

LAPORAN PRAKTIKUM PENGEMBANGAN APLIKASI BERGERAK

Pengenalan Kotlin - WEEK 1



Disusun oleh:

Nama : Alfath Roziq Widhayaka

Nim : L0122012

Kelas : A

PROGRAM STUDI INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI DAN SAINS DATA

UNIVERSITAS SEBELAS MARET

2024

1. Screenshot Source Code

Source code ini merupakan source code yang digunakan untuk praktikum minggu 1 saya. Pengerjaan praktikum ini melibatkan penerapan materi minggu 1, yaitu pengenalan fundamental Kotlin. Di dalam materi ini terdapat poin-poin yang harus diterapkan seperti dijelaskan di bawah ini :

A. Data Types & Variables

```
1 fun main() {
2     //Deklarasi variabel mobil
3     val carModel: String = "Silvia S15"
4     var carTahun: Int = 2010
5     val carWarna: String = "Putih"
6     val carTransmisi: String? = "Manual"
7     val carKondisi: String? = null
8
9     //Named Argument
10    val totalHarga = hitungTotalHarga(hargaAwal = 300000000)
11
12    val namaCustomer: String? = tampilkanCustomer()
13
14    //Periksa apakah nama customer tidak null, lalu tampilkan pesan selamat datang
15    if (namaCustomer != null) {
16        println("\nSelamat datang, $namaCustomer di dealer Nissan!")
17    } else {
18        println("Selamat datang di dealer Nissan!")
19    }
20
21    displayMobilDijual()
22    displayMobil(carModel, carTahun, carWarna, carTransmisi, carKondisi)
23    displayMobilDibeli(totalHarga)
24
25    //Raw String
26    val terimakasih = """
27        Terima kasih atas pembelian mobil di dealer Nissan.
28        Kami mengucapkan terima kasih atas kepercayaan Anda.
29        Semoga Anda puas dengan mobil baru Anda!
30    """.trimIndent()
31
32    println(terimakasih)
33 }
```

```
1 fun hitungTotalHarga(hargaAwal: Int, pajak: Double = 0.05): Int {
```

Potongan kode diatas merupakan implementasi penggunaan data types dan variables. Deklarasi variabel yang digunakan diatas sebagian besar adalah **val** seperti "**val carModel: String = "Silvia S15"**" dan ada salah satu yang menggunakan **var** yaitu "**var carTahun: Int = 2010**". Untuk tipe data yang digunakan pada program ini adalah **String**, **Int**, dan **Double**.

B. Functions (selain main function)



Materi functions pada kode di atas melibatkan beberapa fungsi (functions) yang memberikan fungsionalitas tertentu dalam program. Berikut adalah penjelasan singkat untuk masing-masing fungsi.

1. **fun hitungTotalHarga**

Fungsi ini digunakan untuk menghitung total harga mobil dengan menambahkan pajak ke harga awal. Fungsi ini menerima dua parameter, yaitu `hargaAwal` (tipe data `Int`) dan `pajak` (tipe data `Double` dengan nilai default `0.05`).

2. **fun tampilkanCustomer**

Fungsi ini mengembalikan nilai berupa string, yaitu nama customer ("Alfath"). Selain itu dapat mengembalikan nilai `null` jika tidak ada informasi customer.

3. **fun displayMobilDijual**

Fungsi ini menampilkan daftar mobil yang terjual di dealer Nissan dengan mencetak informasi beberapa mobil ke terminal.

4. **fun displayMobil**

Fungsi ini menampilkan detail mobil yang dibeli ke terminal, termasuk model, tahun, warna, transmisi, dan kondisi.

5. **fun displayMobilDibeli**

Fungsi ini menampilkan total harga pembelian mobil ke terminal dengan harga yang telah ditambah dengan pajak

C. Nullable Types

```
1 val carTransmisi: String? = "Manual"
2 val carKondisi: String? = null
```

Pada bagian deklarasi variabel, terdapat dua variabel yang memiliki tipe nullable, yaitu **carTransmisi** dan **carKondisi**. Variabel dengan tipe nullable dinyatakan dengan tanda tanya (?) setelah tipe datanya. Ini berarti variabel tersebut dapat memiliki nilai null.

```
1 //Periksa apakah nama customer tidak null, lalu tampilkan pesan selamat datang
2 if (namaCustomer != null) {
3     println("\nSelamat datang, $namaCustomer di dealer Nissan!")
4 } else {
5     println("Selamat datang di dealer Nissan!")
6 }
```

Nullable types diatas menggunakan pernyataan if untuk memeriksa apakah namaCustomer tidak null sebelum mencetak pesan selamat datang.

D. Safe Calls

```
1 val carTransmisi: String? = "Manual"
2 val carKondisi: String? = null
```

```
1 println("Kondisi      : ${kondisi?.toString() ?: "Informasi tidak tersedia"}")
```

Pada baris ini, **kondisi?.toString() ?: "Informasi tidak tersedia"** menggunakan safe call (?) pada **kondisi**. Jika **kondisi** adalah null, maka ekspresi ini akan menghasilkan null, dan elvis operator memberikan nilai default.

E. If Expression

```
1 //Periksa apakah nama customer tidak null, lalu tampilkan pesan selamat datang
2 if (namaCustomer != null) {
3     println("\nSelamat datang, $namaCustomer di dealer Nissan!")
4 } else {
5     println("Selamat datang di dealer Nissan!")
6 }
```

If expression pada kondisi ini memeriksa apakah **variabel namaCustomer** tidak null. Jika kondisinya benar (namaCustomer tidak null), maka nama dari customer yang membeli akan tercetak pada terminal. Jika kondisinya salah (namaCustomer null), maka akan mencetak pesan selamat datang tanpa menggunakan nama customer yang akan membeli.

F. Named Argument

```
1 val totalHarga = hitungTotalHarga(hargaAwal = 300000000)
```

Di sini, hargaAwal adalah parameter yang diberi nilai 300000000. Penggunaan **named argument** terlihat saat memanggil **fungsi hitungTotalHarga** dengan menyertakan nama parameter (hargaAwal = 300000000), yang membantu memperjelas argumen yang diberikan kepada fungsi tersebut.

G. Default Argument

```
1 //Fungsi untuk menghitung total harga mobil dengan pajak
2 fun hitungTotalHarga(hargaAwal: Int, pajak: Double = 0.05): Int {
3     val totalHarga = hargaAwal + (hargaAwal * pajak).toInt()
4     return totalHarga
5 }
```

Dalam fungsi ini, parameter pajak memiliki nilai default 0.05 sebagai **default argument**. Jika nilai untuk pajak tidak disediakan saat memanggil fungsi, maka nilai default 0.05 akan digunakan.

```
1 fun displayMobil(model: String, tahun: Int, warna: String, transmisi: String?, kondisi: String?)
```

Pada fungsi ini, parameter transmisi dan kondisi memiliki nilai default null. Jika nilai untuk parameter ini tidak disediakan saat memanggil fungsi, nilai default null akan digunakan.

H. Raw String

```
1 //Raw String
2 val terimakasih = """
3     Terima kasih atas pembelian mobil di dealer Nissan.
4     Kami mengucapkan terima kasih atas kepercayaan Anda.
5     Semoga Anda puas dengan mobil baru Anda!
6 """.trimIndent()
```

Pada kode yang diberikan, raw string digunakan pada **variabel terimakasih** berisi pesan terima kasih kepada pelanggan dan diformat sebagai string mentah multi-baris untuk meningkatkan keterbacaan. **trimIndent()** adalah fungsi pada string untuk menghapus spasi indentasi umum dari setiap baris.

I. String Template

```
1 //Fungsi untuk menampilkan detail mobil yang dibeli
2 fun displayMobil(model: String, tahun: Int, warna: String, transmisi: String?, kondisi: String?) {
3     println("\nDetail mobil yang dibeli:")
4     println("Model      : $model")
5     println("Tahun       : $tahun")
6     println("Warna        : $warna")
7     println("Transmisi    : ${transmisi ?: "Informasi tidak tersedia"}")
8     println("Kondisi     : ${kondisi?.toString() ?: "Informasi tidak tersedia"}")
9 }
```

String template pada kode diatas ditandai dengan penggunaan karakter dollar "\$" sebelum nama variabel yang ingin dimasukkan ke dalam string pada **fungsi displayMobil**. Kode juga menyertakan ekspresi yang lebih kompleks yaitu menggunakan kurung kurawal "\${}" untuk membatasi ekspresi. Di sini, nilai dari variabel seperti **model**, **tahun**, **warna**, **transmisi**, dan **kondisi** dimasukkan langsung ke dalam string melalui string template.

J. Elvis Operators

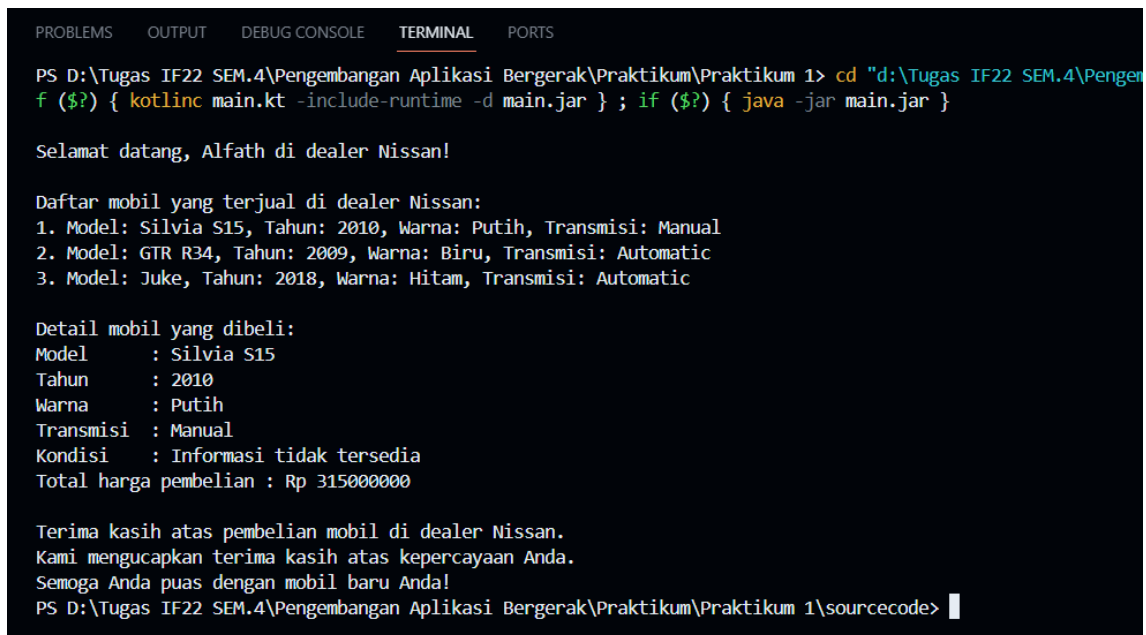
```
1 val carTransmisi: String? = "Manual"
2 val carKondisi: String? = null
```

```
1 println("Transmisi : ${transmisi ?: "Informasi tidak tersedia"}")
2 println("Kondisi   : ${kondisi?.toString() ?: "Informasi tidak tersedia"}")
```

Di sini, elvis operator digunakan untuk menentukan nilai yang akan dicetak. Jika nilai transmisi atau kondisi adalah null, maka akan mencetak "Informasi tidak tersedia".

2. Screenshot Terminal

Berikut adalah hasil pada terminal apabila source code dijalankan. Gambar terminal dibawah ini merupakan hasil setelah kode ini menerapkan beberapa materi diatas



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\Tugas IF22 SEM.4\Pengembangan Aplikasi Bergerak\Praktikum\Praktikum 1> cd "d:\Tugas IF22 SEM.4\Pengembangan Aplikasi Bergerak\Praktikum 1\sourcecode" & .\main.jar
f ($?) { kotlinc main.kt -include-runtime -d main.jar } ; if ($?) { java -jar main.jar }

Selamat datang, Alfath di dealer Nissan!

Daftar mobil yang terjual di dealer Nissan:
1. Model: Silvia S15, Tahun: 2010, Warna: Putih, Transmisi: Manual
2. Model: GTR R34, Tahun: 2009, Warna: Biru, Transmisi: Automatic
3. Model: Juke, Tahun: 2018, Warna: Hitam, Transmisi: Automatic

Detail mobil yang dibeli:
Model      : Silvia S15
Tahun      : 2010
Warna      : Putih
Transmisi  : Manual
Kondisi    : Informasi tidak tersedia
Total harga pembelian : Rp 315000000

Terima kasih atas pembelian mobil di dealer Nissan.
Kami mengucapkan terima kasih atas kepercayaan Anda.
Semoga Anda puas dengan mobil baru Anda!
PS D:\Tugas IF22 SEM.4\Pengembangan Aplikasi Bergerak\Praktikum\Praktikum 1\sourcecode>
```

- Tulisan **Selamat datang, Alfath di dealer Nissan!** Ini merupakan penerapan dari **data type & variables, if expression**, dan **Nullable Types**. Nama customer akan muncul jika kondisinya benar (tidak null).
- Daftar mobil yang terjual diatas merupakan pemanggilan fungsi dari **fun displayMobilDijual()**, selanjutnya fungsi tersebut dipanggil di **fun main**.
- Detail mobil sampai harga mobil merupakan gabungan dari **fun displayMobil** dan **fun hitungTotalHarga**. Materi yang digunakan disini adalah **string template, safe calls** pada **kondisi, elvis operators** pada **transmisi** dan **kondisi, default argument** sebagai nilai pajak yang default ketika dipanggil, **named argument** sebagai deklarasi parameter hargaAwal yang diberi nilai 300000000.
- Untuk kata Terimakasih sampai selesai merupakan bagian dari **Raw String** untuk menampilkan baris multiple sebagai penutup dari program.