

PEMROGRAMAN BERGERAK

PERTEMUAN 2

DOSEN PENGAMPU:

Dr. Yasdinul Huda, MT



Oleh:

Aldeki Adia

21076082

PRODI PENDIDIKAN TEKNIK INFORMATIKA DAN KOMPUTER

JURUSAN TEKNIK ELEKRONIKA

FAKULTAS TEKNIK

UNIVERSITAS NEGERI PADANG

2023

DASAR KOTLIN

A. Pengertian Kotlin

Kotlin adalah bahasa pemrograman modern yang dirancang untuk berjalan di atas platform Java, namun dengan sintaksis yang lebih bersih, ekspresif, dan aman. Kotlin dikembangkan oleh JetBrains dan diadopsi secara luas dalam pengembangan aplikasi Android, serta digunakan untuk berbagai proyek perangkat lunak lainnya. Keunggulan Kotlin termasuk null safety yang terintegrasi, pengkodean yang lebih ringkas, dan dukungan untuk berbagai paradigma pemrograman seperti pemrograman berorientasi objek, fungsional, dan reaktif. Kotlin menjadi pilihan yang populer bagi para pengembang karena meningkatkan produktivitas, mengurangi kesalahan, dan memungkinkan penulisan kode yang lebih mudah dipelihara.

B. Sejarah Kotlin

Kotlin dikembangkan pada tahun 2011 oleh JetBrains yang dikepalai oleh Dmitry Jemerov. Ia ingin menciptakan bahasa pemrograman yang sederhana dan lebih ringkas dibandingkan bahasa pemrograman lain yang banyak digunakan saat itu. Dengan kode yang lebih ringkas, developer jadi tak perlu menulis baris kode yang panjang untuk sebuah perintah. Karenanya, proses pengembangan aplikasi bisa jauh lebih mudah dan cepat. Saat itu, Jemerov memperhatikan Java yang ia nilai memiliki aturan coding terlalu kompleks dan membuat pekerjaan developer kurang efisien.

Namun, ia dan timnya juga menyadari bahwa Java memiliki banyak keunggulan. Seperti fiturnya yang banyak, fleksibel, serta memiliki ekosistem yang luas. Karena itulah Jemerov bersama JetBrains ingin membuat bahasa pemrograman yang lebih mudah, tapi menggunakan ekosistem yang luas milik Java tersebut. Kotlin baru dirilis ke publik dan digunakan secara luas pada tahun 2012. Bahasa pemrograman ini bersifat open-source, dengan lisensi Apache 2. Karena sifatnya yang open-source itulah perkembangannya menjadi sangat cepat karena banyak developer yang bisa terlibat di dalamnya.

Hingga kini, bahasa pemrograman Kotlin banyak digunakan oleh perusahaan-perusahaan besar untuk mengembangkan aplikasi mereka. Beberapa diantaranya adalah

Netflix, Amazon, Pinterest, Uber, dan Trello. Perkembangan ini semakin pesat ketika Kotlin mendapat dukungan dari Google pada 2017. Bahasa pemrograman ini dinobatkan sebagai bahasa pemrograman kelas pertama untuk aplikasi Android. Bahkan, Google lebih memprioritaskan Kotlin dibandingkan C++ atau Java yang sudah ada terlebih dahulu.

C. Mengapa kita harus menggunakan kotlin

Kotlin mulai banyak dipakai di perusahaan besar. Contohnya Gojek, Slack, Trello dan Pinterest. Apa kamu tertarik bisa bekerja di perusahaan seperti yang disebutkan tadi? Kalau tertarik mulai cari tahu benefit dari mempelajari Kotlin. Selain Kotlin itu modern, Kotlin merupakan bahasa yang powerfull tentu cocok bagi developer dalam membuat aplikasi Android. Karena Kotlin adalah bahasa nomor satu untuk pengembangan aplikasi Android. Itulah alasan utama kenapa kamu harus mempelajarinya.

Sebagai developer jaman now, tentu harus keep going ahead (maju terus), karena perkembangan teknologi Android yang begitu pesat. Karena Kotlin sangat berkesinambungan dengan Android. Untuk itu kamu tak boleh lengah dan ketinggalan. Skill Kotlin akan sangat dibutuhkan dan menjadi nilai plus bagi kamu yang menguasainya. Lantas, kenapa kita harus mempelajari Kotlin? Adapaun alasan yakni sebagai berikut

1. Open Source

Open-sources Kotlin dapat bebas kita gunakan secara gratis. Kamu juga bisa melihat proses pengembangan, modifikasi atau bahkan ikut berkontribusi di dalamnya. Selain kamu menggunakan Kotlin untuk pengembangan aplikasi Android, Kotlin juga bisa dipakai dalam membangun aplikasi server dan web. Bahkan saat ini sedang dikembangkan Kotlin/Native yang memungkinkan developer untuk dapat mengembangkan Kotlin di platform iOS dan Mac OS

2. Mudah dipelajari

Kenapa Kotlin begitu digandrungi para developer Android? Karena selain simpel, kesederhanaan dari Kotlin membuat para developer jatuh cinta. Kesederhanaan yang dimaksud berupa sintak, dll. Jika sebelumnya kamu menggunakan Java untuk membuat aplikasi Android, maka tidak akan mengalami kesulitan ketika kamu migrasi ke Kotlin. Ketika kamu mempelajari

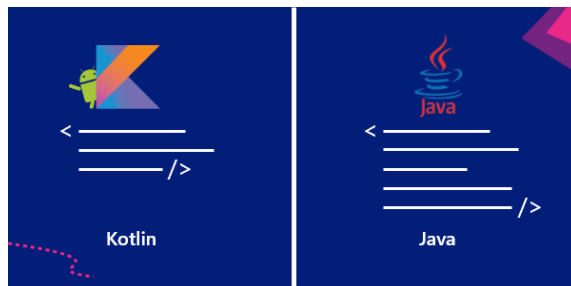
Kotlin, katakan selamat tinggal pada titik koma (;) di akhir kode. Perlu kamu tahu Kotlin bebas dari titik koma, tidak seperti Java tiap akhir kode wajib memakai titik koma (;). Sehingga Kotlin mudah dipelajari.

3. Banyak Perusahaan yang menggunakan kotlin

Perusahaan besar seperti Gojek sudah memakai Kotlin di project Java yang sudah ada. Hal ini tentu banyak diikuti oleh perusahaan besar lainnya. Artinya apa? Semakin banyak startup atau perusahaan besar memakai dan mengimplementasikan Kotlin. Maka peluang akan kebutuhan developer Kotlin pun semakin meningkat. Contoh perusahaan memakai Kotlin antara lain, Gojek, Pinterest, Evernote, Slack Trello, dsb.

4. Bahasanya Simple dan ringkas

Bagi yang suka pusing melihat kerumitan sintak atau simbol bahasa pemrograman lainnya., di Kotlin adanya pengurangan drastis dari jumlah kode boilerplate. Seperti pada Java kodingan mu bisa bertele-tele sampai mengulangi kode yang sama. Sedangkan Kotlin setidaknya akan meringkas kode dari Java, misal pada Java 50 baris, dengan Kotlin kita hanya menghabiskan 35 baris saja.



Gambar ini adalah contoh dari kotlin. Dan sesimple apa codingannya?. Berikut contohnya

Contoh Kode Java

```
TextView text = (TextView) findViewById(R.id.myTV);  
text.setText("Halo Rekan Dicoding");
```

Contoh Kode Kotlin

```
myTV.setText("Halo Rekan Dicoding")
```

5. Kotlin lebih ringan daripada java

Kotlin dapat meningkatkan kecepatan eksekusi fungsi Lambda. Jika dibandingkan dengan Java, kedua bahasa ini sama-sama berjalan pada JVM,

tentunya soal kecepatan Kotlin lebih unggul. Kecepatan seperti apa? Debugging atau me-run aplikasi baik itu pada emulator maupun pada real device. Itu sih yang saya alami, ketika menulis kode program menggunakan Kotlin.

6. Lebih aman dari kesalahan error

Kode atau sintak yang ditulis pada Kotlin akan terhindar dari error. Seperti ketika kamu ngoding di Java maka kamu sudah terbiasa bersahabat dengan NullPointerExceptions (NPE). Exception bisa membuat developer frustrasi, karena menyebabkan aplikasi crash. Oleh karena itu Kotlin bebas dan aman dari NullPointerExceptions (NPE). Menurunkan tingkat bug yang menyebabkan aplikasi kamu berhenti secara tiba-tiba.

7. Terintegrasi dengan android studio

Kotlin dikembangkan oleh JetBrains, perusahaan yang mengembangkan IntelliJ basis awal Android Studio. Kotlin sangat terintegrasi dengan Android Studio. Karena keduanya bisa dibilang dikembangkan oleh perusahaan yang sama.

8. Interopodeble

Kotlin dan Java bisa kamu pakai secara berbarengan. Kode program Java bisa dipanggil dengan Kotlin, begitu pun sebaliknya. Keren juga kan? Selain itu kamu bisa menulis kode Java di dalam kelas Kotlin dan terdapat konversi ke Kotlin secara otomatis.

9. Kotlin tidak hanya unruk android

Bisakah Kotlin buat web? Jawabannya tentu bisa. Kotlin tidak hanya diperuntukan untuk Android saja. Seperti yang dibahas di awal-awal. Kotlin bisa dipakai untuk pengembangan aplikasi server, iOS (melalui Kotlin Native) bahkan sekaligus. Namun, dalam urusan membangun web Kotlin membutuhkan framework seperti Spring, Ktor, Spark Javalin, dsb. Selain itu juga bisa dipakai untuk menulis JavaScript (melalui Kotlin JS). Kotlin memiliki sifat interoperability 100% dengan Java. Maka apapun yang bisa menggunakan basis Java bisa juga menggunakan Kotlin.

D. Kelebihan dan kekurangan kotlin

1. Kelebihan

- Lebih ringkas

Seperti yang sempat dibahas di atas, bahasa pemrograman yang satu ini lebih ringkas jika dibandingkan dengan Java. Pengguna hanya perlu

menulis coding yang lebih pendek saat menggunakan Kotlin. Jadi, hal tersebut bisa menghemat tenaga dan waktu dari para penggunanya.

- Dapat dioperasikan dengan Java

Kelebihan selanjutnya yang dimiliki oleh Kotlin adalah kemudahannya saat dioperasikan dengan bahasa pemrograman lain yaitu Java. Pengguna tidak perlu merasa kesulitan lagi saat menggunakan Kotlin dalam proyek yang awalnya menggunakan bahasa Java.

- Mudah dipelajari

Salah satu bahasa pemrograman yang paling mudah untuk dipelajari adalah Kotlin. Jadi, jika sebelumnya sudah mampu menggunakan Java, pastinya akan merasa lebih mudah saat mempelajari bahasa pemrograman ini. Kotlin sebenarnya merupakan aplikasi multiplatform. Ini berarti Anda bisa menggunakannya untuk membangun aplikasi Android, iOS, dan aplikasi berbasis web. Namun, saat ini Kotlin lebih dikenal sebagai bahasa pemrograman untuk membuat aplikasi Android.

Hal ini tentu memudahkan Anda sebagai developer. Sehingga Anda tak perlu membuat kode dengan banyak bahasa pemrograman untuk membuat aplikasi multiplatform, semuanya bisa Anda kembangkan dengan Kotlin. Beberapa aplikasi populer yang dibangun dengan Kotlin adalah Tinder, Netflix, dan Slack. Aplikasi-aplikasi ini dapat Anda jalankan baik di Android maupun iOS.

Tingkat kesulitan dalam setiap bahasa pemrograman pasti berbeda-beda. Dibandingkan bahasa lain, Kotlin termasuk mudah dipelajari. Salah satu faktornya adalah karena bahasa pemrograman satu ini memiliki syntax yang sederhana dan jauh lebih singkat dibandingkan bahasa pemrograman lainnya seperti Java. Karena itu, tak sulit bagi Anda mengingat dan menerapkan perintah-perintah Kotlin dalam aplikasi Anda. Asalkan konsisten dalam mempelajarinya, tak butuh waktu terlalu lama bagi Anda untuk menguasai bahasa pemrograman ini.

- Mampu meminimalisir kesalahan

Kotlin adalah jenis bahasa pemrograman yang diketik secara statis. Jadi, penggunaanya bisa mengetahui jika terdapat kesalahan pada saat kompilasi. Bahasa pemrograman yang diketik secara statis memang akan melakukan type checking saat sedang melakukan kompilasi.

- Memiliki fitur-fitur yang beraneka ragam

Kelebihan selanjutnya dari Kotlin adalah telah dibekali dengan fitur lanjutan yang beraneka ragam fungsi. Misalnya, operator overloading, lambda expressions, hingga string templates.

- Aman dari error

Kotlin merupakan bahasa pemrograman yang diketik secara statis. Karena itu, lebih mudah bagi pengguna untuk mengidentifikasi kesalahan ketika kompilasi. Selain itu, menggunakan bahasa pemrograman satu ini bisa menghindari error karena terlindung dari NPE (Null Pointer Exceptions).

Seperti yang Anda ketahui, NPE sering menyulitkan developer yang menggunakan Java. NPE sering mengakibatkan error tak terdeteksi pada aplikasi. Akibatnya, aplikasi tersebut beresiko crash saat digunakan oleh user nantinya.

2. Kekurangan

- Kecepatan kompilasi yang berfluktuasi

Kotlin memang bisa melakukan kompilasi dengan waktu yang cepat dan bahkan lebih cepat dari Java. Namun, hal tersebut hanya dalam beberapa kasus saja. Pasalnya, tidak semua pengguna Kotlin merasa bahwa kompilasi yang dilakukan selalu bisa berjalan dengan cepat.

- Komunitas Kotlin masih cukup sedikit

Jika dibandingkan dengan Java, tentu saja Kotlin terbilang masih sangat baru. Bahkan, menurut para pengembangnya, Kotlin dibuat karena terinspirasi dari banyak bahasa pemrograman dan salah satunya adalah Java. Awal perilisannya pun di tahun 2016 sehingga sampai saat ini Kotlin masih belum memiliki banyak komunitas seperti yang dimiliki oleh Java.

- Kotlin developer masih sulit ditemukan

Selain komunitas yang masih belum berkembang pesat, kekurangan selanjutnya dari Kotlin adalah masih sedikitnya orang yang ahli menggunakan bahasa pemrograman ini. Permintaan Kotlin developer memang tiba-tiba meningkat setelah Google mengumumkan bahwa bahasa pemrograman ini dapat digunakan untuk membangun aplikasi Android.

E. Tipe data kotlin

1. Double

Double dipakai untuk variabel bernilai angka decimal.

2. Int

Int adalah data untuk variabel angka bilangan bulat.

3. Boolean

Data Boolean dipakai untuk variabel dengan nilai tertentu, yakni 1 (true) dan 0 (false).

4. Char

Char adalah data yang memuat variabel bernilai 1 karakter, semisal 'b', 'j', '-' dan seterusnya

5. String

String adalah data untuk variabel berupa nilai kumpulan char, misalnya 'selamat pagi' atau 'bintang123'.

F. Variabel

Penulisan variabel di Kotlin sangat mudah sebab kita tidak perlu lagi mendefinisikan tipe data pada variabel. Kotlin sudah punya kemampuan sendiri untuk mengenali tipe data yang digunakan pada variabel. Ada beberapa tipe variabel dalam Kotlin yakni var, val dan null:

1. Var (Nilai Bisa Diubah-Ubah)

Variabel var bersifat mutable atau dapat diubah nilainya pada line sintaks dasar Kotlin berikutnya meski sudah dideklarasikan. Untuk penulisan, Anda cukup mengetikkan var dan diikuti secara berurutan oleh nama variabel, tipe data dan nilai atau valuenya.

2. Val (Nilai Tidak Bisa Diubah)

Berkebalikan dengan var, val adalah variabel yang tidak bisa diganti lagi nilainya setelah dideklarasikan. Sifat variabel val ini dikenal dengan nama

‘assign-once’. Jika Anda memaksa mengganti nilai variabel `val`, maka sintaks dasar Kotlin Anda akan otomatis error. `Val` lebih cocok digunakan untuk data yang nilai variabelnya tetap alias tidak mungkin berubah bentuk dan besarnya. Cara menulis variabel ini sama seperti menulis `var`.

3. Null (Nullable; Bernilai Null)

Tipe variabel ketiga adalah variabel `null` yang berarti nilai dari variabel itu sendiri adalah `null`. Di sini tugas Anda sebagai programmer adalah untuk menentukan jika variabel pada sintaks dasar Kotlin bisa dibuat menjadi `null` atau tidak. Untuk menulis variabel `null`, Anda harus menambahkan ‘?’ setelah data. Semisal: `var percobaan: Int? = 3` `percobaan = null`

4. When

Struktur `when` digunakan untuk mengganti ‘`switch statement`’ yang biasanya terdapat pada bahasa pemrograman lain `when` digunakan untuk melakukan pengecekan struktur sintaks dasar Kotlin yang banyak. Dengan memakai `when`, maka struktur sintaks yang Anda tulis dapat menjadi lebih ringkas. Contoh:

```
when(g) {  
    3 -> println("g bernilai tiga")  
    4 -> println("g bernilai empat")  
}
```

5. If Statement

If dipakai untuk mengisi value atau nilai variabel. Penulisan `if statement` di Kotlin juga lebih ringkas dibandingkan penulisan `if statement` pada bahasa pemrograman lain. Contoh: `var maks = if(g > h) g else if(h < g) h else 0`

6. For

For dipakai untuk membuat pengulangan pada sintaks dasar Kotlin. Contohnya: `for(k in 5 ... 10)`

7. println (k)

Di sini, `k` adalah indeks pointer variabel `var` yang bebas Anda isi sendiri termasuk namanya (bisa `l`, `m`, `n`, `j`, dan seterusnya). ‘`in`’ setelah `k` adalah sintaks yang menghubungkan dalam struktur `for`, yang mana mengartikan tulisan Anda sebagai pengulangan `k` sebanyak `x` kali. Sedangkan jumlah `x` yang dimaksud ditentukan pada bilangan setelahnya yakni: `5...10` (jadi kita hendak melakukan pengulangan dari 5 sampai 10). Anda juga bebas mengisi jumlah pengulangan sesuai kebutuhan.

G. Fungsi kotlin

Kotlin sebenarnya merupakan aplikasi multiplatform. Ini berarti Anda bisa menggunakannya untuk membangun aplikasi Android, iOS, dan aplikasi berbasis web. Namun, saat ini Kotlin lebih dikenal sebagai bahasa pemrograman untuk membuat aplikasi Android. Hal ini tentu memudahkan Anda sebagai developer. Sehingga Anda tak perlu membuat kode dengan banyak bahasa pemrograman untuk membuat aplikasi multiplatform, semuanya bisa Anda kembangkan dengan Kotlin. Beberapa aplikasi populer yang dibangun dengan Kotlin adalah Tinder, Netflix, dan Slack. Aplikasi-aplikasi ini dapat Anda jalankan baik di Android maupun iOS.

H. Menulis sintaks dasar kotlin

1. Komentar

Dua Garis Miring (//)

Cara ini dipakai untuk menulis komentar 1 baris. Contoh:

```
// contoh komentar
```

```
println ("hai!")
```

Garis Miring Bintang (/*)

Tanda ini dipakai untuk menulis komentar yang lebih dari 1 baris. Contoh:

```
/*
```

```
ini adalah contoh komentar kedua
```

```
contoh komentar ini tidak untuk dijalankan.
```

```
*/
```

2. Blok

Blok Kode

Blok kode adalah kumpulan statement yang dibungkus dengan tanda kurung kurawal = ({ ... }). Contoh penulisan blok kode di Kotlin:

```
// blok kode if
```

```
if( b == 3 ){
```

```
println("B sama dengan 3")
```

```
println("Pernyataan ini benar...")
```

```
}
```

3. Array

bisa memakai fungsi `arrayOf(1,2,3, ..)` untuk membuat array pada sintaks Kotlin. Kemudian Anda bisa memakai indices untuk melakukan looping otomatis sesuai total array. Contoh:

```
fun main(){  
    val kota= arrayOf ("Senin", "Selasa", "Rabu") // untuk melooping array dapat  
    menggunakan indices  
    for (i in hari.indices) {  
        println(hari[i])  
    }  
}
```

Nanti outputnya adalah: Senin, Selasa, Rabu.

4. Function

Sesuai namanya, fungsi ini digunakan untuk memanggil aneka fungsi yang telah dibuat di fun main. Struktur data function pada Kotlin adalah:

```
// panggil fungsi-fungsi yang telah dibuat di main  
fun main() {  
    fungsia()  
    println(fungsiReturn()) // disimpan  
    val hasil = hitung(2.0, 8.0)  
    println(hasil)  
} // function biasa  
fun fungsil() {  
    println("contoh fungsi")  
}  
// Output  
// contoh fungsi
```

Sebab membuat fungsi perhitungan nilai 2.0 ditambah 8.0, maka hasil output yang akan keluar adalah 10.0. Untuk membuat function dengan nilai return, bisa

menambahkan kode : TipeData. Sedangkan function dengan parameter ditulis dengan nama Variabel:TipeData

5. Condition

Kita telah membahas kegunaan when pada pembahasan elemen-elemen dasar sintaks pada Kotlin. Contoh penulisan when secara lengkap adalah sebagai berikut:

```
fun main() {  
    val hari = "senin" when (hari) {  
        "senin" -> println("Pergi bekerja")  
        "minggu" -> println("Libur bekerja")  
        else -> println("Selamat Beraktivitas..")  
    }  
}
```

6. Looping

Looping dipakai untuk melakukan pengulangan. Ingat kembali penjelasan elemen 'for' pada bagian sebelumnya. Berikut ini adalah contoh penulisan looping:

```
fun main() {  
    // looping  
    for (k in 6..10) {  
        // tampilkan data  
        println(k)  
    }  
}
```