

Instalación del entorno de desarrollo

Para desarrollar aplicaciones Android es necesario el paquete de software *Android SDK*, el cual fue diseñado por Google con el propósito de proporcionarle a los desarrolladores los elementos necesarios para las apps destinadas a dispositivos Android.

En el paquete Android SDK se incluye: conversor de código, depurador, librerías, emuladores, documentación, ejemplos de código, etc. Todas estas herramientas son accesibles desde la línea de comandos.

No obstante, la mayoría de los desarrolladores prefieren utilizar un IDE (entorno de desarrollo integrado). Un IDE agrupa, en un entorno visual, un editor de código con todas las herramientas de desarrollo. Google recomienda utilizar **Android Studio** (basado en el IDE IntelliJ IDEA).

Instalación de la máquina virtual Java

Las aplicaciones Android están escritas en Java, por lo que necesitas instalar un *software* para ejecutar código Java en tu equipo. Este *software* se conoce como máquina virtual Java, entorno de ejecución Java, Java Runtime Environment (JRE) o Java Virtual Machine (JVM).

Es muy posible que ya tengas instalada la máquina virtual Java en tu equipo. Si es así, puedes pasar directamente a uno de los apartados siguientes. En caso de dudas, puedes pasar también al punto siguiente. Al concluirlo te indicará si la versión de la máquina virtual Java es incorrecta. En caso necesario, regresa a este punto para instalar una que sea adecuada.

Para instalar la máquina virtual Java accede a <http://www.java.com/es/download/>, descarga e instala el fichero correspondiente a tu sistema operativo.

Instalación de Android Studio

En la edición de Google I/O 2014 se lanzó la primera versión estable de Android Studio. Se trata de un nuevo entorno de desarrollo para Android basado en el IDE IntelliJ IDEA. Incorpora nuevas características que no encontramos en el tradicional IDE basado en Eclipse. Google va a potenciar este entorno para el desarrollo de aplicaciones, en detrimento del entorno basado en Eclipse, que va a dejar de mantenerse. Entre las novedades introducidas destacamos:

- Construcción de proyectos usando la herramienta Gradle.
- Previsualización simultánea de un *layout* en varios tipos de dispositivos.
- Facilidades para el testeo de código basado en JUnit.
- Integración con herramientas de gestión de versiones (como GitHub).
- Desarrollo en un mismo proyecto de diferentes versiones (como Android Wear, Android TV y Android Auto).



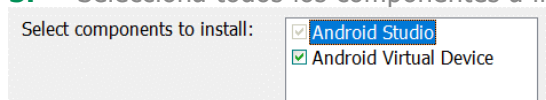
Ejercicio: Instalación de Android Studio

NOTA: Puedes encontrar una descripción más detallada de la instalación en: <https://developer.android.com/studio/install.html>

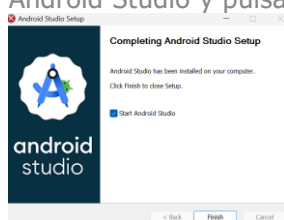
1. Descarga el paquete correspondiente a Android Studio Koala de la siguiente dirección: <http://developer.android.com/sdk/>

2. Ejecuta el fichero obtenido en el paso anterior (android-studio-2024.1.2.12-windows.exe) siguiendo los pasos indicados en <https://developer.android.com/studio/install?hl=es-419>

3. Selecciona todos los componentes a instalar y pulsa *Next*.

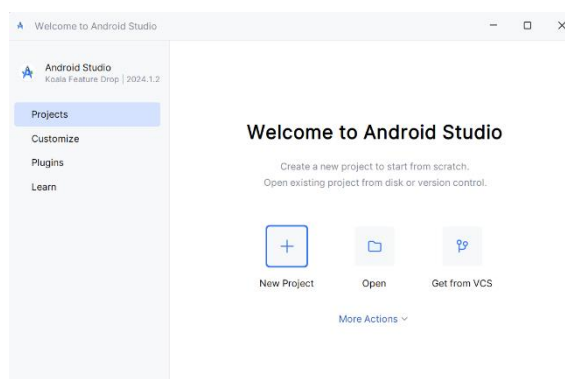


4. Avanza por las diferentes pantallas, hasta acepta el contrato de licencia y selecciona las carpetas donde quieres instalar el IDE Android Studio y el SDK. En el resto de ventanas puedes utilizar las opciones por defecto. En la última ventana indica que quieres arrancar Android Studio y pulsa *Finish*.



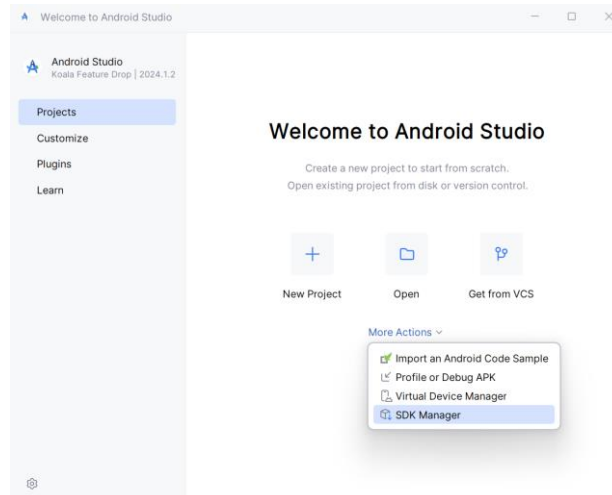
5. Primero te preguntará si quieres importar la configuración desde una instalación anterior. Luego verificará si hay actualizaciones del SDK.

6. Tras pulsar en *Finish* pasamos a la ventana de bienvenida:

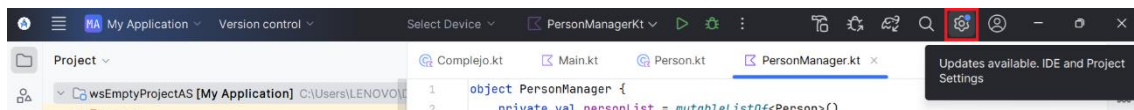


Un aspecto a resaltar es que, cada vez que Google decide realizar una actualización al sistema Android, también modifica el contenido del SDK. Teniendo en cuenta esto, todo desarrollador que trabaje con Android Studio y con el SDK debe estar pendiente para descargar e instalar las oportunas actualizaciones de dicho sistema.

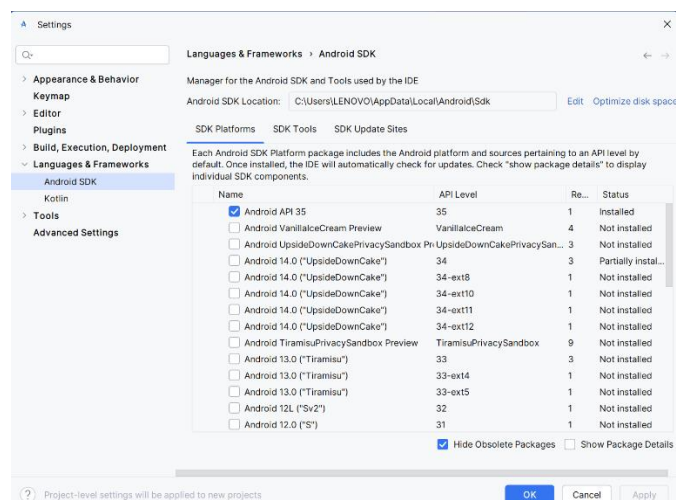
Desde la pantalla de bienvenida, accederemos al SDK Manager, la cual es de gran utilidad para verificar si existen actualizaciones del SDK o nuevas versiones de la plataforma. Podrás acceder a ella desde la ventana de bienvenida de Android Studio pulsando en **More Options/ SDK Manager**.



Si estuviéramos dentro de un proyecto, podríamos entrar al SDK Manager, pulsando sobre la rueda que vemos arriba a la derecha, o en File/**Settings...** y después elegiremos la opción **SDK Manager**.



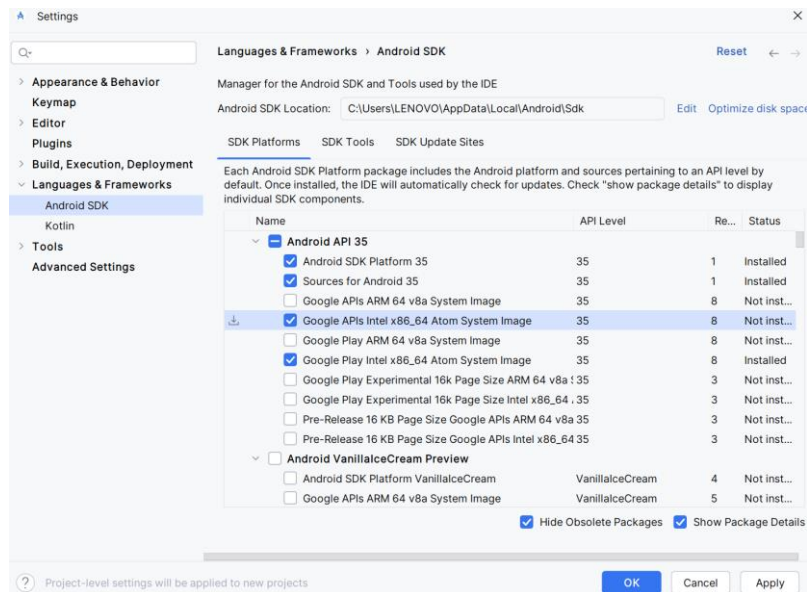
Al entrar en el SDK Manager te muestra los paquetes instalados y los que puedes instalar o actualizar:



En la pestaña SDK Plataforms se muestran los paquetes de plataforma. Pula en *Show Package Details* para ver los diferentes paquetes. Siempre es conveniente que tengas instalados los siguientes paquetes de la última plataforma disponible:

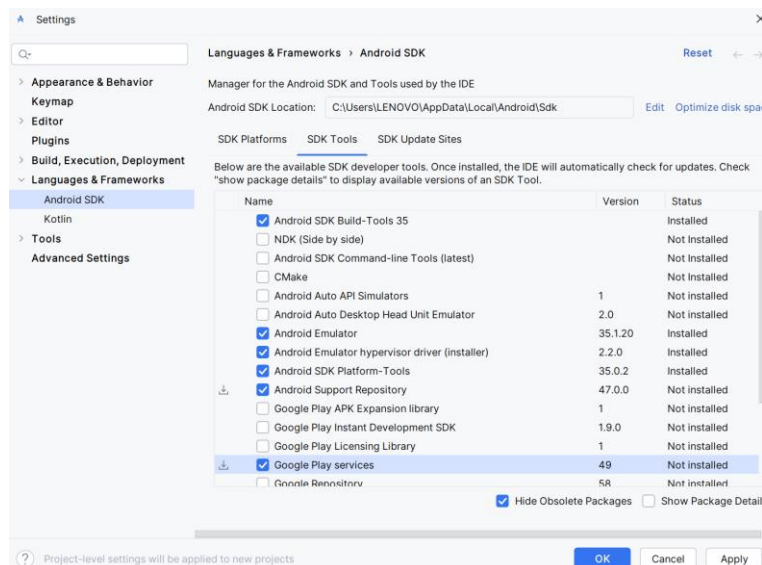
- *Android SDK Platform X (donde X es la última versión disponible)*
- *Sources for Android X (no es imprescindible)*
- *Google APIs ... System Image (para crear emuladores con Google APIs)*
- *Google Play ... System Image (para crear emuladores con Google APIs + Google Play)*

Si falta alguno de los paquetes, seleccionarlo y pulsar en Apply y proceder a la instalación de los paquetes.



En la pestaña SDK Tools se muestran paquetes con herramientas de la plataforma. Siempre es conveniente que tengas actualizados los siguientes paquetes:

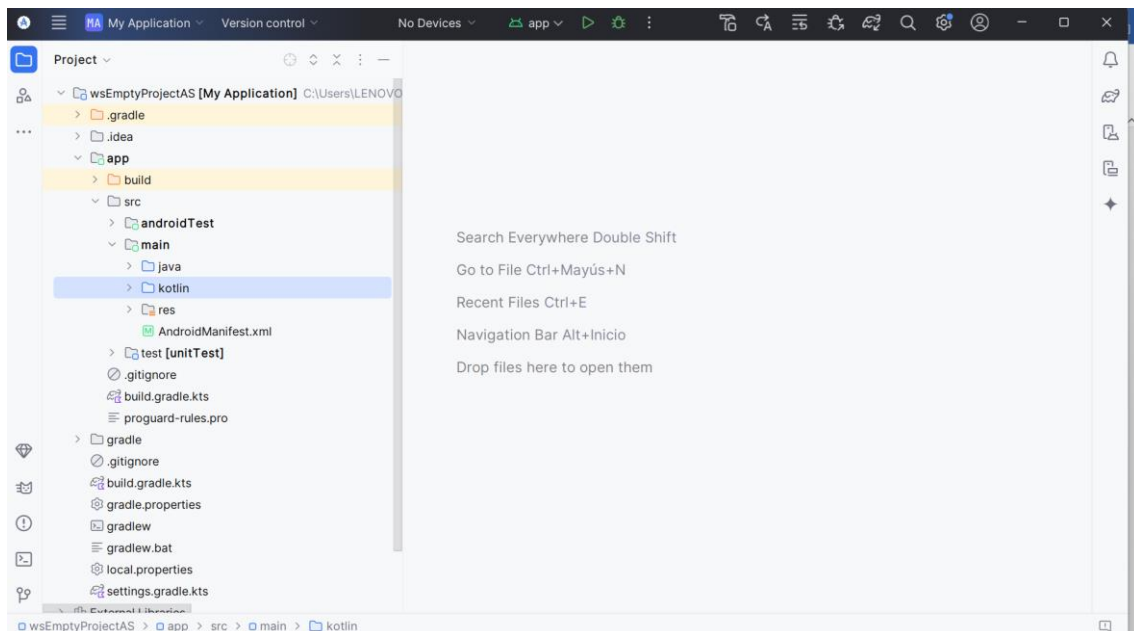
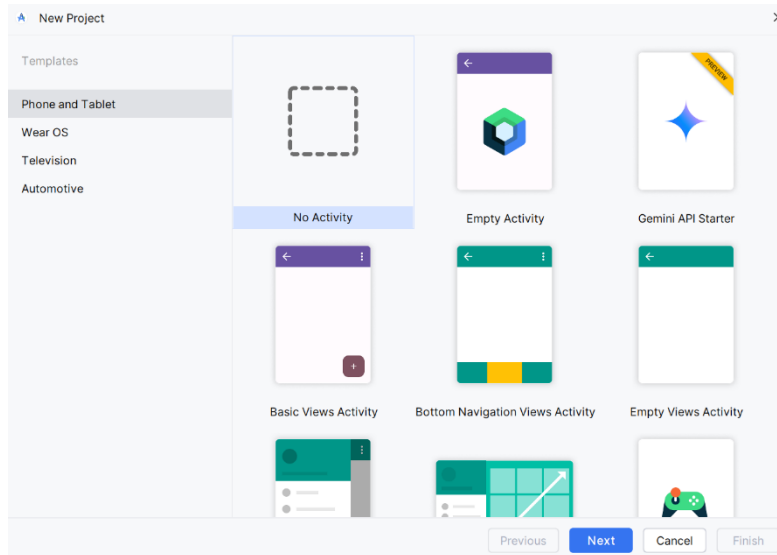
- *Android SDK Build-tools XX*
- *Android SDK Platform-tools XX*
- *Google Play services*
- *Android Support Repository*





Ejercicio: Crear Proyecto Kotlin

Vamos a empezar creando un nuevo proyecto, para lo cual pulsamos en "New Project", tras lo que nos van a aparecer una serie de plantillas, elegiremos la plantilla "No Activity" ya que queremos ejecutar procesos puros Kotlin.





Recursos adicionales:

Teclas de acceso rápido en un Android Studio:

Alt-Intro: Solución rápida (Ej. añade *imports* de las clases no resueltas).
Shift-F10 (Ctrl-R en Mac): Ejecuta el proyecto.
Shift-F9 (Ctrl-D en Mac): Depura el proyecto.
Shift-F6: Cambia el nombre de un identificador.
Ctrl-Alt-L (Option-Command-L en Mac): Formatea automáticamente el código.
Ctrl-Q (F1 en Mac): Muestra documentación del código.
Ctrl-P: Muestra parámetros del método seleccionado.
F4 (Cmd-flecha hacia abajo en Mac): Salta a declaración.
Ctrl-Y (Cmd-Espacio en Mac): Borra línea.
Alt-Insert (Cmd-N en Mac): Inserta método.



Enlaces de interés: [Conoce Android Studio](#)



Preguntas de repaso: [Instalación y entorno de desarrollo](#)