



Muhamad Azriel Akbar

Software Developer & Trainer
Udacoding

APA ITU KOTLIN?

KOTLIN



SEJAK KAPAN ADA KOTLIN ?

KENAPA HARUS KOTLIN?

1. Ringkas (Concise)

```
Barang  getName()
7      public class Barang {
8
9          String nama ;
10         String stock ;
11         String masuk_tgl ;
12         String pemilik ;
13
14
15         public String getName() {
16             return nama;
17         }
18
19         public void setName(String nama) {
20             this.nama = nama;
21         }
22
23         public String getStock() {
24             return stock;
25         }
26
27         public void setStock(String stock) {
28             this.stock = stock;
29         }
30
31         public String getMasuk_tgl() {
32             return masuk_tgl;
33         }
34
35         public void setMasuk_tgl(String masuk_tgl) {
36             this.masuk_tgl = masuk_tgl;
37         }
38
39         public String getPemilik() {
40             return pemilik;
41         }
42
43         public void setPemilik(String pemilik) {
44             this.pemilik = pemilik;
45         }
46     }
47 }
```

```
5  */
6
7  data class Barang (
8
9      var nama: String? = null,
10     var stock: String? = null,
11     var masuk_tgl: String? = null,
12     var pemilik: String? = null,
13
14
15 )
16
```

2. Aman (Safe)

Mistake

```
var output: String  
output = null    // Compilation error
```

Kotlin melindungi Anda dari kesalahan menggunakan operasi pada data null

```
val name: String? = null    // Nullable type  
println(name.length())     // Compilation error
```


3. Serbaguna (Versatile)

4. Mudah Dikenalkan (Interoperable)



Variable

Mutable Variable

Var

```
var myName = "Azriel Akbar"  
myName = "Muhamad Azriel Akbar"  
print(myName)//output Muhamad Azriel Akbar
```

Immutable Variable

Val

```
val cardId = 101200102399102  
cardId = 111111111111 // Compilation Error  
print(cardId)
```

BASIC TYPE

Integer :

```
val a = 20.0  
val b = a.toInt()  
print(b) //20.0
```

Mari kita bandingkan dengan Java :

```
Double a = 20.0;  
Integer b = Integer.valueOf(a.toString());
```

Merepotkan bukan ? Lebih lengkapnya mari kita bahas beberapa Basic Type di Kotlin.

String

```
val company = "Udacoding"
```

```
val greeting = """Hi, $company..  
Welcome to Kotlin Class !"""
```

```
val company = "Udacoding"  
print(company[0]) //U
```

String Literal

```
val a = 20  
val b = 15  
val result = "Hasil penjumlahan $a dan $b adalah ${a + b}"  
print(result) //Hasil penjumlahan 20 dan 15 adalah 35
```


Boolean

```
val isAuthenticated = true  
val isInputValid = false
```

Try online

<https://try.kotlinlang.org/>

Operator

Operator Aritmatika

```
val a = 20
val b = 15

val addition = a + b
val subtraction = a - b
val division = a / b
val multiplication = a * b
val remainder = a % b

println(addition) //35
println(subtraction) //5
println(division) //1
println(multiplication) //300
println(remainder) //5
```

Operator Logika

```
a == b // a sama dengan b
```

```
a != b // a tidak sama dengan b
```

```
a < b // a lebih kecil dari b
```

```
a > b // a lebih besar dari b
```

```
a <= b // a kurang dari sama dengan b
```

```
a >= b // a lebih dari sama dengan b
```

Operator Boolean

- **OR**, `||` → benar atau benar (salah satu syarat terpenuhi)
- **AND**, `&&` → benar dan benar (kedua syarat harus terpenuhi)
- **!** → negasi, kebalikan kondisi yang dinyatakan

Contoh penggunaan :

```
val isAdiHadir = true
val jamMasukAdi = 7.00
val jamMasukSekolah = 8.00

if (isAdiHadir && jamMasukAdi < jamMasukSekolah) {
    print("Adi masuk tepat waktu")
} else {
    print("Adi telat masuk sekolah")
}

// Output: Adi masuk tepat waktu
```

Array

```
val countries = arrayOf("Indonesia","USA","China","Australia","Sri Lanka")
val countries2 = arrayOf<String>("Indonesia","USA","China","Australia","Sri Lanka") //sebuah Array kumpulan dari String
val myArray = arrayOf("Hi",1,1L,1.2,false)

val array1 = arrayOf(1,2,3,4)
val array3 = arrayOf<Long>(1,2,3,4)

array3.get(0)
array3[0]

array1[1] = 6
array1.set(1,6)

println("Ukuran datanya adalah ${array1.size}") //Ukuran datanya adalah 4
```

Data Type Optional

```
val name: String = "Muhamad Azriel Akbar"  
var age: Int = 18  
var height: Double = 143.1  
val gender: Char = 'L'  
!val isTaken: Boolean = false  
val aboutMe = "Hi, my name's $name. I'm $age and I $gender. My height is $height."
```


Function

Control Flow, If Expression

```
val nMTK = 75
val nBIndo = 80
val nBIng = 56

val nRataRataAdi = nMTK + nBIndo + nBIng
val nRataRataLulus = 83

val hasilUjian = if (nRataRataAdi < nRataRataLulus) "Tidak Lulus" else
"Lulus"

val pesan = if (nRataRataAdi < 30) {
    "Sepertinya kamu kurang giat berlatih :) Jangan malas belajar ya!"
} else if (nRataRataAdi < 60) {
    "Tetap semangat dan teruslah berlatih !"
} else if (nRataRataAdi > 70) {
    "Kamu luarbiasa ! Tingkatkan terus prestasimu ya !"
} else {
    "Mantap! Tetap pertahankan!"
}

println(hasilUjian) //Lulus
println(pesan) //Kamu luarbiasa ! Tingkatkan terus prestasimu ya !
```

Control Flow, When Expression

```
val nMTK = 75
val nBIndo = 80
val nBIng = 56
val nRataRataAdi = nMTK + nBIndo + nBIng
val nRataRataLulus = 83

when(nRataRataLulus){
    in 90..100 -> {
        print("Nilai A")
    }
    in 70..90 -> {
        print("Nilai B")
    }
    in 50..70 -> {
        print("Nilai C")
    }
    in 30..50 -> {
        print("Nilai D")
    }
    in 0..30 -> {
        print("Nilai F")
    }
    else -> {
        print("Nilai tidak valid !")
    }
}
```

Data Classes

```
data class User(val name: String, val age: Int)
```

```
class User
{
    private String name;
    private Int age;

    public User(String name, Int age) {
        this.name = name;
        this.age = age
    }
    public String getName() {
        return this.name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public Int getAge() {
        return this.age;
    }
    public void setAge(Int age) {
        this.age = age;
    }
}
```

```
// Buat Data Class Profile
data class Profile(val firstName: String,
                  val lastName: String,
                  val company: String,
                  val phone: String)

// Membuat objek data class
val profileA = Profile ("Muhamad Azriel","Akbar","Udacoding","082120351636")
println("Hi, im ${profileA.firstName} ${profileA.lastName} I work at ${profileA.company}")
```

OOP in Kotlin

```
class namaClass {  
    var namaProperti: TipeData = "isi"  
    fun namaMethod() {  
        //perintah  
    }  
}
```

```
class Kucing {  
    //properties  
    var namaKucing: String? = null  
    var jenisKucing: String? = null  
    var warnaKucing: String? = null  
  
    //method  
    fun mengeong(){  
        println("Kucing $namaKucing mengeong")  
    }  
    fun lari(){  
        println("Kucing $namaKucing berlari")  
    }  
    fun info(){  
        println("Nama : $namaKucing - Jenis: $jenisKucing - Warna:  
$warnaKucing")  
    }  
}
```

```
//Buat object
val anggora = Kucing()

//Isi properti
anggora.namaKucing = "Jojo"
anggora.jenisKucing = "Anggora"
anggora.warnaKucing = "Putih"

//Panggil method
anggora.mengeong()
anggora.lari()
anggora.info()
```