



Mata Kuliah : Pemrograman Mobile
Program Studi : D4 – Teknik Informatika
Semester : 5

Kelas : TI-3E
NIM : 2341720026
Nama : Vanessa Cristin Natalia
Jobsheet Ke- : 4 - Pengantar Bahasa Pemrograman Dart – Bagian 3

Laporan Praktikum

Praktikum 1: Eksperimen Tipe Data List

Langkah	Jawaban/Deskripsi
1	<p>Ketik atau salin kode program berikut ke dalam void main()</p> <pre>1 // Praktikum 1 2 void main() { 3 var list = [1, 2, 3]; 4 assert(list.length == 3); 5 assert(list[1] == 2); 6 print(list.length); 7 print(list[1]); 8 9 list[1] = 1; 10 assert(list[1] == 1); 11 print(list[1]); 12 }</pre> <p>Outputnya: List awal [1,2,3] memiliki panjang 3. Elemen indeks ke-1 awalnya bernilai 2, lalu diubah menjadi 1. Perintah assert memastikan kondisi benar, dan print menampilkan panjang list serta nilai elemen sebelum dan sesudah diubah.</p>



	<pre>3 2 1 Exited.</pre>
2	<p>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. Lalu print dan capture hasilnya. Apa yang terjadi ? Jika terjadi error, silakan perbaiki.</p> <pre>1 // Modifikasi Tugas Praktikum 1 - Eksperimen Tipe Data List 2 void main() { 3 final List<dynamic> list = List.filled(5, null); 4 5 list[1] = "Vanessa Cristin Natalia"; 6 list[2] = "2341720026"; 7 8 print("Nama: \${list[1]}"); 9 print("NIM: \${list[2]}"); 10 }</pre> <p>Outputnya tidak terjadi error karena kombinasi dari dynamic + final + List.filled(5, null) memungkinkan list menampung null di awal, lalu diisi ulang dengan string Nama dan NIM</p> <pre>Nama: Vanessa Cristin Natalia NIM: 2341720026 Exited.</pre>



Praktikum 2: Eksperimen Tipe Data Set

Langkah	Jawaban/Deskripsi
1	<p>Ketik atau salin kode program berikut ke dalam fungsi <code>main()</code>.</p> <pre>1 void main() { 2 var halogen = {'fluorine', 'chlorine', 'bromine', 'iodine', 'astatine'}; 3 4 print(halogen); 5 }</pre> <p>Silakan coba eksekusi (Run) kode pada langkah 1 tersebut. Apa yang terjadi? Jelaskan! Lalu perbaiki jika terjadi error.</p> <pre>{fluorine, chlorine, bromine, iodine, astatine} Exited.</pre> <p>Program berjalan normal menampilkan isi himpunan (set) halogen</p>
2	<p>Tambahkan kode program berikut, lalu coba eksekusi (Run) kode Anda.</p> <pre>1 void main() { 2 var names1 = <String>{}; 3 Set<String> names2 = {}; 4 var names3 ={}; 5 6 print(names1); 7 print(names2); 8 print(names3); 9 }</pre> <p>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 <code>.add()</code> dan <code>.addAll()</code>. Untuk variabel Map dihapus, nanti kita coba di praktikum selanjutnya.</p> <pre>3 {} Exited.</pre> <p>Hasilnya hanya kurung kurawal saja</p>



Modifikasi:

```
1 void main() {  
2     var names1 = <String>{};  
3     Set<String> names2 = {};  
4  
5  
6     names1.add("Vanessa Cristin Natalia");  
7     names2.add("2341720026");  
8     names2.addAll(["Vanessa Cristin Natalia", "2341720026"]);  
9  
10    print(names1);  
11    print(names2);  
12 }
```

Tidak terjadi error, add digunakan untuk menambahkan datanya satu per satu ke dalam set. Sedangkan addAll bisa langsung lebih dari 1 String

```
{Vanessa Cristin Natalia}  
{2341720026, Vanessa Cristin Natalia}  
  
Exited.
```

Praktikum 3: Eksperimen Tipe Data Maps

Langkah	Jawaban/Deskripsi
1	<p>Ketik atau salin kode program berikut ke dalam fungsi main().</p> <pre>void main() { var gift = { 'first': 'partridge', 'second': 'turtledoves', 'fifth': 1 }; var nobleGases = { 2: 'helium', 10: 'neon', 18: 2, }; print(gift); print(nobleGases); }</pre> <p>Silakan coba eksekusi (Run) kode pada langkah 1 tersebut. Apa yang terjadi? Jelaskan!</p>



	<p>Lalu perbaiki jika terjadi error.</p> <p>Kode tersebut berjalan normal tanpa error karena Dart mengenali struktur data sebagai Map, di mana gift merupakan Map<String, dynamic> dengan key berupa string dan value string/angka, sedangkan nobleGases merupakan Map<int, dynamic> dengan key berupa angka dan value string/angka; saat dijalankan program akan menampilkan isi Map dalam format {key: value}.</p>
2	<p>Tambahkan kode program berikut, lalu coba eksekusi (Run) kode Anda.</p> <pre>var mhs1 = Map<String, String>(); gifts['first'] = 'partridge'; gifts['second'] = 'turtledoves'; gifts['fifth'] = 'golden rings'; var mhs2 = Map<int, String>(); nobleGases[2] = 'helium'; nobleGases[10] = 'neon'; nobleGases[18] = 'argon'; }</pre> <pre>{first: partridge, second: turtledoves, fifth: 1} {2: helium, 10: neon, 18: 2} Exited.</pre> <p>Apa yang terjadi ? Jika terjadi error, silakan perbaiki.</p> <p>Tambahkan elemen nama dan NIM Anda pada tiap variabel di atas (gifts, nobleGases, mhs1, dan mhs2). Dokumentasikan hasilnya dan buat laporannya!</p>



3

```
1 void main() {
2     // Map dengan key bertipe String
3     var gifts = {
4         'first': 'partridge',
5         'second': 'turtledoves',
6         'fifth': 'golden rings',
7     };
8
9     // Map dengan key bertipe int
10    var nobleGases = {2: 'helium', 10: 'neon', 18: 'argon'};
11
12    // Map kosong bertipe String-String
13    var mhs1 = Map<String, String>();
14    mhs1['nama'] = 'Vanessa Cristin Natalia';
15    mhs1['nim'] = '2341720026';
16
17    // Map kosong bertipe Int-String
18    var mhs2 = Map<int, String>();
19    mhs2[1] = 'Vanessa Cristin Natalia';
20    mhs2[2] = '2341720026';
21
22    // Menambahkan nama & NIM juga ke gifts dan nobleGases
23    gifts['nama'] = 'Vanessa Cristin Natalia';
24    gifts['nim'] = '2341720026';
25
26    nobleGases[20] = 'Vanessa Cristin Natalia';
27    nobleGases[21] = '2341720026';
28
29    // Cetak hasil
30    print('gifts: $gifts');
31    print('nobleGases: $nobleGases');
32    print('mhs1: $mhs1');
33    print('mhs2: $mhs2');
34 }
```

```
gifts: {first: partridge, second: turtledoves, fifth: golden rings, nama: Vanessa Cristin Natalia, nim: 2341720026}
nobleGases: {2: helium, 10: neon, 18: argon, 20: Vanessa Cristin Natalia, 21: 2341720026}
mhs1: {nama: Vanessa Cristin Natalia, nim: 2341720026}
mhs2: {1: Vanessa Cristin Natalia, 2: 2341720026}
```

Exited.



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

1.	<p>Ketik atau salin kode program berikut ke dalam fungsi main()</p> <pre>1 void main() { 2 var list = [1, 2, 3]; 3 var list2 = [0, ...list]; 4 print(list1); 5 print(list2); 6 print(list2.length); 7 }</pre> <p>p4.dart:85:9: Error: Undefined name 'list1'. print(list1); ^^^^^</p> <p>Exited (254).</p>
2.	<p>Silakan coba eksekusi (Run) kode pada langkah 1 tersebut. Apa yang terjadi? Jelaskan! Lalu perbaiki jika terjadi error.</p> <p>Jawab: Hasilnya error karena nama variabel salah/ tidak ada</p>



```
1  void main() {  
2      var list1 = [1, 2, 3];  
3      var list2 = [0, ...list1];  
4      print(list1);  
5      print(list2);  
6      print(list2.length);  
7  }
```

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

```
Exited.
```

3.

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

```
list1 = [1, 2, null];  
print(list1);  
var list3 = [0, ...?list1];  
print(list3.length);
```

Jawab:



```
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.length);  
12 }
```

```
[1, 2, 3]
```

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

```
4
```

```
[1, 2, null]
```

```
4
```

```
Exited.
```

Apa yang terjadi ? Jika terjadi error, silakan perbaiki.

Tambahkan variabel list berisi NIM Anda menggunakan Spread Operators.
Dokumentasikan hasilnya dan buat laporannya!



```
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.length);  
12  
13    //Menambah variable list berisi NIM  
14    var nim = [2, 3, 4, 1, 7, 2, 0, 0, 2, 6];  
15    var listNIM = [0, ...nim];  
16    print(listNIM);  
17 }
```

```
[1, 2, 3]  
[0, 1, 2, 3]  
4  
[1, 2, null]  
4  
[0, 2, 3, 4, 1, 7, 2, 0, 0, 2, 6]  
  
Exited.
```

4.

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



```
1 // Menambah kode nav dengan collection if  
2 bool promoActive = true;  
3 var nav = ['Home', ' Furniture', 'Plants', if (promoActive) 'Outlet'];  
4 print(nav);
```

Apa yang terjadi ? Jika terjadi error, silakan perbaiki. Tunjukkan hasilnya jika variabel promoActive ketika true dan false.



```
[0, 1, 2, 3]
4
[1, 2, null]
4
[0, 2, 3, 4, 1, 7, 2, 0, 0, 2, 6]
[Home, Furniture, Plants, Outlet]

Exited.
```

Jika promoActive maka Outlet di print jika false maka tidak di print

```
1 // Menambah kode nav dengan collection if
2 bool promoActive = true;
3 var nav = ['Home', ' Furniture', 'Plants', if (promoActive) 'Outlet'];
4 print(nav);
5
6 promoActive = false;
7 var nav2 = ['Home', ' Furniture', 'Plants', if (promoActive) 'Outlet'];
8 print(nav2);
9 }
```

```
[1, 2, 3]
[0, 1, 2, 3]
4
[1, 2, null]
4
[0, 2, 3, 4, 1, 7, 2, 0, 0, 2, 6]
[Home, Furniture, Plants, Outlet]
[Home, Furniture, Plants]

Exited.
```

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

```
1
2 promoActive = false;
3 var nav2 = ['Home', ' Furniture', 'Plants', if (login case 'Manager') 'Inventory'];
4 print(nav2);
5 }
```

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



variabel login mempunyai kondisi lain.

```
p4.dart:105:51: Error: Undefined name 'login'. p4.dart:105
  var nav2 = ['Home', ' Furniture', 'Plants', if (login case 'Manager') 'Inven
  tory'];
                                         ^^^^^^
Exited (254).
```

Perbaiki

```
1 String login = 'Manager';
2 promoActive = false;
3 var nav2 = ['Home', ' Furniture', 'Plants', if (login == 'Manager') 'Inventory'];
4 print(nav2);
5 }
```

```
[1, 2, 3]
[0, 1, 2, 3]
4
[1, 2, null]
4
[0, 2, 3, 4, 1, 7, 2, 0, 0, 2, 6]
[Home, Furniture, Plants, Outlet]
[Home, Furniture, Plants, Inventory]
Exited.
```

Jika variable memiliki kondisi lain

```
1 String login = 'Manager';
2 var nav2 = ['Home', ' Furniture', 'Plants', if (login == 'Staff') 'Inventory'];
3 print(nav2);
```

```
[Home, Furniture, Plants, Outlet]
[Home, Furniture, Plants]
Exited.
```

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

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

Apa yang terjadi ? Jika terjadi error, silakan perbaiki. Jelaskan manfaat **Collection**



For dan dokumentasikan hasilnya.

Jawab:

Collection Fo berfungsi untuk membuat liat baru secara singkat, dinamis, dan mudah dibaca

```
[Home, Furniture, Plants, Outlet]
[Home, Furniture, Plants]
[#0, #1, #2, #3]

Exited.
```

Praktikum 5: Eksperimen Tipe Data Records

1	<p>Ketik atau salin kode program berikut ke dalam fungsi main().</p> <pre>1 void main() { 2 3 var record = ('first', a: 2, b: true, 'last'); 4 print(record); 5 }</pre> <p>Silakan coba eksekusi (Run) kode pada langkah 1 tersebut. Apa yang terjadi? Jelaskan! Lalu perbaiki jika terjadi error.</p> <pre>(first, last, a: 2, b: true) Exited.</pre> <p>Pada percobaan ini terjadi error karena terdapat kekurangan dalam penulisan kode, yaitu tidak disertakan tanda titik koma (;) setelah perintah print(record). Setelah bagian tersebut diperbaiki, program dapat dijalankan dengan baik pada Dart versi 3 yang sudah mendukung tipe data record. Tipe data record berfungsi untuk menyimpan beberapa nilai dalam satu variabel, yang dapat berupa positional fields maupun named fields. Hasil eksekusi program kemudian menampilkan isi record secara lengkap, yaitu ('first', a: 2, b: true, 'last').</p>
2.	<p>Tambahkan kode program berikut di luar scope <code>void main()</code>, lalu coba eksekusi (Run) kode Anda.</p>



```
1 void main() {  
2     var record = ('first', a: 2, b: true, 'last');  
3     print(record);  
4 }  
5  
6 // Langkah 3  
7 (int, int) tukar((int, int) record) {  
8     var (a, b) = record;  
9     return (b, a);  
10 }
```

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

Jawab:

Pada percobaan ini, program dapat dijalankan dengan sukses tanpa menimbulkan error dan menghasilkan output (first, last, a: 2, b: true). Output tersebut berasal dari record yang memuat kombinasi antara positional field ('first' dan 'last') serta named field (a: 2 dan b: true). Ketika dijalankan, Dart secara otomatis menampilkan seluruh isi record sesuai urutan—dimulai dari nilai posisi, kemudian diikuti nilai dengan nama. Hasil ini menunjukkan bahwa record dalam Dart mampu menyimpan berbagai jenis data secara ringkas dan terstruktur di dalam satu variabel.

Perbaiki:



```
1 (int, int) tukar((int, int) record) {  
2     var (a, b) = record;  
3     return (b, a);  
4 }  
5  
6  
7  
8 void main() {  
9     // Langkah 1  
10    var record = ('first', a: 2, b: true, 'last');  
11    print('Record dari Langkah 1: $record');  
12  
13    // Contoh record dengan angka untuk ditukar  
14    var angka = (10, 20);  
15    print('Record awal: $angka');  
16  
17    // Gunakan fungsi tukar  
18    var swapped = tukar(angka);  
19    print('Record setelah ditukar: $swapped');  
20 }
```

```
Record dari Langkah 1: (first, last, a: 2, b: true)  
Record awal: (10, 20)  
Record setelah ditukar: (20, 10)  
  
Exited.
```

Pada percobaan ini, kode pada Langkah 1 diperluas dengan menambahkan fungsi tukar() pada Langkah 3. Fungsi tersebut menerima record bertipe (int, int) dan menukar posisi nilainya dengan memanfaatkan destrukturisasi. Setelah dijalankan, record (10, 20) berhasil diubah menjadi (20, 10). Hal ini menunjukkan bahwa record dalam Dart memiliki sifat yang dinamis dan fleksibel, karena selain mampu menyimpan beragam tipe data dalam satu variabel, juga dapat dimodifikasi atau dimanipulasi melalui fungsi.

3. Tambahkan kode program berikut di dalam scope `void main()`, lalu coba eksekusi (Run) kode Anda.

```
1 // Langkah 4  
2 // Record type annotation in a variable declaration:  
3 (String, int) mahasiswa;  
4 print(mahasiswa);  
5 }
```



```
p4.dart:145:9: Error: Non-nullable variable 'mahasiswa' must be assigned before it can be used.
print(mahasiswa);
          ^^^^^^

Exited (254).
```

Perbaikan:

```
1 // Langkah 4
2 // Record type annotation in a variable declaration:
3 (String, int) mahasiswa = ('Vanessa Cristin Natalia', 2341720026);
4 print(mahasiswa);
5 print('Nama: ${mahasiswa.$1}');
6 print('NIM: ${mahasiswa.$2}');
7 }
```

Setelah dilakukan perbaikan dengan menginisialisasi record bertipe (String, int) menggunakan data nama dan NIM, program dapat dijalankan dengan baik dan menghasilkan output. Setiap nilai pada record tersebut bisa diakses secara langsung menggunakan notasi posisi, misalnya mahasiswa.\$1 untuk mengambil nama dan mahasiswa.\$2 untuk mengambil NIM.

4. Tambahkan kode program berikut di dalam scope `void main()`, lalu coba eksekusi (Run) kode Anda.

```
1
2 // Langkah 5
3 var mahasiswa2 = ('first', a: 2, b: true, 'last');
4
5 print(mahasiswa2.$1); // Prints 'first'
6 print(mahasiswa2.a); // Prints 2
7 print(mahasiswa2.b); // Prints true
8 print(mahasiswa2.$2); // Prints 'last'
```

Apa yang terjadi ? Jika terjadi error, silakan perbaiki.

```
first
2
true
last

Exited.
```

Kode tersebut dapat dijalankan tanpa menimbulkan error karena record dalam Dart mendukung dua jenis field, yaitu positional fields (\$1, \$2, dan seterusnya) serta named fields (a, b, dan sebagainya). Saat dijalankan, program akan menampilkan isi record sesuai dengan field yang diakses pada masing-masing bagian.



Gantilah salah satu isi record dengan nama dan NIM Anda, lalu dokumentasikan hasilnya dan buat laporannya!

```
1 // Langkah 5
2 var mahasiswa2 = ('Vanessa Cristin Natalia', a: 2341720026, b: true, 'last');
3
4 print(mahasiswa2.$1); // Prints 'Vanessa Cristin Natalia'
5 print(mahasiswa2.a); // Prints 2341720026
6 print(mahasiswa2.b); // Prints true
7 print(mahasiswa2.$2); // Prints 'Last'
8 }
```

Pada percobaan ini, program tidak error. Record mahasiswa2 berisi gabungan antara positional field ('Vanessa Cristin Natalia', 'last') dan named field (a: 2341720026, b: true). Nilai pada record dapat diakses menggunakan notasi \$1 dan \$2 untuk field posisi, serta .a dan .b untuk field bernama. Setelah dilakukan perubahan dengan menambahkan data nama dan NIM, program berhasil menampilkan informasi mahasiswa dengan benar.

TUGAS PRAKTIKUM

1. Jelaskan yang dimaksud Functions dalam bahasa Dart!

Jawab:

Function dalam Dart adalah sekumpulan kode yang terstruktur dan dapat dipakai kembali untuk menyelesaikan suatu pekerjaan tertentu. Fungsi berperan dalam memecah program menjadi komponen-komponen yang lebih kecil dan terstruktur, sehingga lebih mudah untuk dikelola, dibaca, dan diperbaiki. Setiap aplikasi Dart harus memiliki fungsi main(), yang merupakan titik awal untuk menjalankan program. Contoh:

```
1 // Tugas Praktikum - No 2
2 String salam(String nama) {
3     return 'Halo, $nama!';
4 }
5
6 void main() {
7     var pesan = salam('Vanessa');
8     print(pesan); // Output: Halo, Vanessa!
9 }
```

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

Jawab:

Dart mendukung dua jenis parameter utama dalam fungsi: 1. Positional Parameters (Parameter Posisi) Parameter yang harus diberikan dalam urutan yang sama seperti yang didefinisikan dalam fungsi. Parameter ini wajib diisi kecuali ditandai sebagai opsional. Parameter Posisi Wajib: Parameter biasa yang didefinisikan dalam tanda kurung (). Contoh:



```
1 void main() {  
2     print(tambah(3, 4)); // 7  
3     print(sapaOpsional('Vanessa')); // Halo, Vanessa  
4     print(sapaOpsional('Vanessa', 'Vaness')); // Halo, Vaness (Vanessa)  
5     print(profil(nama: 'Vanessa', umur: 20)); // Vanessa, 20 tahun, jurusan Teknik  
6 }
```

7

Halo, Vanessa

Halo, Vaness (Vanessa)

Vanessa, 20 tahun, jurusan Teknik

Exited.

- tambah → required positional (wajib diisi).
 - sapaOpsional → optional positional [] dan memiliki default value.
 - profil → named parameters {}; required memaksa caller menyertakan nama.
3. Jelaskan maksud Functions sebagai first-class objects beserta contoh sintaknya! Apa itu Anonymous Functions?

Jawab:

a. First-class functions

Maksud: Fungsi diperlakukan seperti variabel biasa (misalnya, int atau String).

Implikasi: Dapat disimpan dalam variabel. Dapat diteruskan sebagai argumen ke fungsi lain (callback). Dapat dikembalikan sebagai nilai dari fungsi lain.



```
1 // Tugas Praktikum - No 4
2
3 int kuadrat(int x) => x * x;
4
5 void applyAndPrint(int Function(int) f, int value) {
6     var result = f(value);
7     print('Result: $result');
8 }
9
10 void main() {
11     // simpan fungsi ke variabel
12     var op = kuadrat;
13     print(op(5)); // 25
14
15     // kirim fungsi sebagai argumen
16     applyAndPrint(kuadrat, 7); // Result: 49
17
18     // fungsi juga dapat dikembalikan dari fungsi lain
19     int Function(int) buatPengali(int multiplier) {
20         return (int x) => x * multiplier;
21     }
22
23     var kali3 = buatPengali(3);
24     print(kali3(4)); // 12
25 }
26
27
```

```
25
Result: 49
12
Exited.
```

b. Anonymous Functions

Maksud: Fungsi yang tidak memiliki nama.

Penggunaan: Didefinisikan secara inline (langsung) dan sering digunakan sebagai callback atau argumen untuk metode seperti `forEach` atau `map`.



```
1 // Tugas Praktikum - No 5
2 // praktikum3_anonymous.dart
3 void main() {
4     var list = [1, 2, 3, 4];
5     // anonymous function sebagai parameter ke map
6     var doubled = list.map((n) => n * 2).toList();
7     print(doubled); // [2,4,6,8]
8
9     // fungsi anonim multiple-line
10    var filtered = list.where((n) {
11        var ok = n % 2 == 0;
12        return ok;
13    }).toList();
14    print(filtered); // [2,4]
15 }
16
17
18
```

[2, 4, 6, 8]

[2, 4]

Exited.

4. Jelaskan dan berikan contohnya! Jelaskan perbedaan Lexical scope dan Lexical closures!
Berikan contohnya!

Jawab:

- Penjelasan ringkas: Lexical scope: aturan penentuan variable yang dapat diakses berdasarkan struktur posisi kode (scope statis). Variabel yang dideklarasikan di luar fungsi dapat diakses oleh fungsi di dalamnya jika tidak ditimpa.
- Closure: fungsi yang “menangkap” (mengikat) variabel dari lingkungan (lexical environment) saat fungsi dibuat — sehingga variabel tersebut tetap ada walau scope aslinya selesai.



```
1 // Tugas Praktikum - No 6
2 Function buatCounter() {
3   int count = 0; // lexical environment yang ditangkap
4   return () {
5     count += 1;
6     return count;
7   };
8 }
9
10 void main() {
11   var counter1 = buatCounter();
12   print(counter1()); // 1
13   print(counter1()); // 2
14
15   var counter2 = buatCounter();
16   print(counter2()); // 1 (counter2 punya environment sendiri)
17 }
```

1

2

1

Exited.

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

Jawab:

a) Menggunakan Record

```
1 // Tugas Praktikum - No 7
2 // Menggunakan Record
3 // praktikum5_record.dart
4 (String, int) ambilMahasiswa() {
5   return ('Vanessa Cristin Natalia', 2341720026);
6 }
7
8 void main() {
9   var m = ambilMahasiswa();
10  print(m);           // (Vanessa Cristin Natalia, 2341720026)
11  print(m.$1);        // Vanessa Cristin Natalia
12  print(m.$2);        // 2341720026
13 }
```



```
(Vanessa Cristin Natalia, 2341720026)
Vanessa Cristin Natalia
2341720026

Exited.
```

b) Menggunakan Map

```
1 Map<String, dynamic> ambilMahasiswaMap() {
2     return {'nama': 'Vanessa Cristin Natalia', 'nim': 2341720026};
3 }
4 void main() {
5     var m = ambilMahasiswaMap();
6     print('${m['nama']}, ${m['nim']}');
7 }
```

```
Vanessa Cristin Natalia, 2341720026

Exited.
```

c) Menggunakan class (lebih rapi & terstruktur)

```
1 class Mahasiswa {
2     final String nama;
3     final int nim;
4     Mahasiswa(this.nama, this.nim);
5 }
6
7 Mahasiswa ambilMahasiswaClass() => Mahasiswa('Vanessa Cristin Natalia', 2341720026);
8
9 void main() {
10     var m = ambilMahasiswaClass();
11     print('${m.nama}, ${m.nim}');
12 }
```

```
Vanessa Cristin Natalia, 2341720026

Exited.
```