

MODUL 1

DASAR PEMROGRAMAN KOTLIN, FLOW CONTROL



CAPAIAN PEMBELAJARAN

1. Mahasiswa mampu mengimplementasikan dasar-dasar pemrograman dengan Kotlin
2. Mahasiswa mampu menggunakan variabel dengan berbagai tipe data operasinya dalam program
3. Mahasiswa mampu menggunakan ungkapan kontrol aliran/pencabangan program
4. Mahasiswa mampu menggunakan bentuk perulangan (*looping*) dalam program



KEBUTUHAN ALAT/BAHAN/SOFTWARE

1. <https://play.kotlinlang.org/> untuk menjalankan program



DASAR TEORI

Kotlin adalah bahasa pemrograman yang diketik secara statis, dikembangkan oleh JetBrains. Jika Anda memiliki pengetahuan dasar tentang Java, Anda akan dapat mempelajari Kotlin dalam waktu singkat.

Kotlin dan Java adalah *interoperable* yang berarti Anda dapat menggunakannya bersama-sama dalam suatu proyek serta Anda dapat menulis ulang kode Java di Kotlin secara efisien. Sintaksis Kotlin lebih ringkas dari Java.

Fitur Kotlin

Ringkas: Kotlin lebih ringkas daripada Java, Anda dapat menulis sekitar 40% lebih sedikit baris kode dibandingkan dengan Java.

Interoperabilitas: Kotlin sangat *interoperable* dengan Java. Anda tidak akan menghadapi kesulitan menggunakan Kotlin dalam proyek Java.

Open Source: Kotlin adalah bahasa pemrograman *open source*.

Kepercayaan (Trust): Anda dapat mempercayai Kotlin karena ini dikembangkan oleh perusahaan JetBrains yang populer dan terkenal. JetBrains dikenal karena membuat beberapa alat pengembangan. Java IDE IntelliJ IDEA yang populer dikembangkan oleh perusahaan yang sama ini.

Kaya fitur: Kotlin menyediakan beberapa fitur canggih seperti overloading operator, ekspresi Lambda, template string dll.

Mudah: Kotlin adalah bahasa pemrograman yang mudah dipelajari. Jika Anda berasal dari latar belakang Java, Anda akan mudah mempelajari Kotlin.

Kurang rawan kesalahan: Kotlin adalah bahasa pemrograman yang diketik secara statis, yang membuat Anda dapat menangkap kesalahan pada waktu kompilasi karena bahasa pemrograman yang diketik secara statis melakukan pengecekan ketik pada waktu kompilasi.

Contoh Program Kotlin

```
// Tampilkan Hello World! pada layar
fun main(args : Array<String>) {
    println("Hello World!")
}
```

Output:

```
Hello World!
```

1. Baris pertama program adalah:

```
// Tampilkan Hello World! pada layar
```

Ini adalah komentar. Anda dapat menulis apa pun di sini, kompiler mengabaikan komentar ini saat menjalankan program.

2. Baris kedua dari program ini adalah:

```
fun main(args : Array<String>) { }
```

Ini adalah fungsi main. Mirip dengan Java, pelaksanaan program Kotlin dimulai dari fungsi ini. Fungsi ini adalah titik awal dari program Kotlin.

3. Baris ketiga dari program ini adalah:

```
println("Hello World!")
```

Ini mirip dengan pernyataan `System.out.println ("Hello World!")` di Java. Tujuan pernyataan ini untuk menampilkan pesan di dalam tanda kutip ganda di layar.

Variabel dan Tipe Data Kotlin

Ada dua jenis variabel - bisa berubah (mutable) dan tidak berubah (immutable). Variabel tidak berubah adalah variabel yang nilainya tidak dapat diubah, juga dikenal sebagai variabel hanya-baca. Di sisi lain, nilai variabel yang dapat diubah dapat diubah selama program berjalan.

Variabel yang tidak dapat diubah dinyatakan menggunakan kata kunci **val**.

```
val namaMhs = "Susi"  
println("Nama saya adalah: "+namaMhs)
```

Di kotlin, kita menggunakan kata kunci **var** untuk mendeklarasikan variabel yang dapat diubah.

```
val namaMhs = "Susi"  
namaMhs = "Susi Susanti"  
println("Nama saya adalah: "+namaMhs)
```

Jika Anda ingin menyebutkan tipe variabel secara eksplisit, Anda dapat melakukannya seperti ini:

```
var website: String = "beginnersbook"
```

Di sini kita telah secara eksplisit menyebutkan tipe variabel "website" sebagai String.

Tipe Data Kotlin

1. Bilangan – Byte, Short, Int, Long, Float, Double
2. Boolean – True, false
3. Characters
4. Arrays
5. Strings

Operator Aritmetika Operators

1. + → Addition Operator
2. - → Subtraction Operator
3. * → Multiplication Operator
4. / → Division Operator
5. % → Modulus Operator

Array

Array dapat menampung beberapa tipe data yang berbeda.

```
var arr = arrayOf(10, "Susi Susanti", 10.99, 'A')
```

Array yang hanya bisa menampung bilangan bulat:

```
var arr = arrayOf<Int>(1, 22, 55)
```

Kita dapat menampilkan elemen ke-3 dari array menggunakan indeks (`arr [2]`). Kita juga dapat memperoleh elemen array dengan fungsi `get()`. Sebagai contoh `arr[2]` dapat diganti dengan `arr.get(2)`.

Sementara itu untuk mengubah elemen array dapat menggunakan fungsi `set()`. Contoh:

```
array[2] = 100
```

dapat ditulis ulang dengan:

```
arr.set(2, 100)
```

Aliran Kontrol

Dalam Kotlin terdapat sejumlah ungkapan untuk aliran kontrol.

1. Ungkapan if
2. Ungkapan if..else
3. Ungkapan if..else if..else
4. Ungkapan bersarang (nested)

Contoh ungkapan if:

```
val number = 100

// if expression
if (number%2 == 0)
    println("Bilangan genap")

println("Di luar statement If")
```

Contoh ungkapan if..else:

```
val nilai = 90
if(nilai < 50) {
    println("Nilai jelek")
}
```

```
    } else {  
        println("Ayah memberiku hadiah")  
    }
```

Ketika satu ekspresi berada di dalam tubuh ekspresi lain maka itu disebut ekspresi bersarang. Perhatikan contoh berikut.

```
val num = 101  
if(num<0)  
    println("Bilangan Negatif")  
else {  
    //Nested expression  
    if(num%2 == 0)  
        println("Bilangan Genap")  
    else  
        println("Bilangan Ganjil")  
}
```

Ekspresi `when` di Kotlin berfungsi sama dengan `switch case` dalam bahasa pemrograman lain seperti C, C ++ dan Java. Contoh:

```
var ch = 'A'  
  
when(ch){  
  
    'A' -> println("A adalah Vokal")  
    'E' -> println("E adalah Vokal")  
    'I' -> println("I adalah Vokal")  
    'O' -> println("O adalah Vokal")  
    'U' -> println("U adalah Vokal")  
  
    else -> println("$ch adalah Konsonan")  
}
```

Loop for

Loop (perulangan) di Kotlin digunakan untuk melakukan perulangan terhadap elemen array, range, koleksi dll. Contoh:

```
for(n in 10..15){  
    println("Loop: $n")  
}
```

Kode ini akan memberikan hasil pada layar:

```
Loop: 10  
Loop: 11  
Loop: 12  
Loop: 13  
Loop: 14  
Loop: 15
```

Contoh berikut untuk menampilkan elemen-elemen array menggunakan loop for.

```
val myArray = arrayOf("ab", "bc", "cd", "da")  
for (str in myArray){  
    println(str)  
}
```

```
val myArray = arrayOf("Steve", "Robin", "Kate", "Lucy")  
for((index, value) in myArray.withIndex()){  
    println("Value at Index $index is: $value")  
}
```

Loop while

Loop while digunakan untuk mengulangi blok kode berulang kali selama kondisi yang diberikan mengembalikan true. Contoh:

```
var num = 10  
  
while(num>=5){  
    println("Loop: $num")  
    num--  
}
```

Loop do-while

Loop do-while mirip dengan loop while kecuali memeriksa kondisi pada akhir iterasi. Loop do-while setidaknya akan berjalan sekali walaupun kondisi yang diberikan salah.

```
var num = 100  
do {  
    println("Loop: $num")
```

```

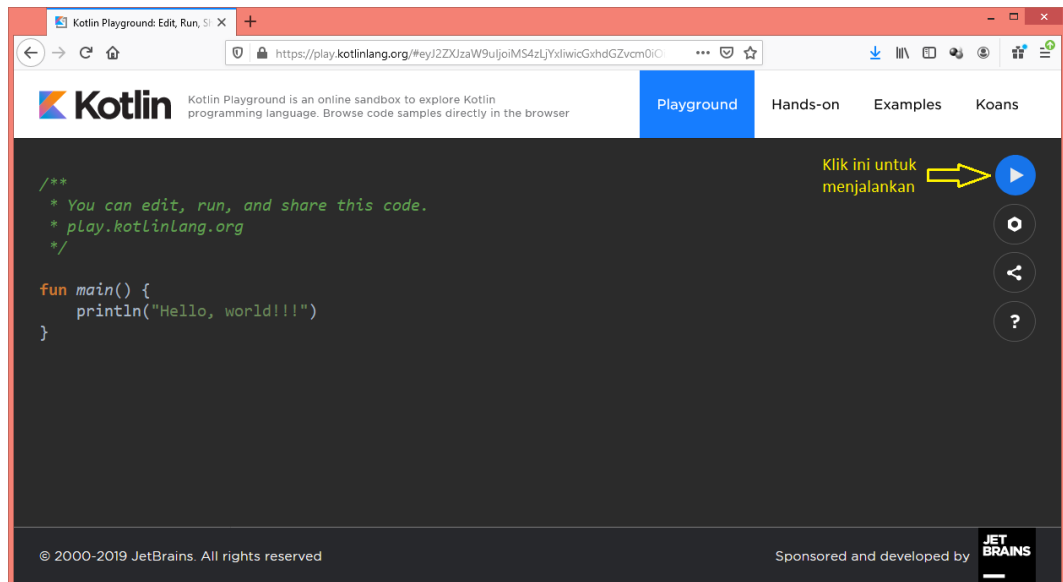
        num++
    }
    while (num<=105)

```



PRAKTIK

1. Anda akan menggunakan compiler Kotlin secara online. Lewat browser buka url: **play.kotlinlang.org** dengan tampilan seperti berikut:



2. Jalankan program yang ada dengan klik tombol lingkaran biru di sebelah kanan atas. Perhatikan hasil running program di bagian bawah jendela browser.
3. Tambahkan kode sehingga menjadi seperti berikut. Kemudian jalankan lagi, perhatikan hasilnya.

```

fun main() {
    println("Hello, world!!!")
    println("Halo, apa kabar")
    println("Ini lagi coding dengan Kotlin")
}

```

4. Tulis program dengan kode seperti berikut, dan perhatikan hasilnya.

```

fun main() {

```



```
val bilangan: Byte = 99
println("bilangan: $bilangan")
}
```

5. Gantilah bilangan 99 dengan 128, apa yang terjadi?
6. Tulis program dengan kode seperti berikut, dan perhatikan hasilnya.

```
fun main() {
    var bigByte: Byte = Byte.MAX_VALUE
    var smallByte: Byte = Byte.MIN_VALUE
    println("Nilai Byte terbesar adalah: " + bigByte)
    println("Nilai Byte terkecil adalah: $smallByte")
}
```

7. Ulangi langkah 6. dengan tipe data Byte diganti dengan: Short, Int, Long.
8. Tulis program dengan kode seperti berikut, dan perhatikan hasilnya.

```
fun main() {
    // all floating numbers are double by default
    // unless you suffix the value with F letter
    val num = 101.99
    println("$num")
}
```

9. Tulis program dengan kode seperti berikut, dan perhatikan hasilnya.

```
fun main() {
    val boolValue = false
    println("$boolValue")

    val ch = 'A'
    println("$ch")

    val ch2: Char
    ch2 = 'Z'
    println("$ch2")
}
```

10. Tulis program dengan kode seperti berikut, dan perhatikan hasilnya.

```
fun main(args: Array<String>) {

    val num1 = 101.99
    val num2 = 100.50
    var op: Double

    op = num1 + num2
    println("Addition: $op")

    op = num1 - num2
    println("Subtraction: $op")
}
```

```

    op = num1 * num2
    println("Multiplication: $op")

    op = num1 / num2
    println("Division: $op")

    op = num1 % num2
    println("Modulus: $op")
}

```

11. Tulis program dengan kode seperti berikut, dan perhatikan hasilnya.

```

fun main(args : Array<String>){

    var nama = "Susi Susanti"

    /**
     * String interpolation
     */
    println("Panjang String: ${nama.length}")

    /**
     * Atau anda dapat menampilkan seperti ini
     */
    println("Panjang String: " + nama.length)
}

```

12. Tulis program dengan kode seperti berikut, dan perhatikan hasilnya.

```

fun main() {
    var arrNilai = arrayOf<Int>(70, 80, 60, 90)

    println("Nilai ke 2: " + arrNilai.get(1))
}

```

13. Modifikasilah program di atas, sehingga elemen ke-2 diganti menjadi 85.

14. Tulis program dengan kode seperti berikut, dan perhatikan hasilnya.

```

fun main() {
    val nilai = 90
    if(nilai < 50) {
        println("Nilai jelek")
    } else {
        println("Ayah memberiku hadiah")
    }
}

```

15. Modifikasilah program ini sehingga menampilkan 3 kemungkinan:

50 > nilai → nilai jelek

51 ≤ nilai < 70 → nilai cukup

nilai ≥ 71 → nilai bagus

16. Tulis program dengan kode seperti berikut, dan perhatikan hasilnya.

```
fun main() {  
    val num = 101  
    if(num<0)  
        println("Bilangan Negatif")  
    else {  
        //Nested expression  
        if(num%2 == 0)  
            println("Bilangan Genap")  
        else  
            println("Bilangan Ganjil")  
    }  
}
```

17. Gantilah nilai varabel num dengan: -23, kemudian 12. Perhatikan hasilnya.

18. Tulis program dengan kode seperti berikut, dan perhatikan hasilnya.

```
fun main() {  
    var umur = 15  
  
    when(umur) {  
        in 1..16 -> {  
            val selisih = 17 - umur  
            println("Anda akan berhak ikut pemilu dalam $selisih tahun  
lagi")  
        }  
        in 17..100 -> println("Anda berhak untuk ikut pemilu")  
    }  
}
```

19. Gantilah nilai varabel umur dengan: 21, perhatikan hasilnya.

20. Tulis program dengan kode seperti berikut, dan perhatikan hasilnya.

```
fun main() {  
    val myArray = arrayOf(3, 5, 8, 2, 4, 7, 6)  
    for((index, value) in myArray.withIndex()){  
        println("Nilai pada indeks $index adalah: $value")  
    }  
}
```

21. Gantilah loop **for** dengan loop **while** dengan tetap memberikan hasil tampilan yang sama.
22. Tambahkan kode program ini untuk menampilkan tertinggi.



LATIHAN

1. Tulis program untuk membandingkan dua buah string apakah sama ataukah tidak. Misalnya, string1 = "Susi Susanti", string2 = "Susi susanti".
2. Diketahui suatu variabel bernilai 45. Buat program yang dapat menentukan bahwa bilangan ini lebih atau kurang dari 50, dan kelipatan 5 atau bukan kelipatan 5.
3. Diketahui suatu array yang berisi: 21, 32, 43, 24, 27, 28. Tulislah program yang dapat menampilkan semua bilangan genap dalam array.
4. Modifikasilah program pada langkah 18 sehingga dapat memberikan output 4 pilihan:
 - a. jika umur 6..12 → Anda masih lama untuk ikut pemilu
 - b. jika umur 13..16 → Anda akan berhak ikut pemilu n tahun lagi
 - c. jika umur 17..100 → Anda berhak untuk ikut pemilu
 - d. jika umur selain tersebut di atas → Jangan pikirkan dulu untuk ikut pemilu
5. Buatlah program untuk menampilkan bilangan kelipatan 3, kelipatan 5, dan kelipatan 3 dan 5, dari range 0 – 50.



TUGAS

1. Tuliskan soal tugas yang harus dikerjakan mahasiswa di sini (modul untuk mahasiswa tidak perlu diberi jawaban; modul untuk dosen perlu disertakan jawaban)
- 2.



REFERENSI

1. <https://play.kotlinlang.org/byExample/overview>
2. <https://www.guru99.com/kotlin-tutorial.html>
3. <https://codelabs.developers.google.com/android-kotlin-fundamentals/>
4. <https://beginnersbook.com/2017/12/kotlin-tutorial/>