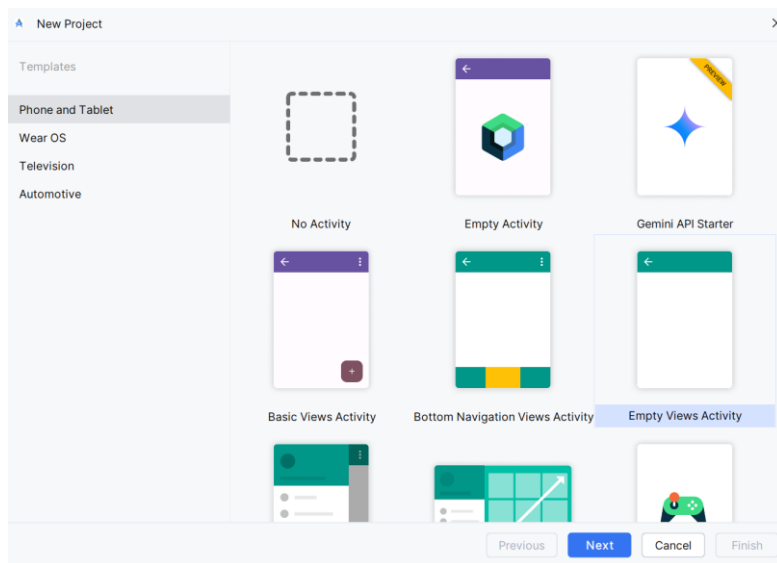




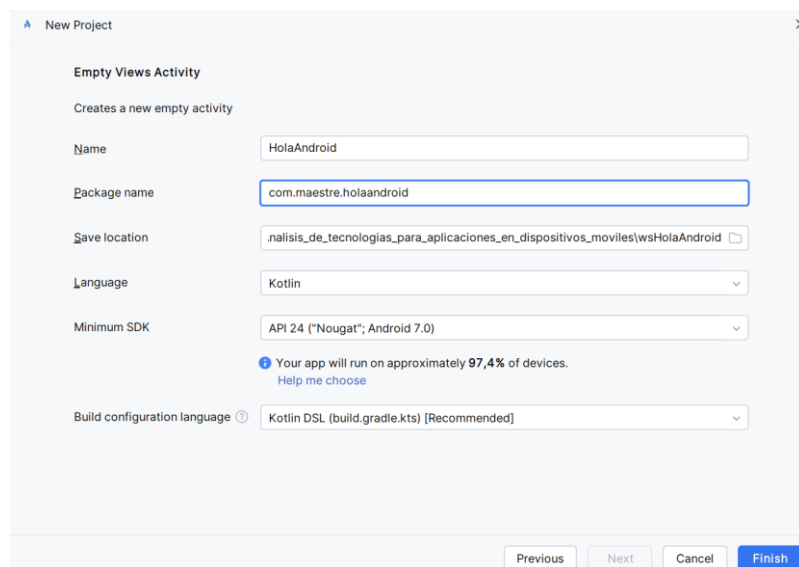
Ejercicio: Crear Proyecto Android

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 con la que queremos comenzar nuestro proyecto, elegiremos "Empty Views Activity" para probar que nuestro entorno funciona y pulsamos Next.

NOTA: Si por el contrario queremos ejecutar procesos puros Kotlin, elegiremos "No Activity".



Después hay que poner el nombre del proyecto y elegir el directorio donde vamos a guardarlo. Rellenamos los campos tal y como se indica y se pulsa Finish.



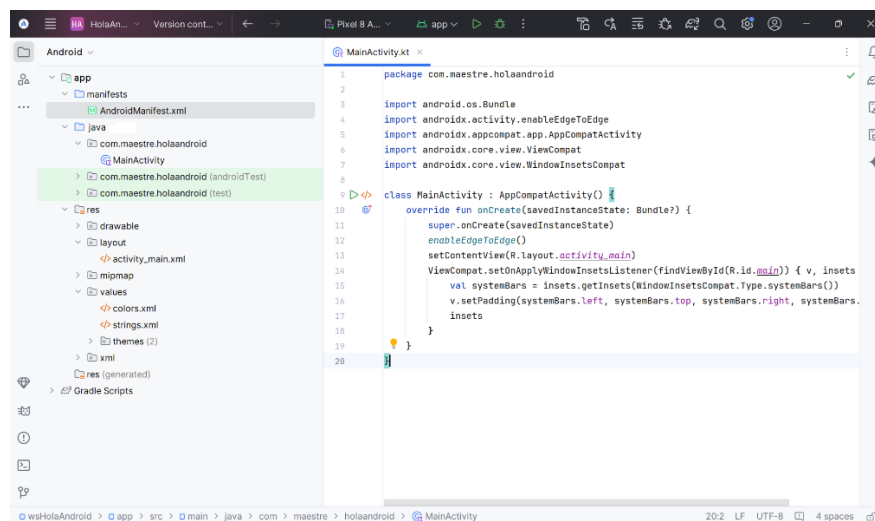
Vemos como quedó nuestro proyecto:

-. **AndroidManifest.xml**: Define la información esencial de la aplicación, como el icono y el nombre de la app, la pantalla o activity principal de la aplicación, los servicios, los permisos y el api objetivo (o target) de la aplicación.

-. **res**: Es la carpeta donde se encuentran los recursos de la aplicación:

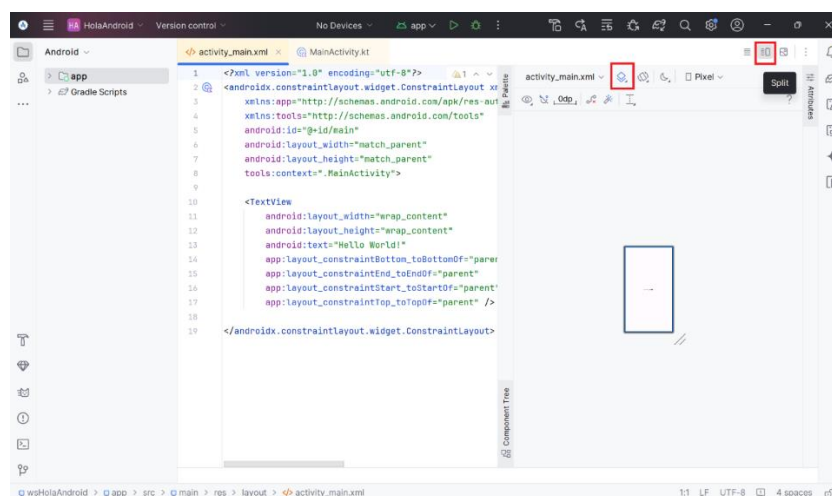
Drawable (imágenes), layout (estructura de las ventanas cuando se hacen en XML, XML que tengan que ver con ventanas), mipmap (imágenes con múltiples resoluciones), values (colores definidos en mi aplicación-colors.xml-, textos de mi aplicación para hacer la aplicación multiidioma -strings.xml-, temas para el estilo de mi aplicación -themes-)

-. **MainActivity.kt**: Es la pantalla principal y donde vamos a programar las funciones en código kotlin con lo que queramos que haga nuestra aplicación. Conectaremos los controles o views e indicaremos que se quiere hacer.



-. **activity_main.xml** es donde vamos a diseñar la interfaz de la pantalla principal (MainActivity.kt) de la aplicación, en el que ahora solo tenemos un contenedor y un texto que dice "Hello World".

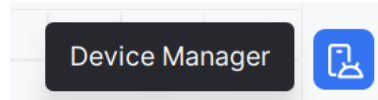
En la vista de diseño, podemos elegir que la superficie de diseño, sea solo "Design" en lugar de como aparece por defecto en "Design & Blueprint". Nos aseguraremos de dejar la vista configurada con la opción Split, para ver el código xml y la vista de diseño a la vez.



Antes de ejecutar nuestra aplicación, tenemos que **crear un Dispositivo Virtual Android (AVD)** en Android Studio.

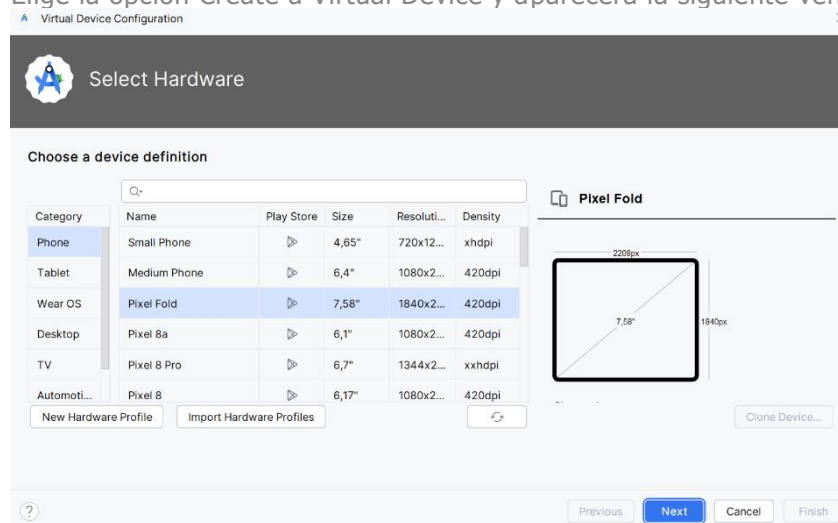
Un dispositivo virtual Android (AVD) te va a permitir emular en tu ordenador diferentes tipos de dispositivos basados en Android. De esta forma podrás probar tus aplicaciones en una gran variedad de teléfonos, tabletas, relojes o TV con cualquier versión de Android, tamaño de pantalla o tipo de entrada.

1. Abrir la vista de **Device Manager**.



Aparecerá la lista con los AVD creados. La primera vez estará vacía.

2. Pulsa a continuación el botón + o sobre **Add a new device...** para crear un nuevo AVD. Elige la opción **Create a Virtual Device** y aparecerá la siguiente ventana:



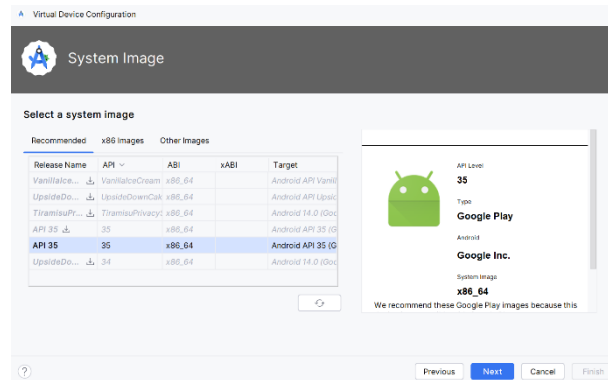
3. En la primera columna podremos seleccionar el tipo de dispositivo a emular (móvil, tableta, dispositivo *wearable* o Google TV). A la derecha, se muestran distintos dispositivos que emulan dispositivos reales. Junto al nombre de cada dispositivo, se indica si tiene la posibilidad de incorporar Google Play, el tamaño de la pantalla en pulgadas, la resolución y el tipo de densidad gráfica.

NOTA: Los tipos de pantalla se clasifican en Android según su densidad gráfica: *ldpi*, *mdpi*, *hdpi*, *xhdpi*, ...

Si quisieras añadir a esta lista crear un nuevo tipo de dispositivo, puedes seleccionar **New Hardware Profile**. Podrás indicar las principales características del dispositivo y ponerle un nombre.

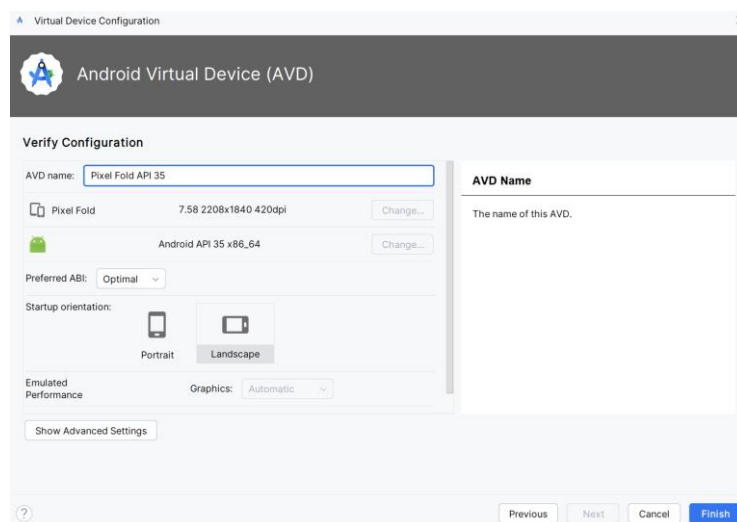
Pulsando con el botón derecho sobre un tipo de dispositivo podrás clonarlos o exportarlos a un fichero. Usando **Clone** podrás crear un nuevo tipo de AVD a partir del actual.

4. Selecciona un tipo de dispositivo y pulsa *Next* para pasar a la siguiente ventana, donde podrás seleccionar la imagen del sistema que tendrá el dispositivo y el tipo de procesador:

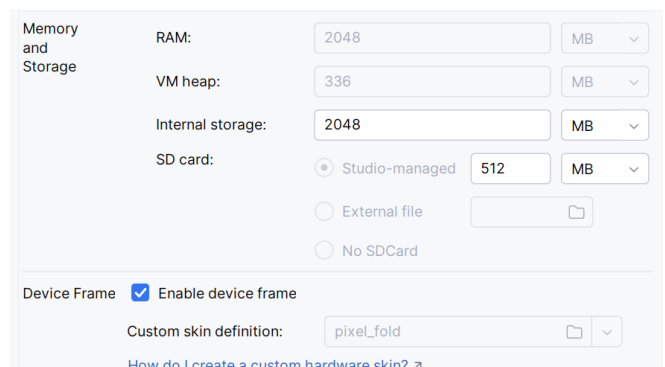
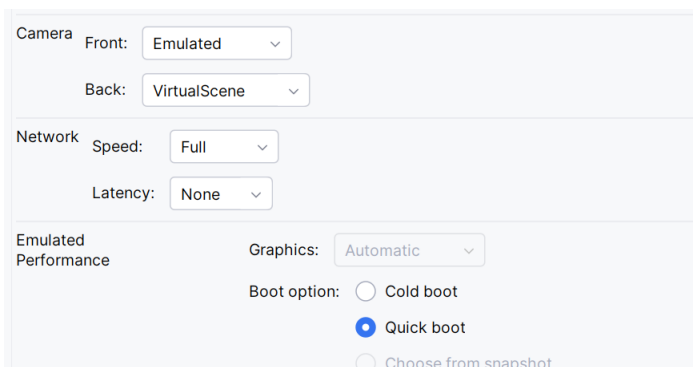


Observa como las distintas versiones de Android que se pueden seleccionar. Selecciona el api actual y pulsa *Next*.

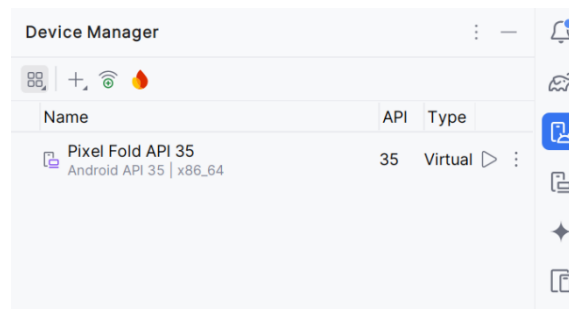
5. En la ventana actual, nos mostrará un resumen con las opciones seleccionadas, además podremos seleccionar la orientación inicial del AVD, si queremos usar el coprocesador gráfico (GPU) de nuestro ordenador o si queremos que dibuje un marco alrededor del emulador simulando un dispositivo real.



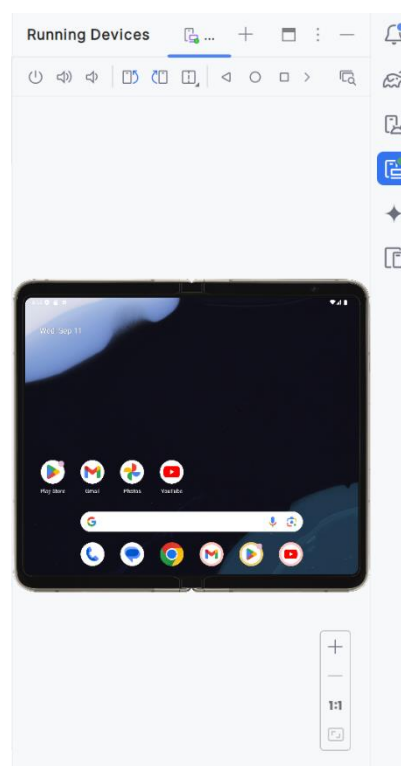
Si pulsas en el botón *Show Advanced Settings* se muestran algunas configuraciones adicionales. Podemos hacer que el emulador utilice la cámara o teclado de nuestro ordenador. Podemos limitar la velocidad y latencia en el acceso a la red. Podemos ajustar la memoria utilizada: RAM total del dispositivo, memoria dinámica usada por Java y memoria para almacenamiento, tanto interna como externa.



6. Una vez introducida la configuración deseada, pulsa el botón *Finish*. Aparecerá el dispositivo creado en la lista:

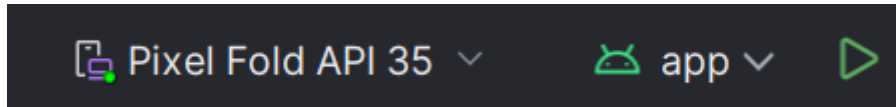


8. Para arrancarlo, pulsa el botón con forma de triángulo verde que encontrarás en la columna de la derecha. Es posible que te pregunte por la entrada de vídeo para emular la cámara del AVD.



NOTA: Algunas características de hardware no están disponibles en el emulador; por ejemplo, el multi-touch o los sensores.

9. Una vez que ya tenemos dispositivo, ya podemos ejecutar nuestra aplicación. Nos fijamos que tenemos seleccionado el dispositivo en el listado y después pulsamos sobre la opción **Run App** pulsando sobre el triángulo.



Tras la instalación de la app en el móvil, podremos ver que se ha ejecutado correctamente en el apartado **Running Devices**:



Como conectar tu dispositivo a Android Studio

<https://developer.android.com/codelabs/basic-android-kotlin-compose-connect-device?hl=es-419#0>



Información sobre cómo administrar tu proyecto Android:

<https://developer.android.com/studio/projects?hl=es-419>



Como configurar tu compilación:

Gradle: Herramienta implementada por el sistema de Android para realizar el proceso de compilación. Ver: <https://developer.android.com/build?hl=es-419>



Interfaz de usuario de Android Studio

<https://developer.android.com/studio/intro/user-interface>