



CREA TU PRIMER

PROYECTO ANDROID CON KOTLIN

EN 15 MINUTOS



devexperto.com

Crea tu primer proyecto Android con Kotlin en 15 minutos

¡BIENVENIDO!

Qué bien que hayas decidido empezar a desarrollar Apps con Kotlin, o al menos echarle un vistazo.

Por si no me conoces, soy Antonio Leiva. Llevo 9 años trabajando como desarrollador software, y 5 de ellos especializado en Android.

En los tres últimos años, me he dedicado a estudiar Kotlin y cómo este lenguaje puede simplificar el proceso de desarrollo de Apps en Android.

He escrito el libro que tanto Google como JetBrains recomiendan para aprender Kotlin aplicado al desarrollo de aplicaciones Android.



¿QUÉ VAS A ENCONTRAR EN ESTA GUÍA?

En esta guía vas a ver cómo crear un proyecto Android en menos de 15 minutos (una vez que lo hayas hecho por primera vez, te aseguro que será mucho menos) y, de paso, te mostraré alguna sorpresilla al final :)

Pero vamos al lío. Ya te he hecho perder un par de minutos de esos 15 tan valiosos que me has reservado.

Los pantallazos y menús se corresponden a los de un Mac, pero encontrarás fácilmente los equivalentes en Windows o Linux.

Espero que te sea de ayuda y que empieces desde hoy a crear tus aplicaciones en Kotlin.

¡Gracias!

Antonio

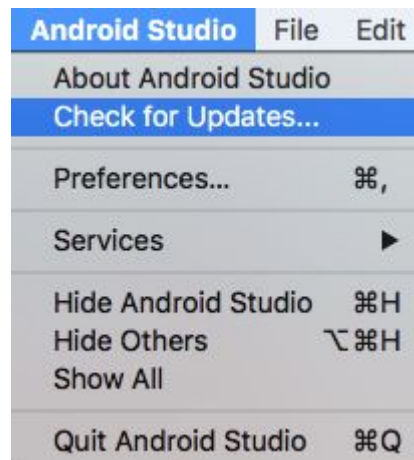
Crea tu primer proyecto Android con Kotlin en 15 minutos

Paso 1: Instala la última versión estable de Android Studio

Es muy posible que esto ya lo tengas. Ten cuidado con las versiones Canary (las que liberan para testeo), porque hay veces que durante unos días Kotlin es inestable en ellas.

La última versión es Android Studio 3.0.

Para comprobar que tienes la última, entra en Android Studio y pulsa en *Android Studio* --> *Check for Updates...*



Comprueba si te ofrece alguna actualización del Android Studio. No está de más que actualices también el SDK de Android si te lo ofrece.

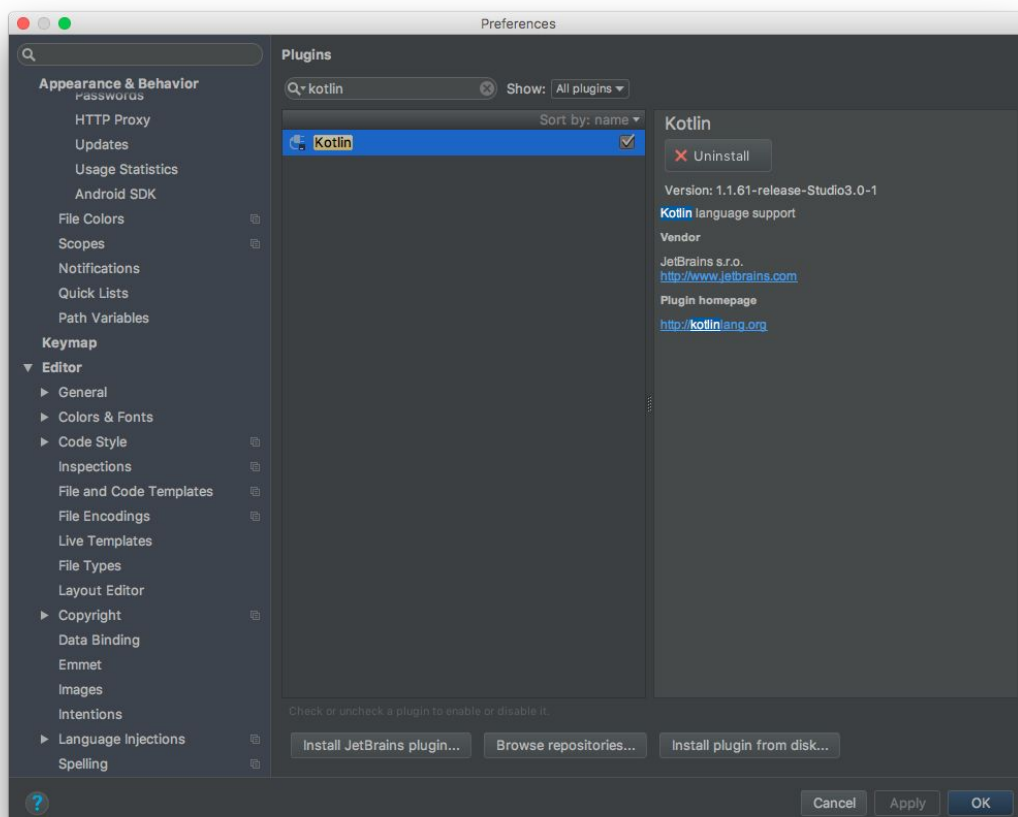
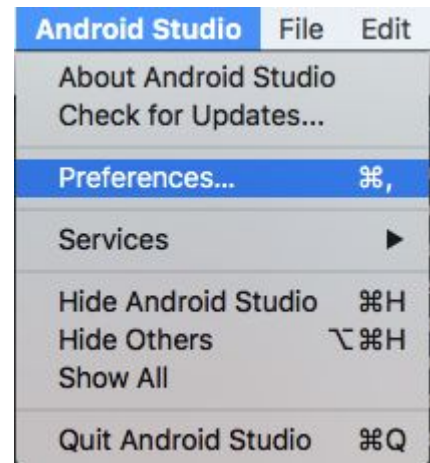
Crea tu primer proyecto Android con Kotlin en 15 minutos

Paso 2: Actualiza el plugin de Kotlin

Android Studio 3.0 ya incluye por defecto el plugin de Kotlin instalado,, así que no tenemos que instalarlo manualmente.

Para asegurarte de que tienes la última versión instalada, ve a la sección de configuración de plugins de Android Studio. Para ello, pulsa en *Android Studio* → *Preferences...*

Después busca *Plugins* en el menú lateral y accede. Escribe “Kotlin” en la caja superior de búsqueda. Comprueba si aparece un botón de “Update” y haz click en él. El mío está actualizado:

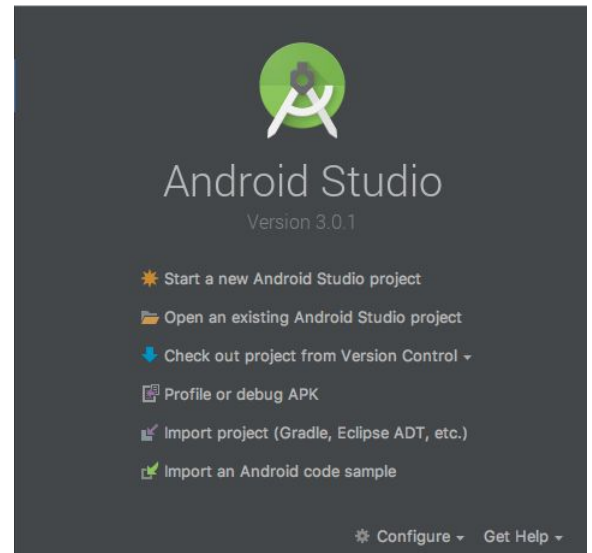


Crea tu primer proyecto Android con Kotlin en 15 minutos

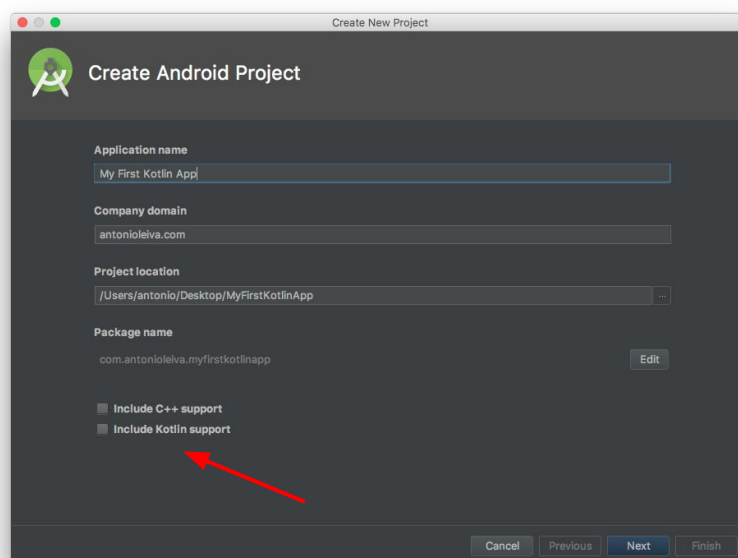
Paso 3: Crea un nuevo proyecto en Android Studio

Es muy sencillo. Como comentaba, desde Android Studio 3.0, Kotlin está totalmente integrado en el entorno, así que puedes crear tus proyectos directamente en Kotlin. No tiene ningún misterio.

Sin embargo, yo quiero enseñarte a configurarlo a mano, que será el caso que tendrás si ya tienes un proyecto en Java y quieres empezar a usar Kotlin en él.



Elige *Start a new Android Studio project*, y te aparecerá la siguiente pantalla:

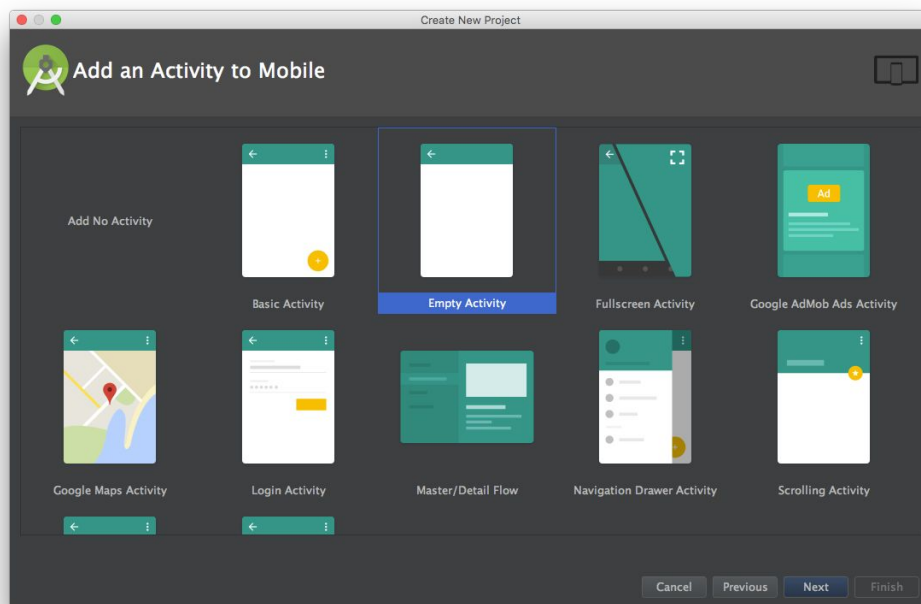


Dale el nombre y la ubicación que quieras. Si ves, abajo del todo puedes marcar *Include Kotlin Support*. Normalmente usarás esta opción, pero déjala desmarcada porque te quiero enseñar algunas cosas. Pulsa *Next*.

Crea tu primer proyecto Android con Kotlin en 15 minutos

En la página *Target Android Devices* elige sólo *Phone and Tablet*, y la versión mínima que prefieras. Yo en mi ejemplo estoy utilizando API 15, aunque a día de hoy es perfectamente válido usar API 19.

En la pantalla de creación de Activity, elige *Empty Activity*:



La última la puedes dejar por defecto. Pulsa *Finish*.

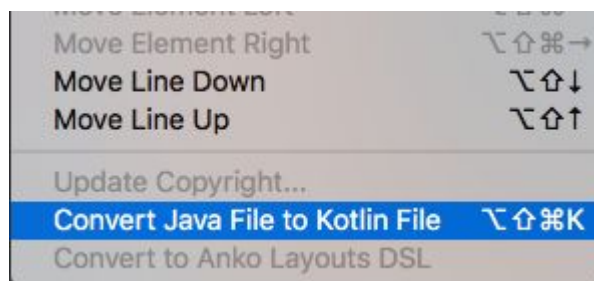
Crea tu primer proyecto Android con Kotlin en 15 minutos

Paso 4: Convierte la MainActivity a Kotlin

La forma más sencilla de convertir un proyecto a Kotlin, es tener un archivo Kotlin dentro del mismo.

Así que vamos a convertir la *MainActivity*. El plugin de Kotlin tiene una funcionalidad para hacerlo por nosotros. Recuérdala porque te vendrá muy bien durante tu aprendizaje del lenguaje.

Selecciona el archivo de *MainActivity* en el árbol de la izquierda. En el menú *Code*, elige la última opción: *Convert Java File to Kotlin File*



Tras un par de segundos, este será el resultado:

```
import android.support.v7.app.AppCompatActivity
import android.os.Bundle

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
    }
}
```

Crea tu primer proyecto Android con Kotlin en 15 minutos

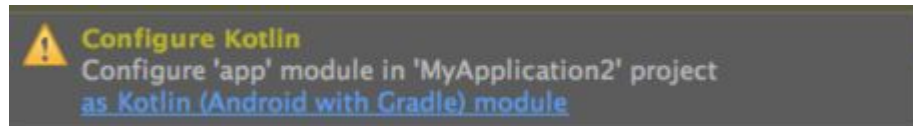
En esta guía no vamos a entrar en las diferencias entre lenguajes a fondo (ya te mandaré algunas cosillas extra para que las aprendas).

Pero verás que, aunque haya alguna cosa diferente, el código sigue siendo perfectamente legible si conoces Java.

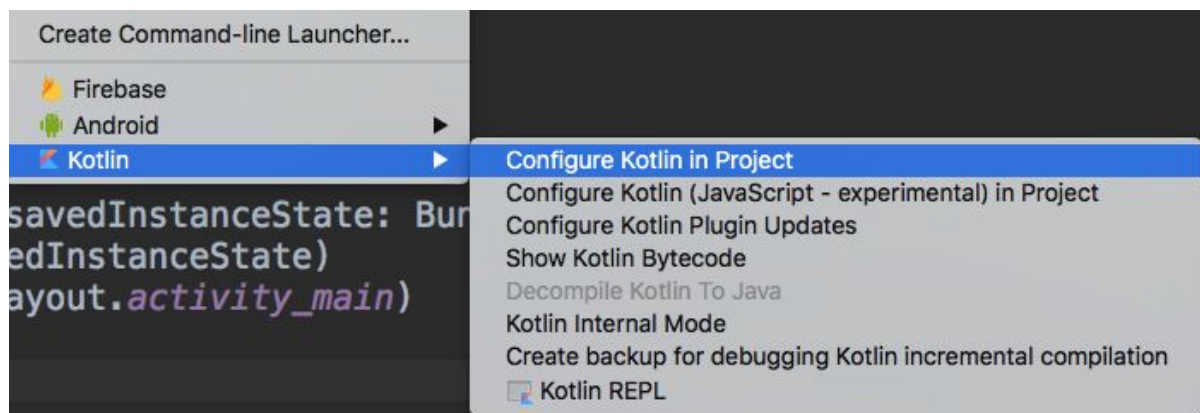
Crea tu primer proyecto Android con Kotlin en 15 minutos

Paso 5: Configura Kotlin en tu proyecto

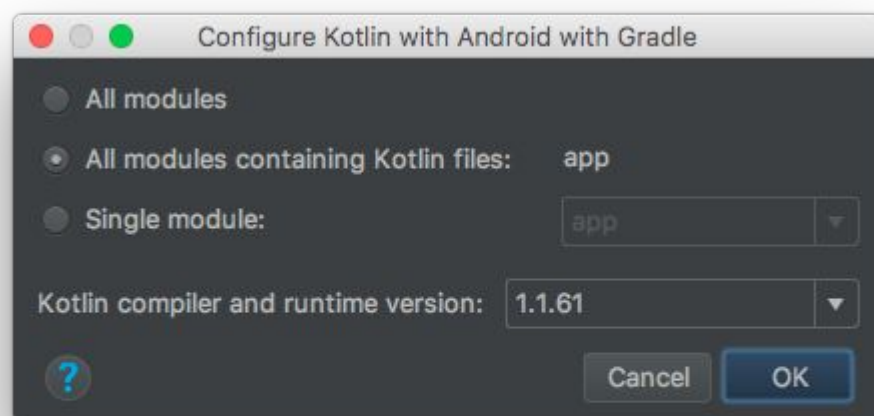
Es posible que ya te haya salido una alerta como esta:



Puedes pulsar ahí. Pero si no, no te preocupes. En el menú *Tools*, encontrarás un submenú llamado *Kotlin*. Selecciona ahí *Configure Kotlin in project*:



Esta es la ventana que te aparecerá:



Crea tu primer proyecto Android con Kotlin en 15 minutos

Aquí tienes todo seleccionado correctamente (asegúrate de que está elegida la última versión del runtime de Kotlin). Al tener un único módulo en el proyecto, las 3 opciones significan lo mismo para nosotros en este caso. Pulsa *OK*.

Esto modificará el *build.gradle* de la carpeta raíz y el del módulo *app*.

Si te fijas, verás que ha añadido la dependencia al plugin de Kotlin en *build.gradle* raíz:

```
buildscript {  
    ext.kotlin_version = '1.1.61'  
  
    repositories {  
        google()  
        jcenter()  
    }  
    dependencies {  
        classpath 'com.android.tools.build:gradle:3.0.1'  
        classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:$kotlin_version"  
  
        // NOTE: Do not place your application dependencies here; they belong  
        // in the individual module build.gradle files  
    }  
}
```

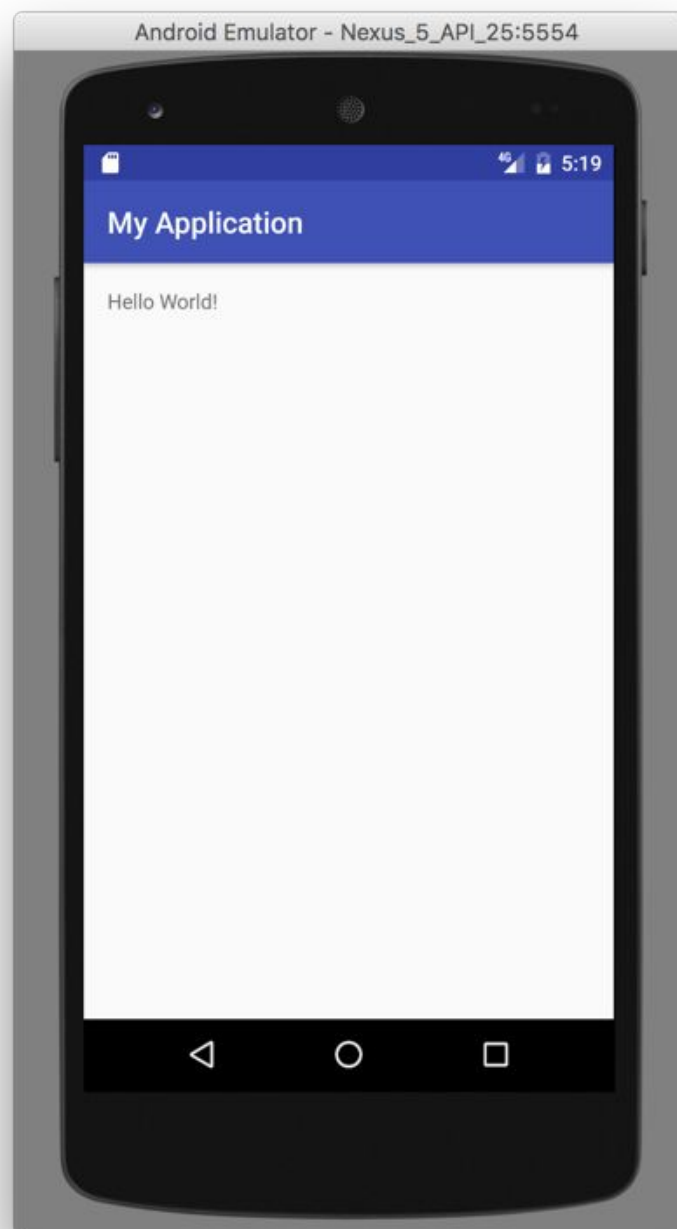
En el de *app*, podrás ver algunos cambios más:

- El *apply* del plugin: *kotlin-android*
- La dependencia de la librería de Kotlin

Crea tu primer proyecto Android con Kotlin en 15 minutos

Paso 6: Ejecuta el proyecto

¡Ya lo tienes todo! Lanza el proyecto y estarás ejecutando tu primera aplicación en Android escrita on Kotlin.



Crea tu primer proyecto Android con Kotlin en 15 minutos

¡Bonus! Paso 7: Interactuando con el layout

No quiero dejarte sin enseñarte algunas cosas interesantes de Kotlin, así que vamos a hacer un layout con un *EditText* y un *Button*.

Cuando pulses en el *Button*, mostrará un toast indicando el valor en el *EditText*.

Para ello, utiliza un XML como el siguiente:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <EditText
        android:id="@+id/input"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:hint="Insert your message"/>

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Click"/>
</LinearLayout>
```

Ahora vas a ver algo sorprendente. Aplica este nuevo plugin en el *build.gradle* del módulo de *app*:

```
apply plugin: 'com.android.application'
apply plugin: 'kotlin-android'
apply plugin: 'kotlin-android-extensions'
```

Crea tu primer proyecto Android con Kotlin en 15 minutos

El plugin *kotlin-android-extensions* te va a permitir acceder a las vistas del XML sin necesidad de recuperar las vistas mediante *findViewById*.

Añade además la librería [Anko](#), a la que te aconsejo que le eches un ojo.

```
implementation "org.jetbrains.kotlin:kotlin-stdlib-jre7:$kotlin_version"
implementation "org.jetbrains.anko:anko-common:0.10.3"
```

Aquí sólo la vamos a usar para utilizar su método *toast*. Es muy sencillo, pero quiero que veas lo que puedes llegar a hacer con Kotlin.

```
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        button.setOnClickListener { toast("Message: ${input.text}") }
    }
}
```

¡Así de sencillo! No necesitas nada más para setear un listener cuando se haga click sobre el botón, y de mostrar un *toast* con el contenido.

Ejecútalo ahora y prueba qué tal funciona.

Crea tu primer proyecto Android con Kotlin en 15 minutos

¡Cuéntame tu opinión!

¿Qué te ha parecido? En poco tiempo has sido capaz de crear un proyecto en Kotlin y de añadir una funcionalidad, que en Java habría requerido de unas 10 líneas, en tan solo una línea.

Esto es una pequeña muestra de lo que Kotlin puede hacer por ti como desarrollador Android.

No te preocupes si no entiendes todo el código del ejemplo. Si quieres seguir aprendiendo, en los próximos días recibirás algunos emails extra en los que te voy a explicar esto y alguna cosa más para que puedas crear aplicaciones completas.

Puedes contarme qué te ha parecido escribiendo a **contacto@devexperto.com**

Si has recibido esta guía por algún otro medio, te recomiendo que te suscribas en devexperto.com para recibir los artículos con el contenido extra.

También puedes seguir todas las novedades en [Twitter](https://twitter.com/devexperto), donde también comparto artículos que considero interesantes.

¡Gracias y nos vemos pronto!

Un abrazo,

Antonio Leiva

devexperto.com