

**Pemrograman Mobile**  
**Jobsheet 4**



**From:**

DAVIS MAULANA HERMANTO

**Class:**

3 I

**Absence:**

05

**Student Number Identity:**

2241720255

**Department:**

Information Technology

**Study Program:**

Informatics Engineering

## Practicum 1

```
Run | Debug | Codeium: Refactor | Explain | Generate
13 void main(){
14     var list = [1,2,3];
15     assert(list.length == 3);
16     assert(list[1] == 2);
17     print(list.length);
18     print(list[1]);
19
20     list[1] = 1;
21     assert(list[1] == 1);
22     print(list[1]);
23 }
```

PROBLEMS   DEBUG CONSOLE   ...   🔍   ☰   ^   ✕

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

Connecting to VM Service at ws://127.0.0.1:59605/Tnq2DfokZQU=/ws  
Connected to the VM Service.

3  
2  
1

Exited.

- at the first line, there is an array. we're using var for creating a variable in dart
- `assert(list[1] == 2)` is like a condition process, checking if the element at index 1 is 2. Dart uses 0-based indexing, so `list[1]` refers to the second element, which is indeed 2.
- `print(list.length)` prints the length of the list, which is 3.
- `print(list[1])` prints the second element of the list, which is 2.
- `list[1] = 1` changes the value at index 1 (the second element of the list) from 2 to 1. The list now looks like `[1, 1, 3]`.
- `assert(list[1] == 1)` checks if the second element is now 1, which is true.

```
Run | Debug | Codeium: Refactor | Explain | Generate Function Comment | X
13 void main() {
14     final List<String?> list = List.filled(5, null);
15     list[1] = "Davis Maulana Hermanto";
16     list[2] = "2241720255";
17
18     print(list);
19     print(list[1]);
20     print(list[2]);
21 }
22
```

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

Connecting to VM Service at ws://127.0.0.1:60410/A98vF9R-07s=/ws  
Connected to the VM Service.  
[null, Davis Maulana Hermanto, 2241720255, null, null]  
Davis Maulana Hermanto  
2241720255  
  
Exited.

## Practicum 2

```
Run | Debug | Codeium: Refactor | Explain | Generate Function Comment | X
1 void main(){
2     var halogens = {'fluorine', 'chlorine', 'bromine', 'iodine', 'astatine'}
3     print(halogens);
4 }
5
```

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

Connecting to VM Service at ws://127.0.0.1:60494/SLPcLcQmX4A=/ws  
Connected to the VM Service.  
{fluorine, chlorine, bromine, iodine, astatine}  
  
Exited.

- The result was like this. and I didn't find the error.

```

16 void main() {
17     var names1 = <String>{};
18     Set<String> names2 = {};
19     var names3 = {};
20
21     // Adding element by using .add()
22     names1.add("Davis Maulana Hermanto");
23     names1.add("2241720255");
24
25     // Adding element by using .addAll()
26     names2.addAll(["Davis Maulana Hermanto", "2241720255"]);
27
28     print("names1 (Set): $names1");
29     print("names2 (Set): $names2");
30
31     // Map names3
32     names3["Name"] = "Davis Maulana Hermanto";
33     names3["NIM"] = "2241720255";
34
35
36     print("names3 (Map): $names3");
37 }

```

DEBUG CONSOLE

...

Filter (e.g. text, \exclude, \escape)

```

Connecting to VM Service at ws://127.0.0.1:61263/bpALcb-NENQ=/ws
Connected to the VM Service.
names1 (Set): {Davis Maulana Hermanto, 2241720255}
names2 (Set): {Davis Maulana Hermanto, 2241720255}
names3 (Map): {Name: Davis Maulana Hermanto, NIM: 2241720255}

```

## Practicum 3

```
1  void main() {
2  var gifts = {
3      // Key:    Value
4      'first': 'partridge',
5      'second': 'turtledoves',
6      'fifth': 1
7  };
8  var nobleGases = {
9      2: 'helium',
10     10: 'neon',
11     18: 2,
12 };
13
14 print(gifts);
15 print(nobleGases);
16 }
17
```

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

Connecting to VM Service at ws://127.0.0.1:62280/lBGDgKzkv1Y=/ws  
Connected to the VM Service.  
{first: partridge, second: turtledoves, fifth: 1}  
{2: helium, 10: neon, 18: 2}  
  
Exited.

- The result was like this. var gifts is a map where each key is a string, and the corresponding value is either a string or an integer.
- var nobleGases is another map where the key is an integer and mixed value

```

18  void main() {
19      var mhs1 = Map<String, String>();
20      mhs1['first'] = 'partridge';
21      mhs1['second'] = 'turtledoves';
22      mhs1['fifth'] = 'golden rings';
23
24      var mhs2 = Map<int, String>();
25      mhs2[2] = 'helium';
26      mhs2[10] = 'neon';
27      mhs2[18] = 'argon';
28
29      print(mhs1);
30      print(mhs2);
31  }
32

```

DEBUG CONSOLE

...

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

Connecting to VM Service at ws://127.0.0.1:62709/EuI2wcdCTLM=/ws

Connected to the VM Service.

{first: partridge, second: turtledoves, fifth: golden rings}

{2: helium, 10: neon, 18: argon}

Exited.

- I've fixed the code by changing the defining variables mhs1 and mhs2 and adding print codes. and the result was like this.

```
1  ∨ void main() {
2  ∨    var gifts = {
3      // Key:    Value
4      'first': 'partridge',
5      'second': 'turtledoves',
6      'fifth': 1,
7      'name' : 'Davis Maulana Hermanto',
8      'nim' : '2241720255',
9  };
10 ∨    var nobleGases = {
11        2: 'helium',
12        10: 'neon',
13        18: 2,
14        20: 'Davis Maulana Hermanto',
15        22: '2241720255',
16    };
17
18    var mhs1 = Map<String, String>();
19    mhs1['first'] = 'partridge';
20    mhs1['second'] = 'turtledoves';
21    mhs1['fifth'] = 'golden rings';
22    mhs1['name'] = 'Davis Maulana Hermanto';
23    mhs1['nim'] = '2241720255';
24
25    var mhs2 = Map<int, String>();
26    mhs2[2] = 'helium';
27    mhs2[10] = 'neon';
28    mhs2[18] = 'argon';
29    mhs2[20] = 'Davis Maulana Hermanto';
30    mhs2[22] = '2241720255';
31
```

```

32     print(mhs1);
33     print(mhs2);
34
35     print(gifts);
36     print(nobleGases);
37 }
38

```

```

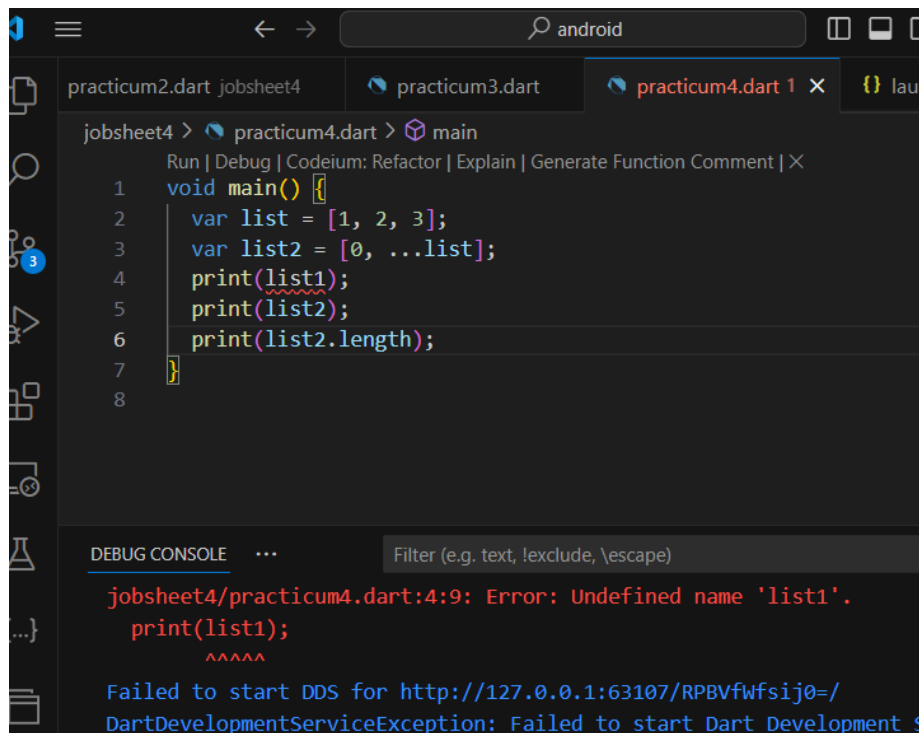
Connecting to VM Service at ws://127.0.0.1:63004/55EHat03uTU=/ws
Connected to the VM Service.
{first: partridge, second: turtledoves, fifth: golden rings, name: Davis Maulana Hermanto, nim: 2241720255}
{2: helium, 10: neon, 18: argon, 20: Davis Maulana Hermanto, 22: 2241720255}
{first: partridge, second: turtledoves, fifth: 1, name: Davis Maulana Hermanto, nim: 2241720255}
{2: helium, 10: neon, 18: 2, 20: Davis Maulana Hermanto, 22: 2241720255}

Exited.

```

## Practicum 4

### Step 1





- There is an error like this because there is a typo in the first print. it should be list not list1

```

jobsheet4 > practicum4.dart > main
Run | Debug | Codeium: Refactor | Explain | Generate Function Comment | X
1 void main() {
2   var list = [1, 2, 3];
3   var list2 = [0, ...list];
4   print(list);
5   print(list2);
6   print(list2.length);
7 }
8

DEBUG CONSOLE ... Filter (e.g. text, lexclude, \escape)
Connecting to VM Service at ws://127.0.0.1:63154/VKUEAE9dDEU=/ws
Connected to the VM Service.
[1, 2, 3]
[0, 1, 2, 3]
4
Exited.

```

- done I've fixed that

### Step 3

```

jobsheet4 > practicum4.dart > main
Run | Debug | Codeium: Refactor | Explain | Generate Function Comment | X
1 void main() {
2   var list = [1, 2, 3];
3   var list2 = [0, ...list];
4   print(list);
5   print(list2);
6   print(list2.length);
7
8   list1 = [1, 2, null];
9   print(list1);
10  var list3 = [0, ...?list1];
11  print(list3.length);
12 }

DEBUG CONSOLE ... Filter (e.g. text, lexclude, \escape)
Connecting to VM Service at ws://127.0.0.1:63242/P02-C7hl0lw=/ws
jobsheet4/practicum4.dart:8:3: Error: Setter not found: 'list1'.
  list1 = [1, 2, null];
  ^^^^^
jobsheet4/practicum4.dart:9:9: Error: Undefined name 'list1'.
  print(list1);
  ^^^^^
jobsheet4/practicum4.dart:10:23: Error: Undefined name 'list1'.
  var list3 = [0, ...?list1];
  ^^^^^
jobsheet4/practicum4.dart:10:23: Error: Unexpected type 'invalid-type' of a spread. Expected 'dynamic' or an Iterable.
  var list3 = [0, ...?list1];

```

- The error will be like this because in the list1 variable definition, there is a null so we haven't added var before the name of the variable on the variable definition.

```

1 void main() {
2     var list = [1, 2, 3];
3     var list2 = [0, ...list];
4     print(list);
5     print(list2);
6     print(list2.length);
7
8     var list1 = [1, 2, null];
9     print(list1);
10    var list3 = [0, ...?list1];
11    print(list3);
12    print(list3.length);
13 }

```

DEBUG CONSOLE

...

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

Connecting to VM Service at ws://127.0.0.1:63433/ywjC6Pgl6T0=/ws  
Connected to the VM Service.

[1, 2, 3]

[0, 1, 2, 3]

4

[1, 2, null]

[0, 1, 2, null]

4

Exited.

- The ...?list1 operator is used to spread the contents of list1 into list3

## Step 4

```
1 void main() {
2     var list = [1, 2, 3];
3     var list2 = [0, ...list];
4     print(list);
5     print(list2);
6     print(list2.length);
7
8     var list1 = [1, 2, null];
9     print(list1);
10    var list3 = [0, ...?list1];
11    print(list3);
12    print(list3.length);
13
14    var promoActive = false;
15    var nav = ['Home', 'Furniture', 'Plants', if (promoActive) 'Outlet'];
16    print(nav);
17 }
18
```

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

Connecting to VM Service at ws://127.0.0.1:63557/Connm9HZnvU=/ws  
Connected to the VM Service.

[1, 2, 3]  
[0, 1, 2, 3]  
4  
[1, 2, null]  
[0, 1, 2, null]  
4  
[Home, Furniture, Plants]  
Exited.

```
1 void main() {
2     var list = [1, 2, 3];
3     var list2 = [0, ...list];
4     print(list);
5     print(list2);
6     print(list2.length);
7
8     var list1 = [1, 2, null];
9     print(list1);
10    var list3 = [0, ...?list1];
11    print(list3);
12    print(list3.length);
13
14    var promoActive = true;
15    var nav = ['Home', 'Furniture', 'Plants', if (promoActive) 'Outlet'];
16    print(nav);
17 }
18
```

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

Connecting to VM Service at ws://127.0.0.1:63608/S1ZpGP0TR6s=/ws  
Connected to the VM Service.

[1, 2, 3]  
[0, 1, 2, 3]  
4  
[1, 2, null]  
[0, 1, 2, null]  
4  
[Home, Furniture, Plants, Outlet]  
Exited.

- The output was like this. there was an error before I ran this code so I've fixed this by adding promoActive variable before the array definition

## Step 5

```
18  var login = 'Manager';
19  var nav2 = [
20      'Home',
21      'Furniture',
22      'Plants',
23      if (login case 'Manager') 'Inventory'
24  ];
25  print(nav2);
26  }
27
```

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

```
Connecting to VM Service at ws://127.0.0.1:63786/dexHQL_ZyFY=/ws
Connected to the VM Service.
[1, 2, 3]
[0, 1, 2, 3]
4
[1, 2, null]
[0, 1, 2, null]
4
[Home, Furniture, Plants, Outlet]
[Home, Furniture, Plants, Inventory]
```

- The output was like this. As before there was an error and we had to add a variable and we have to define also. In the array, there is a condition if the login variable contains 'Manager', 'Inventory' will be added to the array

## Step 6

```

27   var listOfInts = [1, 2, 3];
28   var listOfStrings = ['#0', for (var i in listOfInts) '#$i'];
29   assert(listOfStrings[1] == '#1');
30   print(listOfStrings);
31 }
32

```

DEBUG CONSOLE

...

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

Connecting to VM Service at ws://127.0.0.1:63859/\_cu7wD18780=/ws

Connected to the VM Service.

[1, 2, 3]

[0, 1, 2, 3]

4

[1, 2, null]

[0, 1, 2, null]

4

[Home, Furniture, Plants, Outlet]

[Home, Furniture, Plants, Inventory]

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

- The output was like this. we have converted the integer array to a string array by adding # before the number. and we are using looping on the array to add # in every number on the array. so we shouldn't type or add the string array manually.

## Practicum 4

### Step 1

```
1 void main() {  
2     var record = ('first', a: 2, b: true, 'last');  
3     print(record);  
4 }  
5
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL ... 🔍 ☰ ^ ✕

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

Connecting to VM Service at ws://127.0.0.1:64372/1nb0sRqVF38=/ws  
Connected to the VM Service.  
(first, last, a: 2, b: true)  
  
Exited.

- The output was like this. We can add mixed values like that by using that method.

### Step 3

```
Run | Debug | Codeium: Refactor | Explain | Generate Function Comment | X
1 void main() {
2     var record = ('first', a: 2, b: true, 'last');
3     print(record);
4
5
6     var test = (1, 2);
7     var swapped = tukar(test);
8
9     print('-----');
10
11     print('Original: $record');
12     print('Swapped: $swapped');
13 }
14
Codeium: Refactor | Explain | Generate Function Comment | X
15 (int, int) tukar((int, int) record) {
16     var (a, b) = record;
17     return (b, a);
18 }
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL ... 🔍 ☰ ^ >

Filter (e.g. text, lexclude, \escape)

Connecting to VM Service at ws://127.0.0.1:64782/z8YALsw780I=/w  
Connected to the VM Service.  
(first, last, a: 2, b: true)  
-----  
Original: (first, last, a: 2, b: true)  
Swapped: (2, 1)  
  
Exited.

- The output was like this. The tukar method is to change the index position of the array.

#### Step 4

```
15 (String, int) mahasiswa = ('Davis Maulana Hermanto', 2241720255);
16 print(mahasiswa);
17 }
18
Codeium: Refactor | Explain | Generate Function Comment | ✕
19 (int, int) tukar((int, int) record) {
20     var (a, b) = record;
21     return (b, a);

```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL ... Filter (e.g. text, lexclude, \escape)

Connecting to VM Service at ws://127.0.0.1:65181/rC3tfW71e0s=/ws  
Connected to the VM Service.  
(first, last, a: 2, b: true)  
-----  
Original: (first, last, a: 2, b: true)  
Swapped: (2, 1)  
-----  
(Davis Maulana Hermanto, 2241720255)

#### Step 5

```
17
18     print('-----');
19     ⚡
20     var mahasiswa2 = ('first', a: 2, b: true, 'last');
21
22     print(mahasiswa2.$1); // Prints 'first'
23     print(mahasiswa2.a); // Prints 2
24     print(mahasiswa2.b); // Prints true
25     print(mahasiswa2.$2); // Prints 'last'
26 }
27
Codeium: Refactor | Explain | Generate Function Comment | ✕

```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL ... Filter (e.g. text, lexclude, \escape)

Connected to the VM Service.  
(first, last, a: 2, b: true)  
-----  
Original: (first, last, a: 2, b: true)  
Swapped: (2, 1)  
-----  
(Davis Maulana Hermanto, 2241720255)  
-----  
first  
2  
true  
2



```
19
20 var mahasiswa2 = ['Davis Maulana Hermanto', a: 2241720255, b: true, '']
21
22 print(mahasiswa2.$1); // Prints 'first'
23 print(mahasiswa2.a); // Prints 2
24 print(mahasiswa2.b); // Prints true
25 print(mahasiswa2.$2); // Prints 'last'
26 }
27
```

Codeium: Refactor | Explain | Generate Function Comment | X

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL ... Filter (e.g. text, lexclude, \escape) 🔍

Connected to the VM Service.  
(first, last, a: 2, b: true)  
-----  
Original: (first, last, a: 2, b: true)  
Swapped: (2, 1)  
-----  
(Davis Maulana Hermanto, 2241720255)  
-----  
Davis Maulana Hermanto  
2241720255

## Practicum Assignment

1. Explain what is meant by Functions in Dart language!
  - a. Functions in Dart are re-executable blocks of code that are used to wrap logic that can be called multiple times without rewriting the code. Functions can take arguments and return values.
2. Explain the types of parameters in Functions along with examples of their syntax!
  - a. Positional Parameters

```
void greet(String name, int age) {  
    print('Hello $name, you are $age years old.');
```

```
}  
  
void main() {  
    greet('Alice', 25); // Output: Hello Alice, you are 25 years old.  
}
```

```
...
```

- b. Named Parameters

```
void greet({required String name, int age = 30}) {  
    print('Hello $name, you are $age years old.');
```

```
}  
  
void main() {  
    greet(name: 'Bob'); // Output: Hello Bob, you are 30 years old.  
}
```

```
...
```

- c. Optional Positional Parameters

```
void greet(String name, [int age = 30]) {  
    print('Hello $name, you are $age years old.');
```

```
}  
  
void main() {  
    greet('Charlie'); // Output: Hello Charlie, you are 30 years old.  
}
```

```
...
```

3. Explain the meaning of Functions as first-class objects along with examples of their syntax!

- a. Functions in Dart are first-class objects, meaning they can be stored in variables, passed as arguments to other functions, and returned from other functions.

```
void printMessage(String message) {  
    print(message);  
}  
  
void executeFunction(void Function(String) func, String msg) {  
    func(msg);  
}  
  
void main() {  
    var myFunc = printMessage;  
    executeFunction(myFunc, 'Hello World!'); // Output: Hello World!  
}
```

4. What are Anonymous Functions? Explain and give examples!
- a. Anonymous Functions are functions without a name, often used as arguments for other functions or when a temporary function is needed.

```
void main() {  
    var list = [1, 2, 3, 4];  
    list.forEach((item) {  
        print('Item: $item'); // Output: Item: 1, Item: 2, etc.  
    });  
}
```

5. Explain the difference between Lexical scope and Lexical closures! Give examples!
- a. Lexical Scope: Refers to the visibility and accessibility of variables based on their physical location in the code. Variables can only be accessed from within the block where they are defined.

```
void main() {
  var a = 10;
  void printA() {
    print(a); // Output: 10
  }
  printA();
}
```

- b. Lexical Closures: A function that "captures" variables from its surrounding environment, allowing it to access those variables even after the outer scope has finished executing.

```
Function createCounter() {
  int count = 0;
  return () {
    count++;
    return count;
  };
}

void main() {
  var counter = createCounter();
  print(counter()); // Output: 1
  print(counter()); // Output: 2
}
```

6. Explain with examples how to create multiple return values in Functions!
- Dart does not natively support returning multiple values, but you can use records or tuples (from Dart 3.0 onward) to return more than one value.

```
(int, String) getDetails() {  
    return (123, 'John Doe');  
}  
  
void main() {  
    var (id, name) = getDetails();  
    print('ID: $id, Name: $name'); // Output: ID: 123, Name: John Doe  
}
```