

INSTITUTO TECNOLÓGICO DE NUEVO LAREDO

Materia: Programación móvil 1 - Android

Docente: Ing. Juan Ahumada

Investigación Kotlin

Alumno: Yesenia Mendoza Martínez.

Especialidad: Ingeniería en Sistemas Computacionales.

Semestre: 8° **#Control:** 16100236

Nuevo Laredo Tamps. Febrero 2020

KOTLIN NATIVE

Kotlin es sin duda uno de los lenguajes de moda, sobretodo después del reciente anuncio de Google en Google I/O (el evento anual de Google dedicado a los desarrolladores) donde se hizo oficial que Kotlin pasa a ser un lenguaje de primer nivel para Android. A partir de ahora, Android Studio vendrá con 100% soporte para Kotlin.

Kotlin fue creado en 2010 por JetBrains, la empresa detrás del famoso IDE para Java IntelliJ IDEA (aunque se puede usar en cualquier otro IDE, como Eclipse).

Inicialmente Kotlin fue creado para aplicaciones Desktop (que es el mercado de JetBrains) pero ahora mismo Kotlin es multiplataforma con soporte para Android, compilación a JavaScript e incluso compilación nativa via LLVM para entornos donde JVM no sea una buena opción.

Proporciona una sintaxis y conceptos similares a los de otros lenguajes, como C#, Java y Scala, entre muchos otros. No pretende ser único, sino que se inspira en décadas de desarrollo del lenguaje. Cuenta con variantes que se orientan a la JVM (Kotlin/JVM), JavaScript (Kotlin/JS) y el código nativo (Kotlin/Native).

Kotlin / Native es compatible con las siguientes plataformas:

- iOS (arm32, arm64, simulador x86_64)
- macOS (x86_64)
- watchOS (arm32, arm64, x86)
- tvOS (arm64, x86_64)
- Android (arm32, arm64, x86, x86_64)
- Windows (mingw x86_64, x86)
- Linux (x86_64, arm32, MIPS, MIPS little endian, Raspberry Pi)
- WebAssembly (wasm32)

VENTAJAS DE USAR KOTLIN

* Ahorro de getters, setters y cia. Para cada atributo normalmente quieres tener el correspondiente get y set, incluir la propiedad en el método equals, ... Muchos IDEs te los generan automáticamente, pero al final, aunque sea código generado lo vas a tener que mantener (ej. cuando añadas un nuevo atributo). Kotlin te los proporciona automáticamente. Es decir, no los ves, el código no está, pero los puedes.

* Casting automático. Kotlin asume que si la comprobación tiene éxito el objeto pasa a ser del tipo concreto inmediatamente y lo trata como tal

* Organización de paquetes. Ya no hace falta que la organización de tu código en paquetes siga la estructura de carpetas en tu directorio

* Valores nulos. Kotlin no tiene el concepto de null.

* Funciones. En Kotlin, las funciones están al mismo nivel que las clases. Puedes definir funciones sin tener que crear primero una clase para ellas.

* Extensiones a clases existentes. Kotlin permite añadir nuevos métodos a una clase existente SIN tener que crear la subclase correspondiente. Esto permite que el nuevo método se pueda ejecutar sobre todas las instancias de la clase original y no sólo sobre las de la subclase que, en Java, estaría obligado a crear. De la misma forma, Kotlin también te permite añadir nuevas propiedades.

Compartir código entre plataformas

Los proyectos multiplataforma son compatibles entre diferentes objetivos Kotlin y Kotlin / Native. Esta es la forma de compartir el código Kotlin común entre muchas plataformas, incluidas Android, iOS, del lado del servidor, JVM, del lado del cliente, JavaScript, CSS y nativo.

Las bibliotecas multiplataforma proporcionan las API necesarias para el código Kotlin común y ayudan a desarrollar partes compartidas de un proyecto en el código Kotlin una vez y compartirlo con todas las plataformas de destino.

¿Por qué Kotlin / Native?

Kotlin / Native está diseñado principalmente para permitir la compilación de plataformas donde las máquinas virtuales no son deseables o posibles, por ejemplo, dispositivos integrados o iOS. Resuelve las situaciones en que un desarrollador necesita producir un programa autónomo que no requiera un tiempo de ejecución adicional o una máquina virtual.

Conclusión:

Creo que es una muy buena opción de programación y me agrada mucho la idea de aprender a cómo interactuar con un nuevo lenguaje de programación, además de que nos ofrece numerosas ventajas como el programar una sola vez lo cual me parece un punto bastante notorio y llamativo puesto que de esta manera solo programamos una vez y luego podemos utilizar ese código en diferentes plataformas.

Fuentes de información:

→ <https://medium.com/@diegop88/kotlin-native-pt1-fe67689018bb>

→ <https://kotlinlang.org/docs/reference/native-overview.html>

→ <https://medium.com/@maniakhitoccori/los-prototipos-r%C3%A1pidos-con-flutter-kotlin-native-e646f65f73e7>