? ''}");
}
```


- **Named parameter**

```
Windsurf: Refactor | Explain | Generate Function Comment | X
void buatAkun({required String nama, int umur = 18}) {
  print("Nama: $nama, Umur: $umur");
}
```

3. Jelaskan maksud Functions sebagai first-class objects beserta contoh sintaknya!
Jawab : di Dart fungsi diperlakukan sebagai nilai , bisa disimpan di variabel, dikirim ke fungsi lain, atau dikembalikan dari fungsi.

```
void sapa(String nama) {
  print("Halo, $nama!");
}

Run | Debug | Windsurf: Refactor | Explain |
void main() {
  var fungsi = sapa;
  fungsi("Farhan");
}
```

4. Apa itu Anonymous Functions? Jelaskan dan berikan contohnya!
Jawab : Fungsi tanpa nama, biasanya dipakai cepat sebagai argumen atau callback.

```
void main() {
  var angka = [1, 2, 3, 4];

  angka.forEach((n) {
    print(n * 2);
  });
}
```

5. Jelaskan perbedaan Lexical scope dan Lexical closures! Berikan contohnya!
Jawab :

- **Lexical Scope**

Variabel hanya bisa diakses dalam ruang lingkup (scope) tempat ia didefinisikan.

```
void main() {
  var a = 10;
  Windsurf: Refactor | Explain |
  void cetakA() {
    print(a);
  }
  cetakA();
}
```

- **Lexical Closure**

Function “mengingat” variabel dari scope luar meski scope luar sudah selesai.

```
void sapa(String nama) {
  print("Halo, $nama!");
}

Run | Debug | Windsurf: Refactor | Explain | Generate
void main() {
  String nama = 'Farhan';
  sapa(nama);
}
```

6. Jelaskan dengan contoh cara membuat return multiple value di Functions!
Jawab :membuat return multiple menggunakan tipe data records

```
Windsurf: Refactor | Explain | Generate Function Comment |>  
(String, int) getMahasiswa() {  
    return ('Farhan', 2341720258);  
}  
  
Run | Debug | Windsurf: Refactor | Explain | Generate Function Comm  
void main() {  
    var user = getMahasiswa();  
    print("Nama: ${user.$1}, NIM: ${user.$2}");  
}
```

Link Repo: [FarhanMawaludin/Pemrograman-mobile-2025](https://github.com/FarhanMawaludin/Pemrograman-mobile-2025)