



NIM

11S21014

Nama

Dedi Andre Martua Raja Panggabean

Tanggal dibuat

02 February 2024, 11:29

score

100

Praktikum 1 (100%)

100 / 100

Latihan Praktikum

25 / 25

A. Dasar Kotlin

100 / 100

1. Variable

100 / 100

Catatan

1. tipe variable Konstanta `val` berfungsi untuk variable yang artinya constanta tersebut valuenya tidak dapat diubah sejak di inialisasi sejak pertama kali.dalam potongan kode tersebut telah di set nilai `PI = 3.14` dan tidak dapat diubah lagi
2. Dengan menggunakan `val`, Anda memberi tahu Kotlin bahwa nilai dari variabel `pi` tidak akan berubah setelah pertama kali diinisialisasi. Dalam hal ini, nilai `pi` diatur sebagai `3.14`, dan nilai ini tidak dapat diubah selama siklus hidup program.
3. variable mutable artinya variable yang nilai nya dapat berubah,pada cotlin menggunakan variable `var` yang artinya variable tersebut dapat berubah
4. tipe data otomatis dimana `val name = "Abdullah"` telah di set dan tidak dapat berubah
5. nullable types dimana `var think: String? = null` adalah variable dengan value `null`.dari potongan code,nilai `think` dapat berubah-ubah.`String?` merupakan pendeklarasian untuk bernilai `null` yang ditandai dengan sintaks `?`.

2. Tipe Data

100 / 100

Catatan

1. Tipe data `Int` digunakan untuk mendeklarasikan tipe data bilangan bulat
2. `Double` digunakan untuk mendeklarasikan tipe data yang merupakan bilangan desimal
3. `Char` berfungsi untuk deklarasi variable dengan tipe character
4. `Boolean` untuk conditional `true` or `false`
5. `String` untuk tipe data `String` yaitu jumlah karakter lebih dari 1
6. `Array` untuk manampung banyak dengan tipe data yang sama
7. `Nullable(?)` untuk menampung variable dengan tipe `null`
8. `Any` adalah tipe data yang mewakili segala jenis objek. Semua tipe data non-primitif di Kotlin secara tidak langsung menggantikan tipe data `Any`. Dengan menggunakan `Any`, Anda dapat menyimpan objek apa pun, termasuk objek dengan tipe data yang berbeda-beda.

Catatan

Kode tersebut adalah contoh penggunaan string dalam bahasa pemrograman Kotlin. Berikut adalah rangkumannya:

1. **Deklarasi String:**

- `nama1` dan `nama2` dideklarasikan sebagai string dengan nilai "Abdullah" dan "Ubaid" secara eksplisit dan implisit.

2. **Operasi String:**

- `hasil1` merupakan hasil penggabungan string menggunakan operator `+`.
- `hasil2` menggunakan interpolasi string dengan `\$nama1 \$nama2`.
- `hasil3` melakukan manipulasi string dengan menambahkan teks tambahan.

3. **Fungsi pada String:**

- `teks.length` untuk mendapatkan panjang string.
- `teks.lowercase()` mengonversi string ke huruf kecil.
- `teks.uppercase()` mengonversi string ke huruf besar.
- `teks.contains("Kotlin")` untuk memeriksa apakah string mengandung substring "Kotlin".
- `teks.repeat(3)` mengulangi string sebanyak 3 kali.

4. **String Raw:**

- `teksMultiline` adalah contoh string multiline yang ditandai dengan triple double-quotes (`"""`). Ini memungkinkan penggunaan multiple lines tanpa menggunakan karakter escape.

5. **Pencetakan Hasil:**

- Menggunakan `println` untuk mencetak hasil operasi dan fungsi pada string.

Rangkuman ini mencakup aspek-aspek kunci dari kode Kotlin yang diberikan untuk manipulasi dan operasi pada string.

4. Data Collection

Catatan

Berikut adalah beberapa catatan dari kode Kotlin yang Anda berikan:

1. List:

- `val angka: List = listOf(1, 2, 3, 1, 2, 3)` membuat list tidak dapat diubah dengan elemen integer.
- `angka.forEach { print("${it} | ") }` digunakan untuk mencetak setiap elemen list dengan pemisah |.

2. Mutable List:

- `val mutableList: MutableList = mutableListOf(1, 2, 3, 1, 2, 3)` membuat list yang dapat diubah.
- `mutableList.add(4)` menambahkan elemen baru ke dalam list.
- `mutableList.forEach { print("${it} | ") }` mencetak semua elemen di dalam list setelah penambahan.

3. Set:

- `val huruf: Set = setOf('a', 'b', 'c', 'a', 'b', 'c')` membuat set dengan elemen karakter.
- `huruf.forEach { print("${it} | ") }` mencetak setiap elemen set dengan pemisah |.

4. Mutable Set:

- `val mutableSet: MutableSet = mutableSetOf("apple", "banana", "orange", "apple", "banana", "orange")` membuat set yang dapat diubah dengan elemen string.
- `mutableSet.add("grape")` menambahkan elemen baru ke dalam set.
- `mutableSet.forEach { print("${it} | ") }` mencetak semua elemen di dalam set setelah penambahan.

5. Map:

- `val map: Map = mapOf("satu" to 1, "dua" to 2, "tiga" to 3)` membuat map tidak dapat diubah.

- map.forEach { (kunci, nilai) -> print("\$kunci -> \$nilai | ") } mencetak pasangan kunci dan nilai dari map.

6. Mutable Map:

- val mutableMap: MutableMap = mutableMapOf("one" to 1, "two" to 2) membuat map yang dapat diubah.
- mutableMap["three"] = 3 menambahkan pasangan kunci dan nilai baru ke dalam map.

7. Array:

- val numbers: Array = arrayOf(1, 2, 3, 4, 5) membuat array dengan elemen integer.
- numbers.forEach { print("\${it} | ") } mencetak setiap elemen array dengan pemisah |.

8. Sequence:

- val sequence: Sequence = sequenceOf(1, 2, 3, 4, 5) membuat sequence dengan elemen integer.
- sequence.forEach { print("\${it} | ") } mencetak setiap elemen sequence dengan pemisah |.

Setiap bagian kode memberikan contoh penggunaan tipe data koleksi yang berbeda-beda di Kotlin.

5. Data List

100 / 100

Catatan

Berikut adalah catatan dari kode Kotlin yang Anda berikan:

1. List:

- val angka: List = listOf(1, 2, 3, 4, 5) membuat list dengan tipe data Int.
- val kata: List = listOf("satu", "dua", "tiga") membuat list dengan tipe data String.
- val campuran: List = listOf(1, "dua", 3.0, true) membuat list dengan tipe data campuran (Any).
- val listNullable: List? = null membuat list yang dapat bernilai null.

2. Iterasi dan Cetak List:

- angka.forEach { print("\${it} | ") } mencetak setiap elemen dari list angka dengan pemisah |.
- kata.forEach { print("\${it} | ") } mencetak setiap elemen dari list kata dengan pemisah |.
- campuran.forEach { print("\${it} | ") } mencetak setiap elemen dari list campuran dengan pemisah |.
- println(listNullable) mencetak nilai list yang mungkin null.

3. List Mutable:

- val mutableList: MutableList = mutableListOf(1, 2, 3) membuat list yang dapat diubah.
- mutableList.add(4) menambahkan elemen baru ke dalam list.
- mutableList.removeAt(0) menghapus elemen pertama dari list.
- mutableList.forEach { print("\${it} | ") } mencetak semua elemen di dalam list setelah operasi penambahan dan penghapusan.

4. Mengakses Elemen dalam List:

- val elemenPertama: Int = angka[0] mengakses elemen pertama dari list angka.
- val elementTerakhir: Int = angka[angka.size - 1] mengakses elemen terakhir dari list angka.
- println(elemenPertama) dan println(elementTerakhir) mencetak nilai elemen pertama dan terakhir.

5. Operasi List:

- val jumlahElemen: Int = angka.size menghitung jumlah elemen dalam list angka.
- val isEmpty: Boolean = angka.isEmpty() mengecek apakah list angka kosong.
- val kaliDua: List = angka.map { it * 2 } membuat list baru dengan melakukan transformasi (penggandaan) pada setiap elemen list angka.
- println(jumlahElemen), println(isEmpty), dan kaliDua.forEach { print("\${it} | ") } mencetak hasil dari operasi-operasi tersebut.

1. Input & Output

100 / 100

Test Case	Status	Score
TC1	PASS	50 / 50
TC2	PASS	50 / 50
Total Score		100 / 100