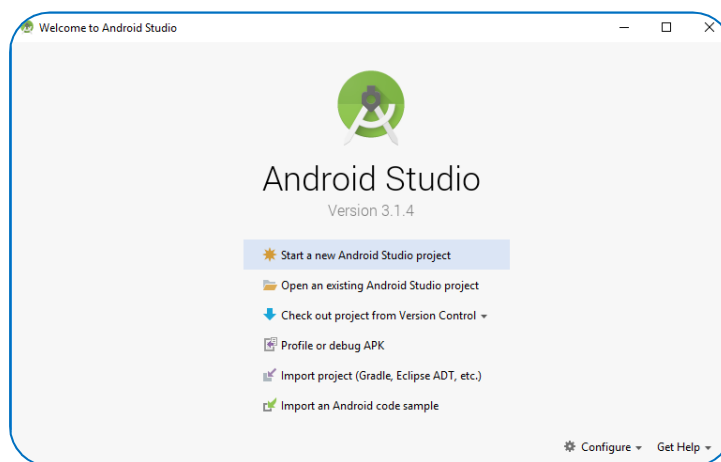


PRIMERA APLICACIÓN ANDROID: “HOLA ANDROID”

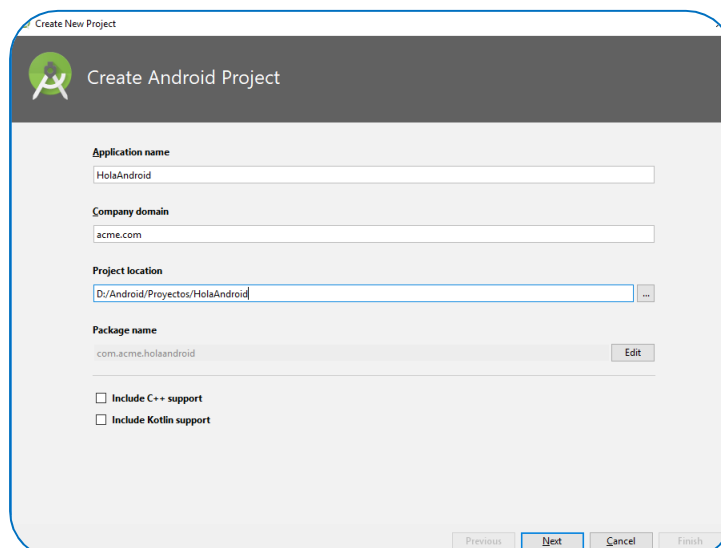
Nuestra primera aplicación Android va a ser una adaptación del famoso “*Hello World*” que sirvió para presentar el lenguaje C allá por 1974: “*Hola Android*”.

Se trata de una aplicación bastante modesta pues se va a limitar a mostrar en pantalla el mensaje: “*Hola Android*”, pero será suficiente para comprobar que nuestro entorno de trabajo está preparado y nuestro terminal virtual funciona correctamente.

Para crear una nueva aplicación pulsaremos sobre “*Start a new Android Studio project*” en la pantalla de bienvenida de Android Studio.

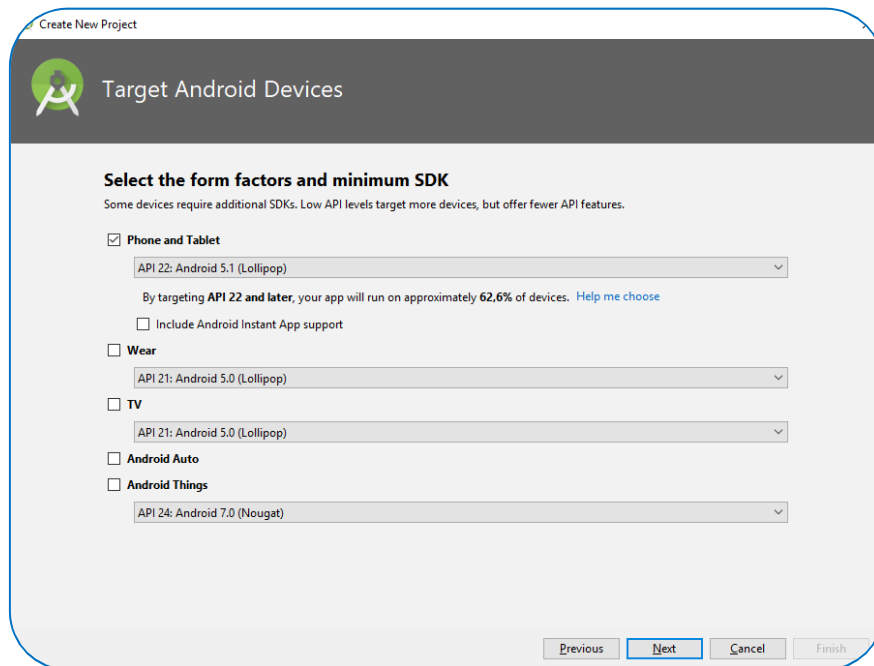


En la siguiente pantalla estableceremos el nombre de la aplicación, el nombre de la compañía y la ruta donde queremos que se cree el proyecto. Si quisiéramos implementar todo o alguna parte de nuestro proyecto en C++ deberíamos marcar la opción “*Include C++ support*”, pero no es el caso.



A continuación, seleccionaremos el dispositivo y la versión mínima del sistema Android requerida para que nuestra aplicación funcione.

Hay que tener en cuenta que tal y como nos advierte un mensaje, cuanto menor sea el nivel de API mayor será el número de dispositivos sobre los que se puede ejecutar nuestra aplicación, pero menor será también el número da características disponibles.



Create New Project

Target Android Devices

Select the form factors and minimum SDK
Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.

☒ **Phone and Tablet**
API 22: Android 5.1 (Lollipop) v
By targeting **API 22 and later**, your app will run on approximately **62.6%** of devices. [Help me choose](#)
☐ Include Android Instant App support

☐ **Wear**
API 21: Android 5.0 (Lollipop) v

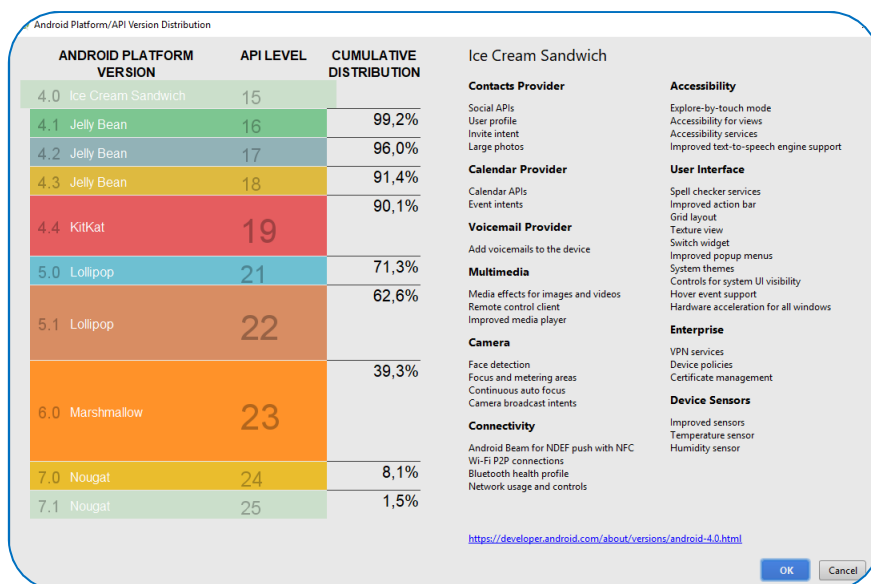
☐ **TV**
API 21: Android 5.0 (Lollipop) v

☐ **Android Auto**

☐ **Android Things**
API 24: Android 7.0 (Nougat) v

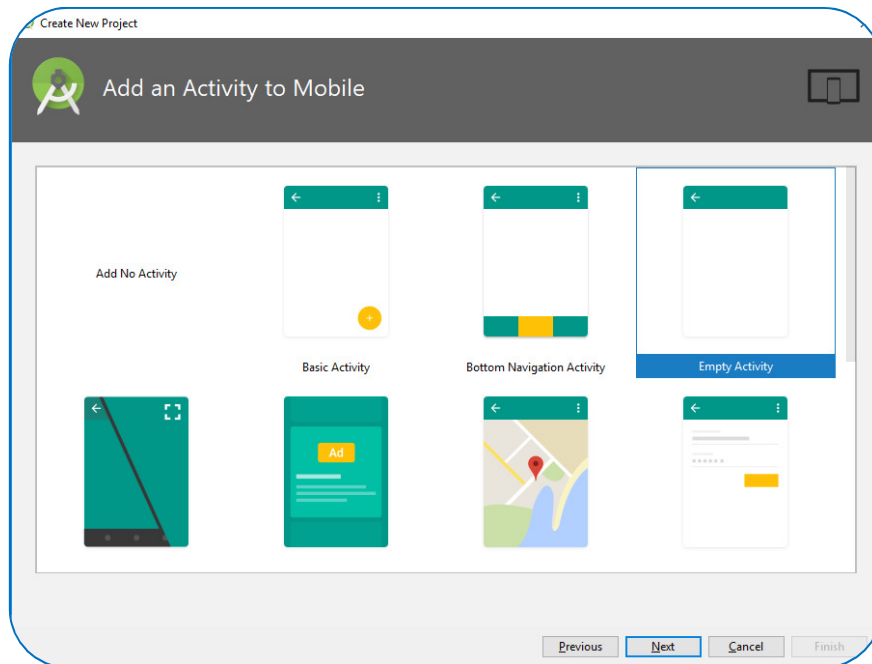
Previous Next Cancel Finish

Si pulsamos sobre el texto “*Help me choose*” se nos mostrará el porcentaje de móviles que podrán ejecutar nuestra aplicación en función del nivel de API seleccionado y las características que podremos utilizar en cada uno de ellos.

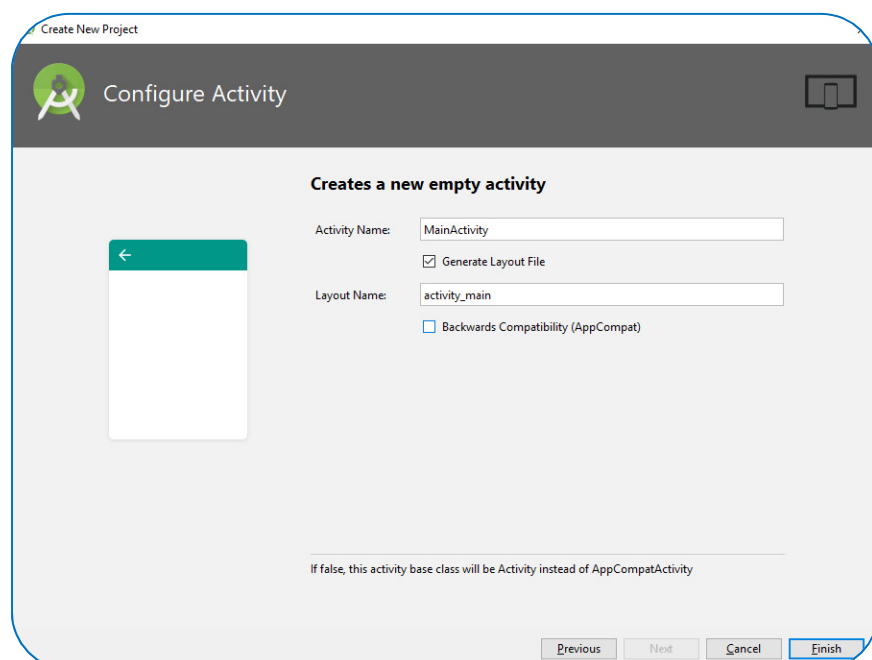


Según esta información con el nivel de API 22 (Android 5.1) podremos alcanzar el 62,6% de los dispositivos móviles Android activos.

En la siguiente pantalla seleccionaremos el tipo de Actividad, variando las opciones disponibles en función del nivel de API seleccionado en el paso anterior. Ya veremos más adelante que son las Actividades y sus diferentes tipos, de momento nos limitaremos a seleccionar una actividad vacía “*Empty Activity*”.

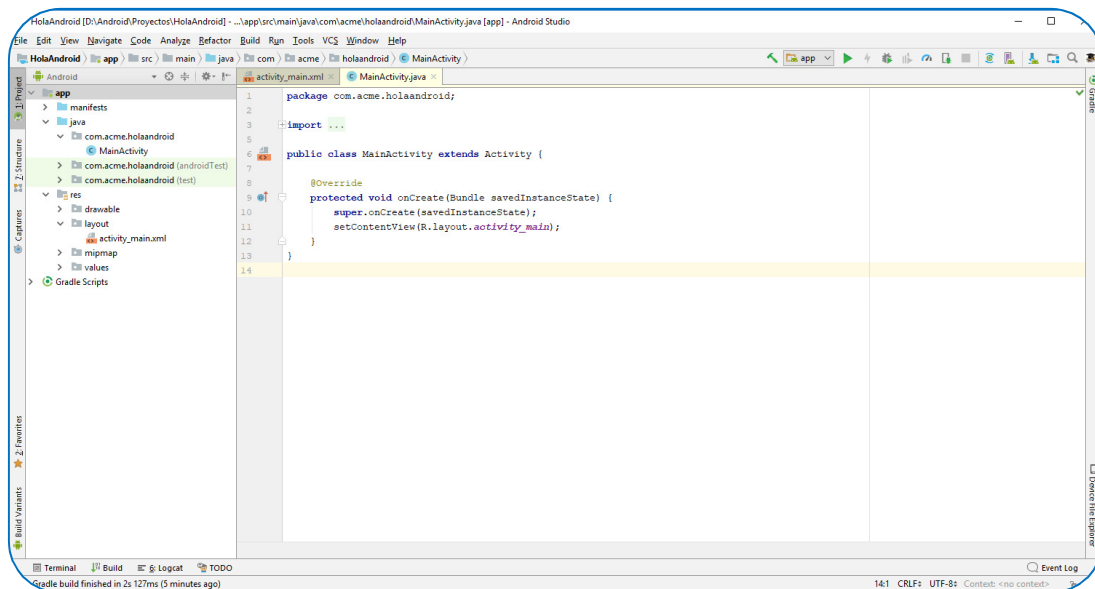


Por último, podemos personalizar el nombre de la actividad (*activity*) y de la pantalla de diseño (*layout*) y desactivar la compatibilidad hacia atrás (AppCompat).

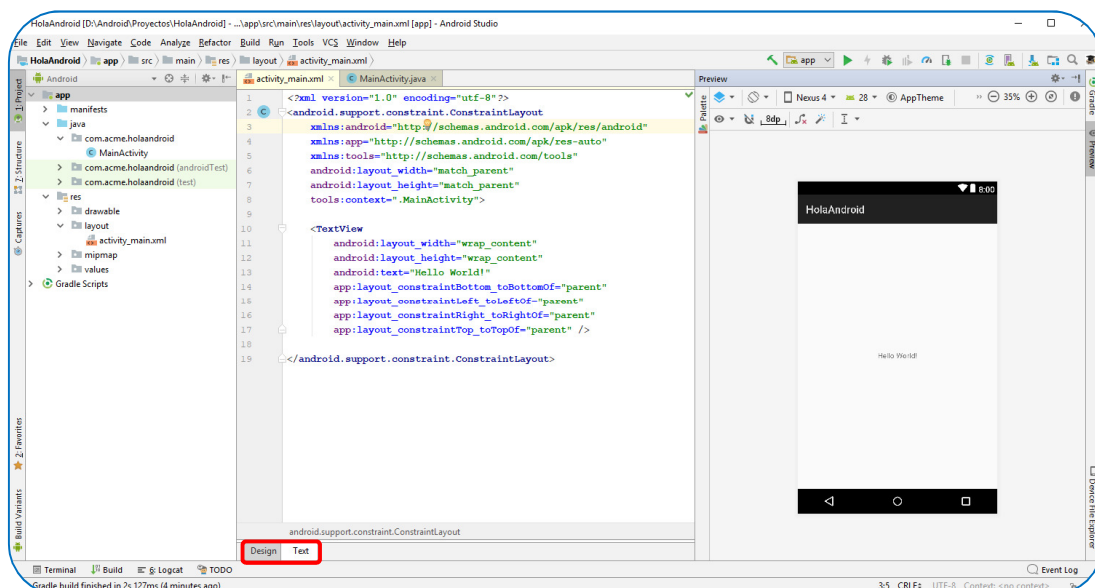


Tras unos instantes tendremos creados nuestro proyecto y se nos mostrarán en pantalla dos archivos:

- **MainActivity.java**: es la actividad principal de nuestra aplicación, tiene un único método “*onCreate*” que se limita a mostrar el diseño de pantalla establecido en el archivo de recursos *activity_main.xml*.

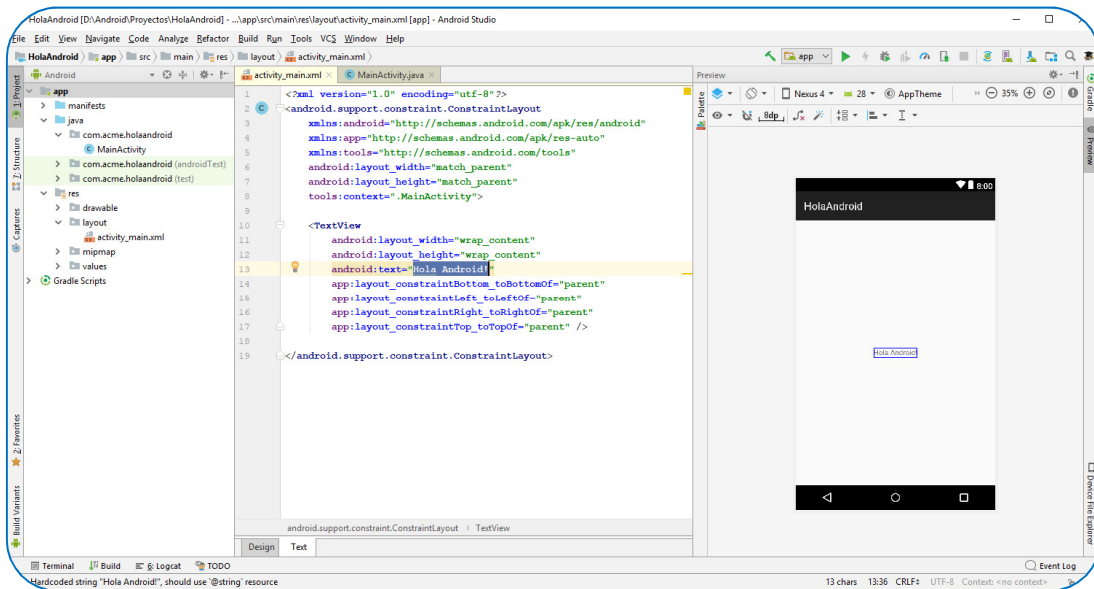


- **activity_main.xml**: es un archivo de recursos, en concreto de diseño de pantalla (*layout*) que establece los elementos que se van a mostrar en pantalla y su disposición. Cuando abramos un archivo de este tipo podremos elegir en la parte inferior entre vista de diseño (paleta de controles) o vista de texto (contenido del archivo).

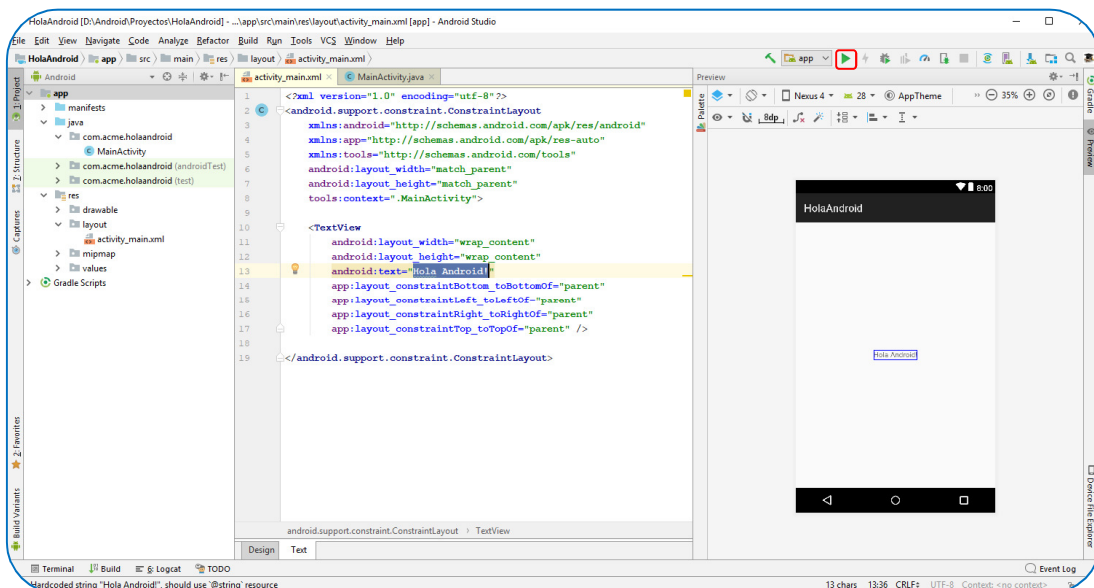


Nos vamos a la vista “Text”, buscamos dentro del elemento “TextView” la propiedad “Android:text” y modificamos su valor por “Hola Android!”.

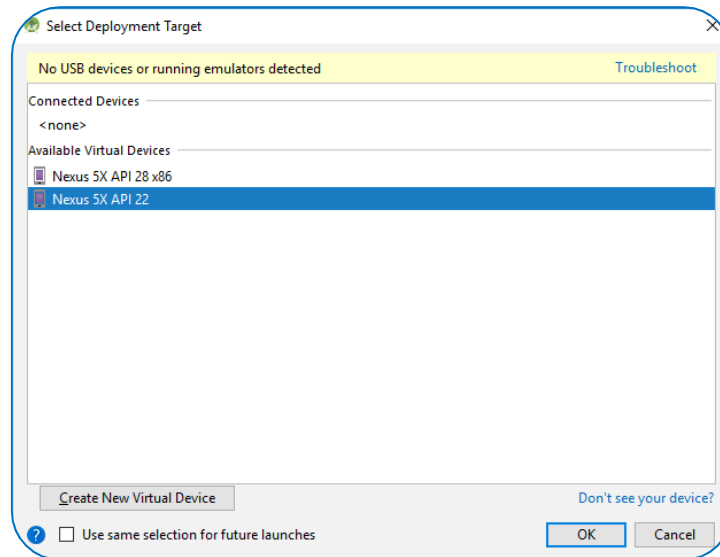
En la parte derecha tenemos una vista previa para ver cómo van a quedar nuestros cambios.



Ya hemos terminado nuestra primera aplicación Android y es el momento de probarla, para ello pulsaremos sobre el botón “Run app” de la barra de herramientas superior.



Se nos mostrará una pantalla donde podremos seleccionar el dispositivo (físico o virtual) sobre el que queremos ejecutar la aplicación. Si no tuviéramos ningún dispositivo disponible podríamos crearlo desde aquí.



Una vez seleccionado el dispositivo virtual se lanzará el emulador y tendrá que iniciar el sistema Android antes de instalar y ejecutar la aplicación.

Este proceso puede ser lento por lo que se recomienda que una vez lanzado el emulador no se cierre para no tener que repetir todo el proceso en futuras ejecuciones.

Además, como se ha comentado, Android Studio incorpora un sistema denominado “*Instant Run*” que permite visualizar los cambios realizados en el código sin tener que recompilar toda la aplicación lo que acelera mucho las siguientes ejecuciones.

Este sería el aspecto de nuestra aplicación...

