**LAPORAN WEBINAR**

**Mobile programing**

**(FUNCTION DAN OOP PADA KOTLIN)**

****

**Disusun Oleh:**

**LIDIA FITA SARI**

**3042019052**

**TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI KETAPANG**

**TAHUN AKADEMIK**

**2020/2021**

**Functional dan OOP pada Kotlin**

**OOP** (**Object Oriented Programming**) adalah suatu metode pemrograman yang berorientasi kepada objek. Tujuan dari **OOP** diciptakan adalah untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di kehidupan sehari-hari



**Apa itu Paradigma?**



Cara untuk mengklasifikasikan bahasa pemograman berdasarkan fitur yang dimilikinya.

**Paradigma pada Kotlin**

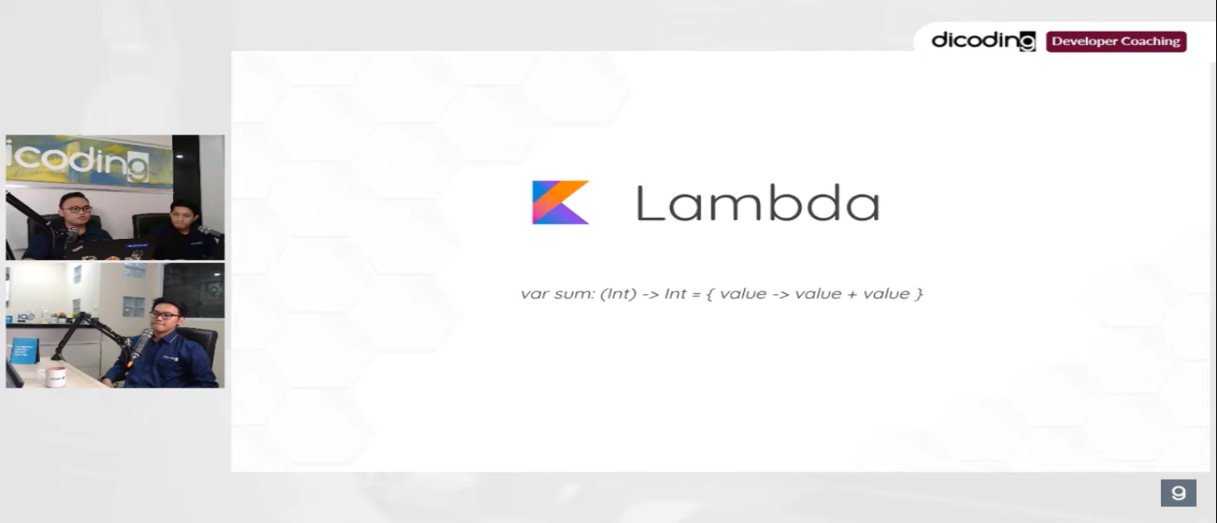
****

Kotlin adalah bahasa pemrograman yang dapat berjalan untuk berbagai macam platform (Multiplatform) dan juga mempunyai beberapa paradigma (Multiparadigm). Kotlin merupakan bahasa pemrograman yang ringkas, pragmatis, aman dan berfokus pada interoperabilitas dengan bahasa Java. Kotlin berjalan lancar dengan hampir semua library dan frameworks Java yang sudah ada.

**Functional**

****Function sederhananya adalah sebuah kumpulan baris kode yang dapat kita panggil dengan sederhana (sering kali cukup dipanggil hanya dengan 1 baris saja) tanpa perlu menulis baris kode yang banyak berulang-ulang kali.

**Lambda**

****Lambda adalah fungsi tanpa nama, dapat diperlakukan sebagai valuelambda dapat diperlakukan sebagai value, contohnya bisa dimasukkan ke variabel dan argument. Lambda juga dapat mengakses variabel sesuai pada scope lambda tersebut didefinisikan.

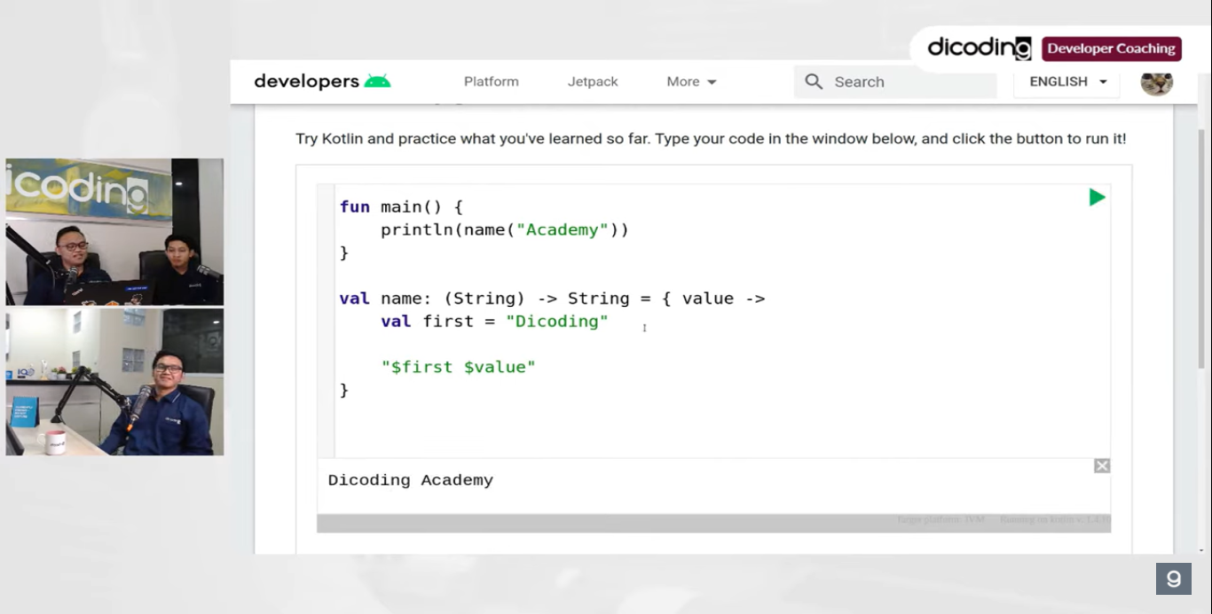
* Lambda adalah fungsi tanpa nama.
* Lambda dapat dipakai sebagai argument fungsi.
* Contoh umum penggunaan lambda pada Android.
* Lambda punyai syntax yang lebih ringkas.
* Lambda dapat mengakses variabel sesuai pada scope lambda tersebut didefinisikan.
* Penggunaan lambda pada collections.

****



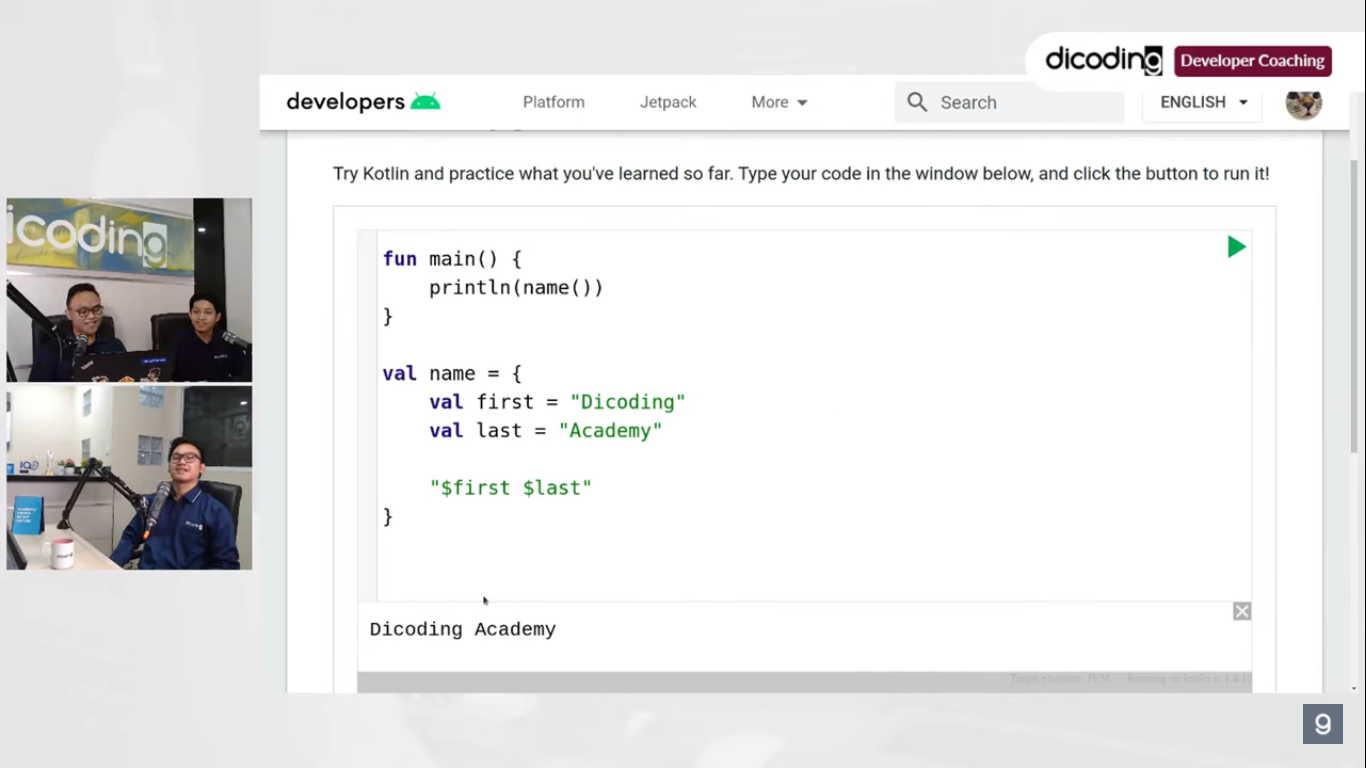
**Mendeklarasikan lambda**

Contoh Lambda yang mempunyai parameter

****

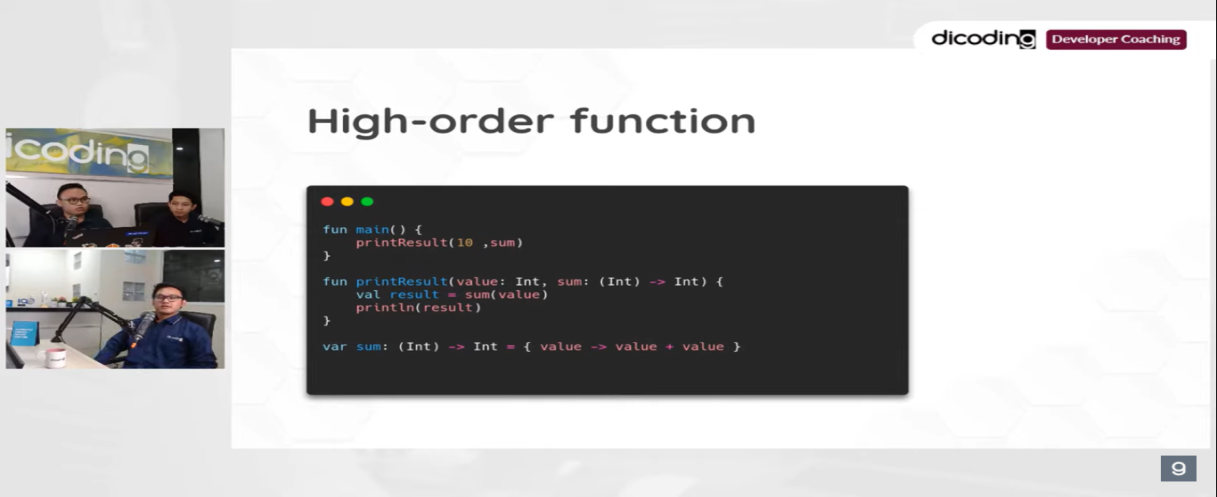
Contoh Lambda yang tidak mempunyai parameter

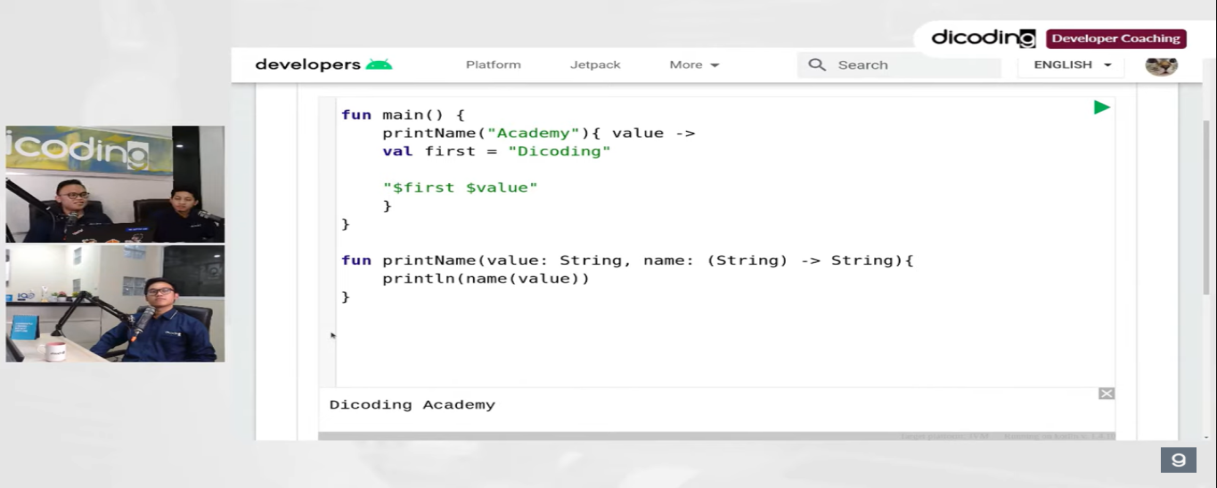
Tanpa return value, dengan menggunakan Unit.Tanpa parameter, dengan menggunakan kurung buka-tutup yang kosong.

****

**High-Order Function**

Higher order function adalah fungsi yang menggunakan fungsi lain sebagai parameter

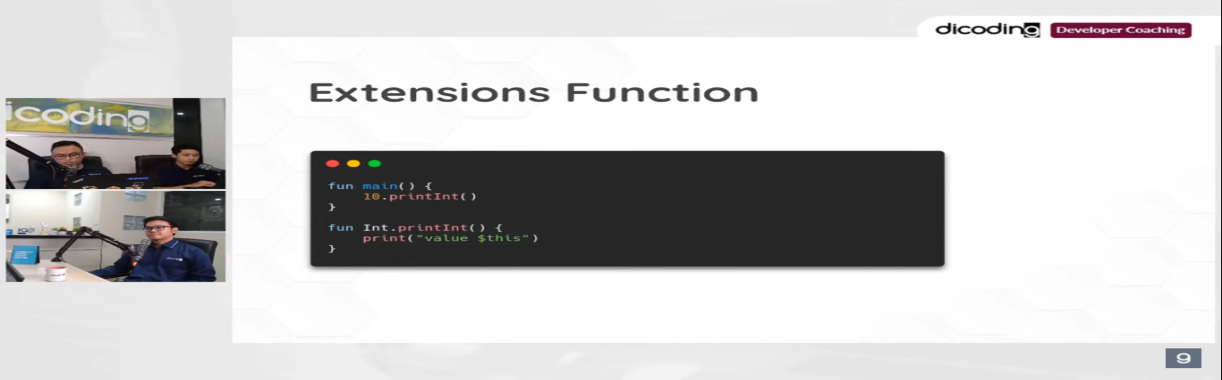
****

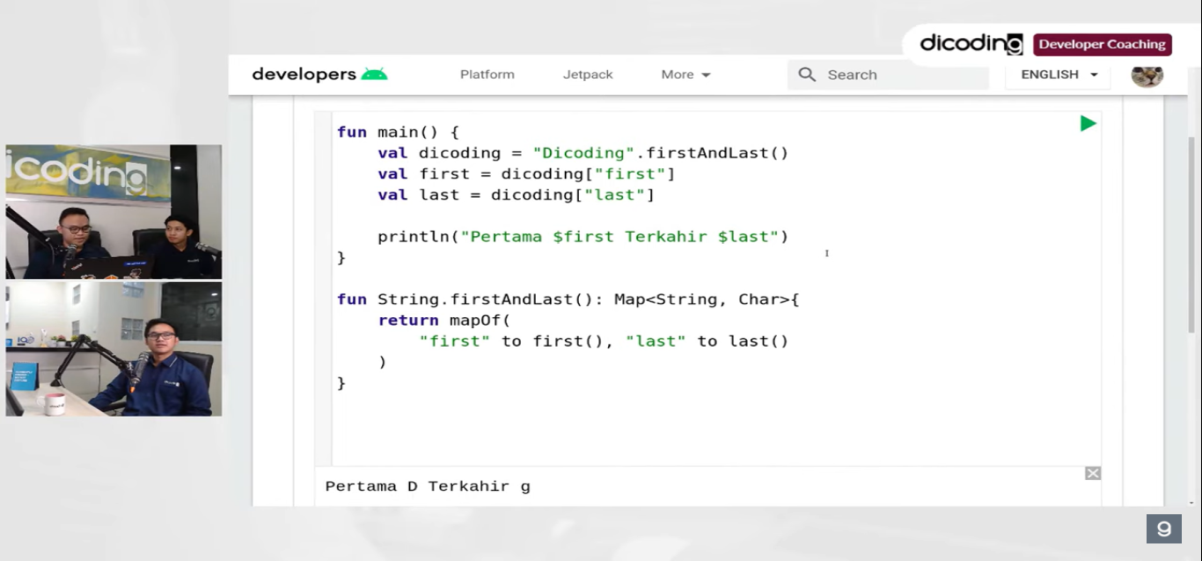
****

**Extention**

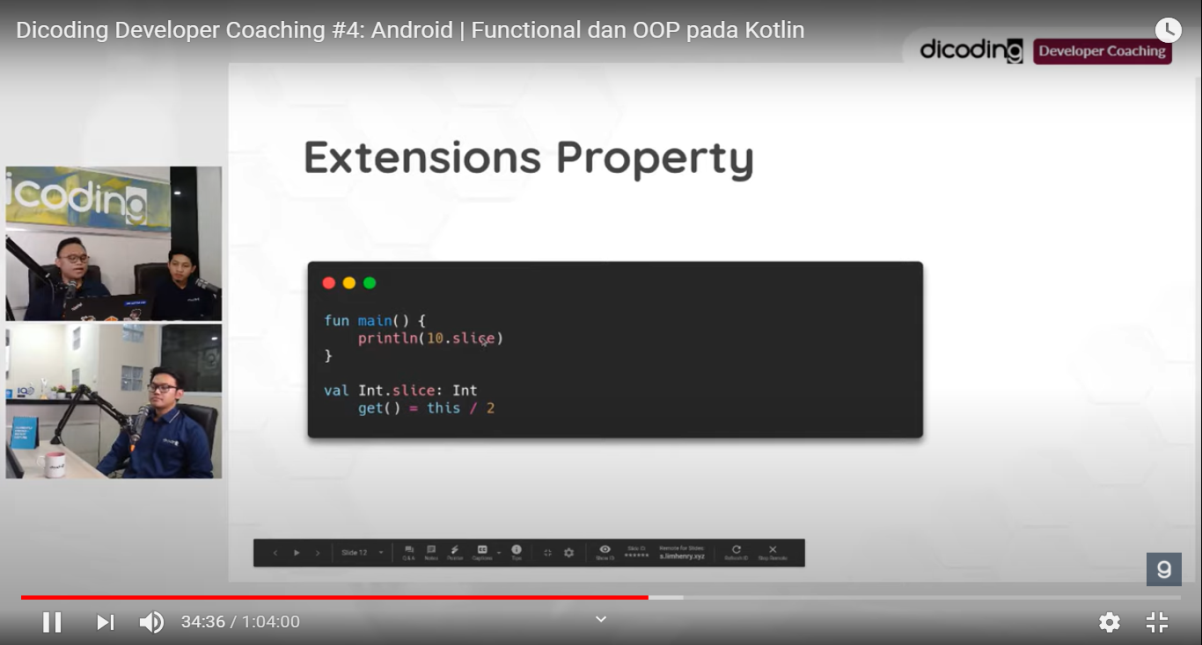
Extension adalah fitur pada Kotlin yang memungkinkan kita untuk menambahkan property dan function pada suatu kelas tanpa mengubah atau meng-extend kelas tersebut.

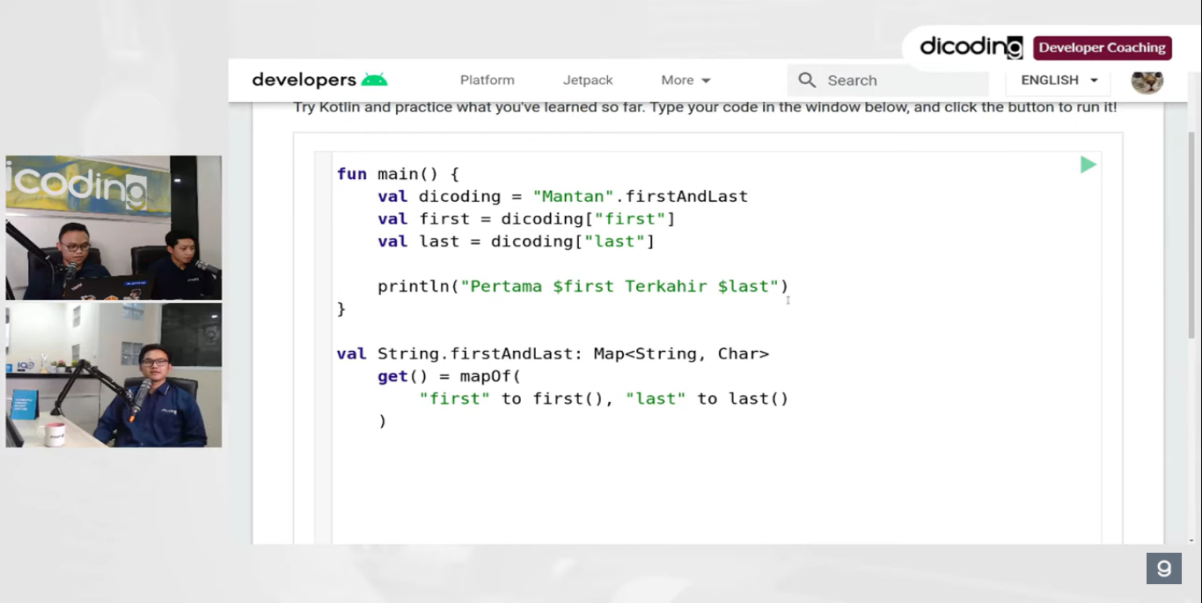
Contoh Extention function

****



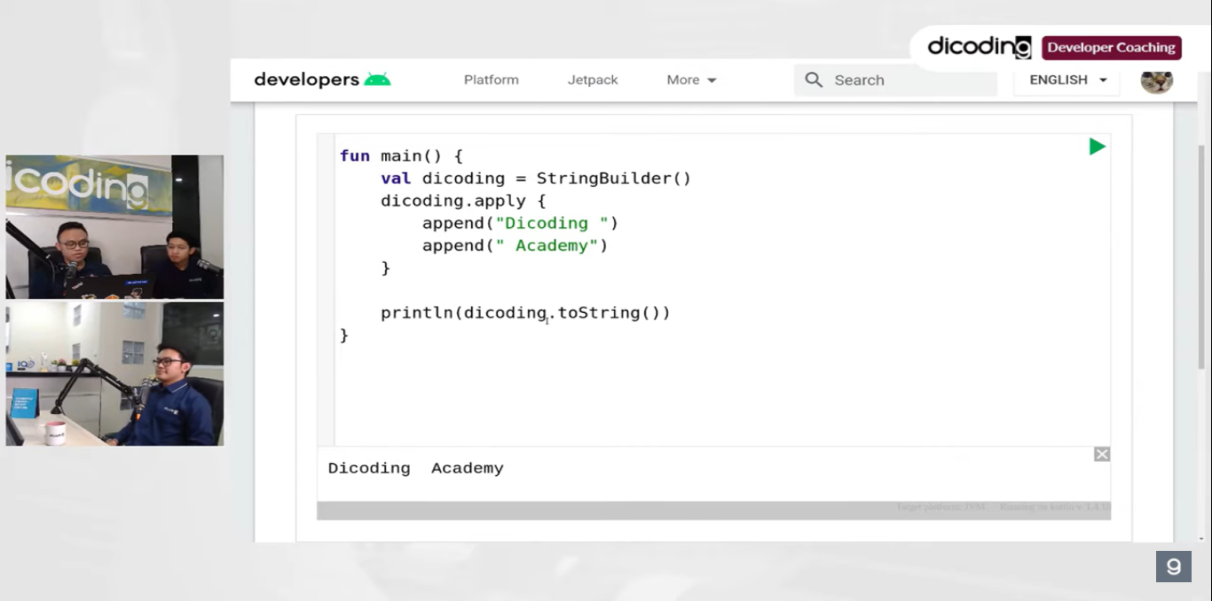
Contoh Extention property



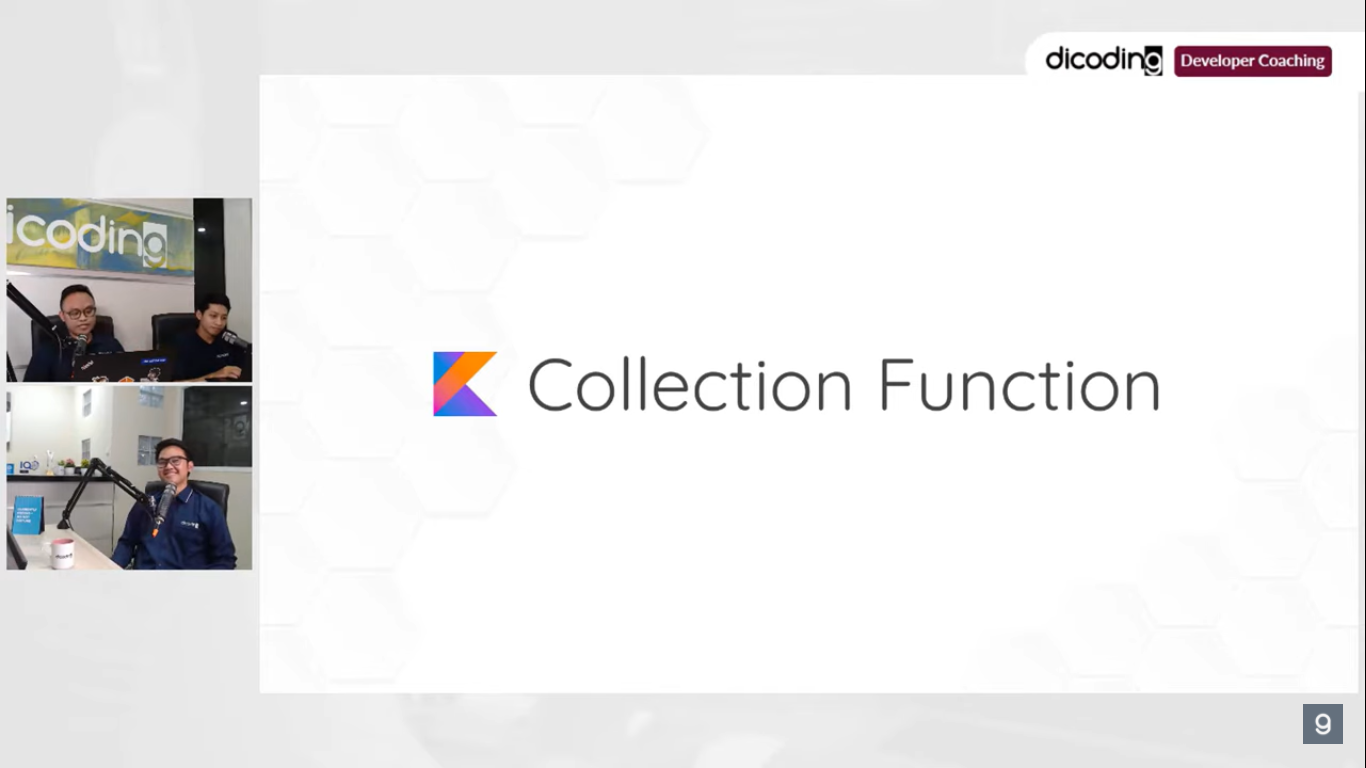


**Scope Function**

**Scope Function -**Kotlin adalah sebuah multiparadigm programming language. Artinya selain merupakan bahasa pemrograman berorientasi objek, dalam penulisan sintaksnya Kotlin menggunakan gaya functional programming.

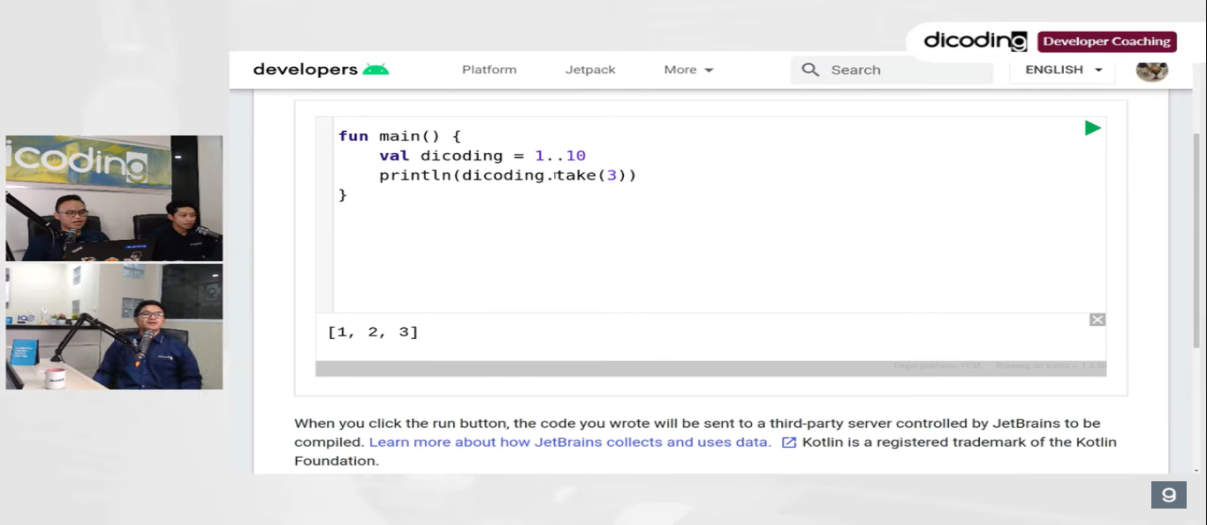
Contoh: ****

**Collection Function**

****

Collection Functiondigunakan untuk menampilkan daftar tautan yang dibuat sebagai koleksi di area kerja Ektron CMS400.NET. Fungsi ecmCollection sangat dapat disesuaikan, memungkinkan Anda untuk dengan mudah menentukan bagaimana data koleksi muncul di halaman Web. Yang ditampilkan di sini adalah format fungsi ecmCollection.

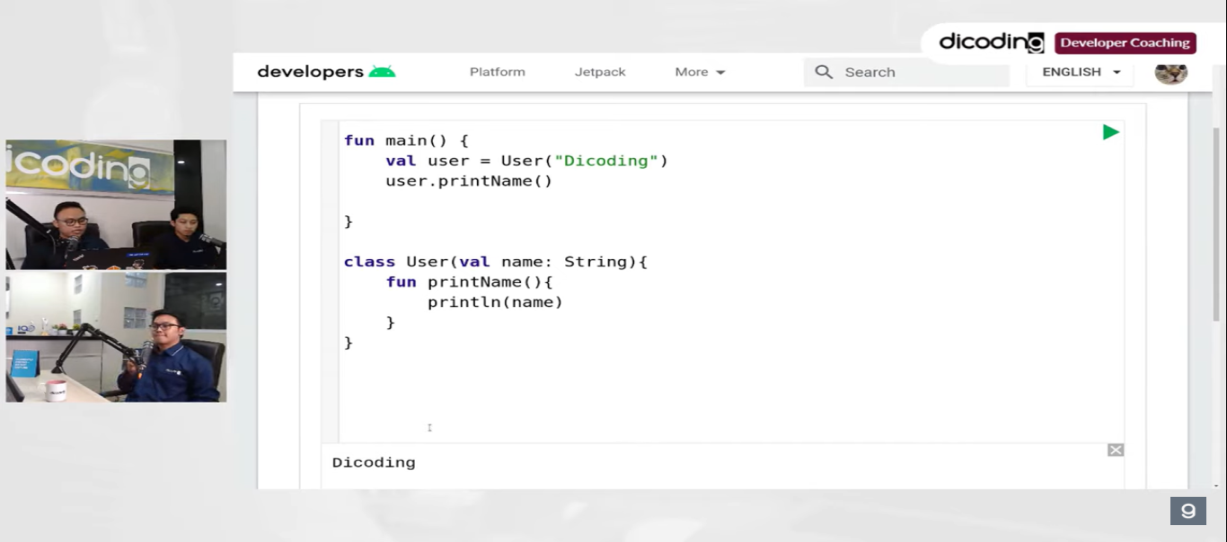
Contoh:

****

**Object-Oriented Programming**

**Pemrograman berorientasi objek**  ([Inggris](https://id.wikipedia.org/wiki/Bahasa_Inggris" \o "Bahasa Inggris): **object-oriented programming** disingkat **OOP**) merupakan [paradigma pemrograman](https://id.wikipedia.org/wiki/Paradigma_pemrograman" \o "Paradigma pemrograman) berdasarkan konsep "objek", yang dapat berisi [data](https://id.wikipedia.org/wiki/Data), dalam bentuk field atau dikenal juga sebagai atribut; serta kode, dalam bentuk fungsi/prosedur atau dikenal juga sebagai method. Semua data dan fungsi di dalam paradigma ini dibungkus dalam kelas-kelas atau objek-objek.

Contoh:

****