



# **Ciclo 4**

## **Fundamentos de programación**

### **Aplicaciones móviles Android con Java**

#### **Sesión 2: Creación de dispositivos virtuales (AVD)**

Programa Ciencias de la Computación e Inteligencia Artificial  
Escuela de Ciencias Exactas e Ingeniería  
Universidad Sergio Arboleda  
Bogotá





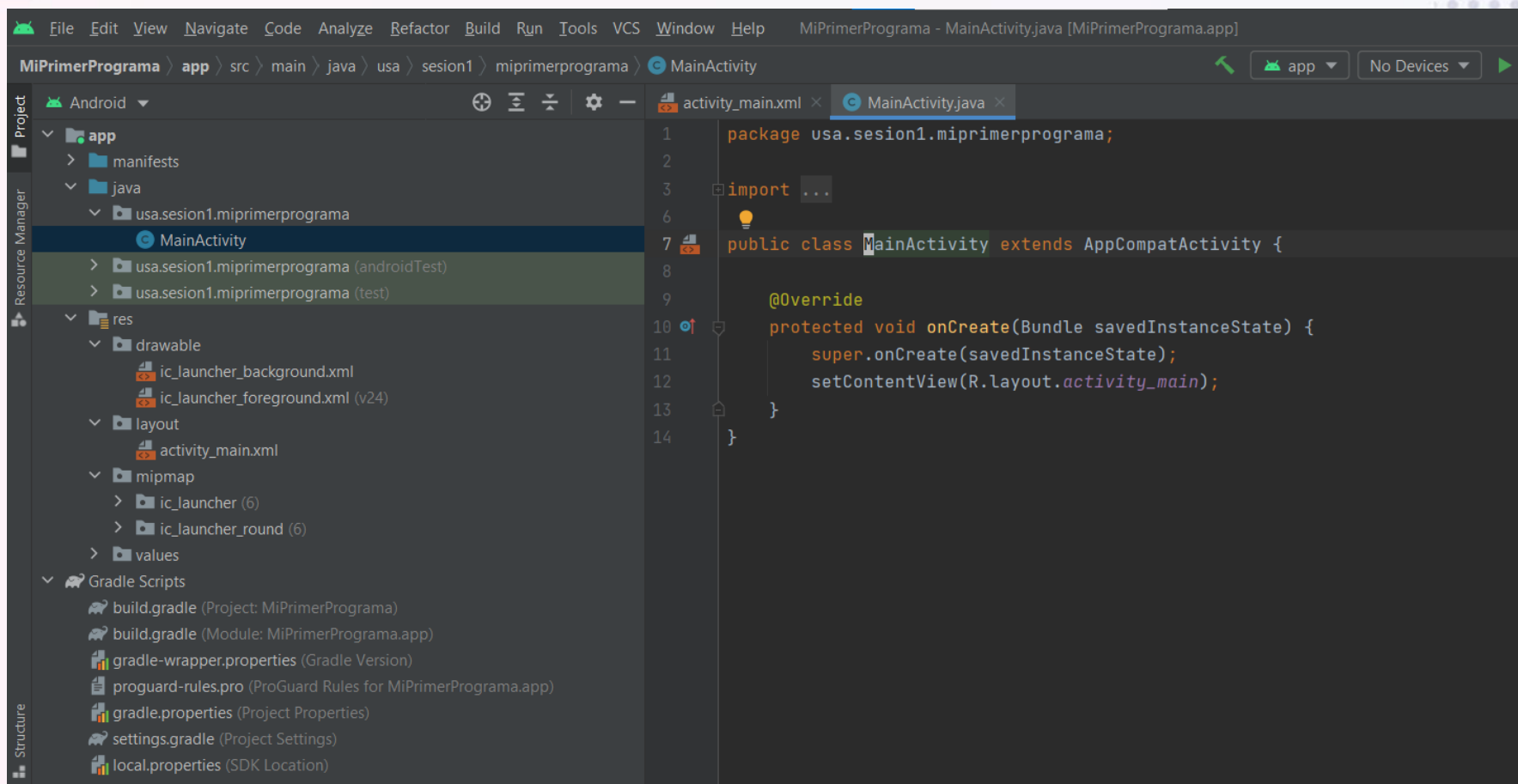
# Agenda

1. Creación y configuración y depuración sobre un emulador (AVD)
2. Configuración y depuración sobre dispositivo real
3. Ejecución de una App.
4. Descarga del APK





# MiPrimerPrograma



Para ejecutar la App: MiPrimerPrograma necesitaremos primero instalar, lo siguiente:

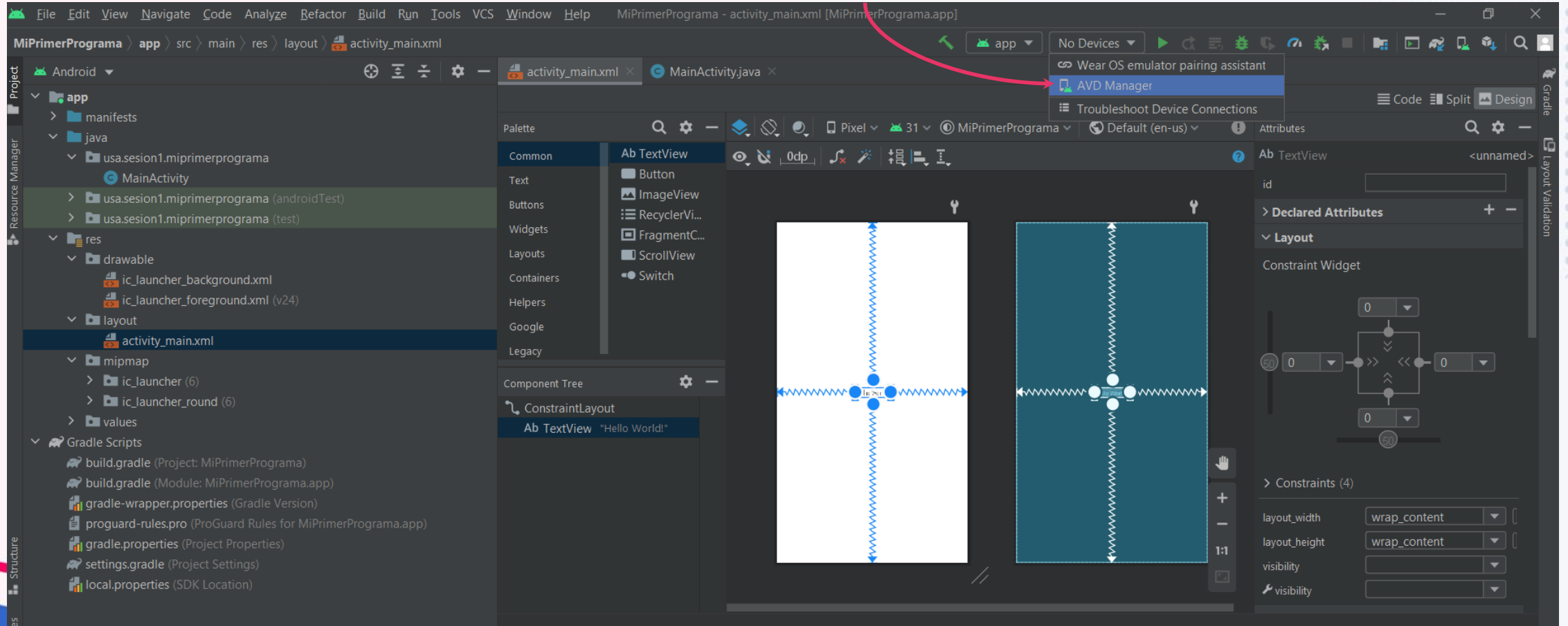
1. AVD Manager.
2. Dispositivo real





# 1. Depuración sobre emulador (AVD)

## AVD Manager







# 1. Depuración sobre emulador (AVD)

Para abrir el Administrador de AVD, realiza una de las siguientes acciones:

1. Selecciona Tools > AVD Manager.
2. En la barra de herramientas, haz clic en AVD Manager



Virtual devices allow you to test your application without having to own the physical devices.

+ Create Virtual Device...

To prioritize which devices to test your application on, visit the [Android Dashboards](#), where you can get up-to-date information on which devices are active in the Android and Google Play ecosystem.

## Xiaomi Redmi Note 5 y Xiaomi Redmi Note 7

Menu Tools → AVD Manager

Clic sobre "Create Virtual Device"

Seleccione "Phone" ubicado a la izquierda

Clic sobre "New Hardware Profile" boton de abajo

Use la siguiente información en la pantalla siguiente:

Device Name: **Xiaomi Redmi Note 7**

Device Type: Phone/ Tablet

Screen Size: 5,99 Inch

Resolution: 1080 x 2160 px

Memory: 4 GB

Input: <Leave these un-selected>

El resto de los campos se mantiene igual.

Seleccione y guarde.

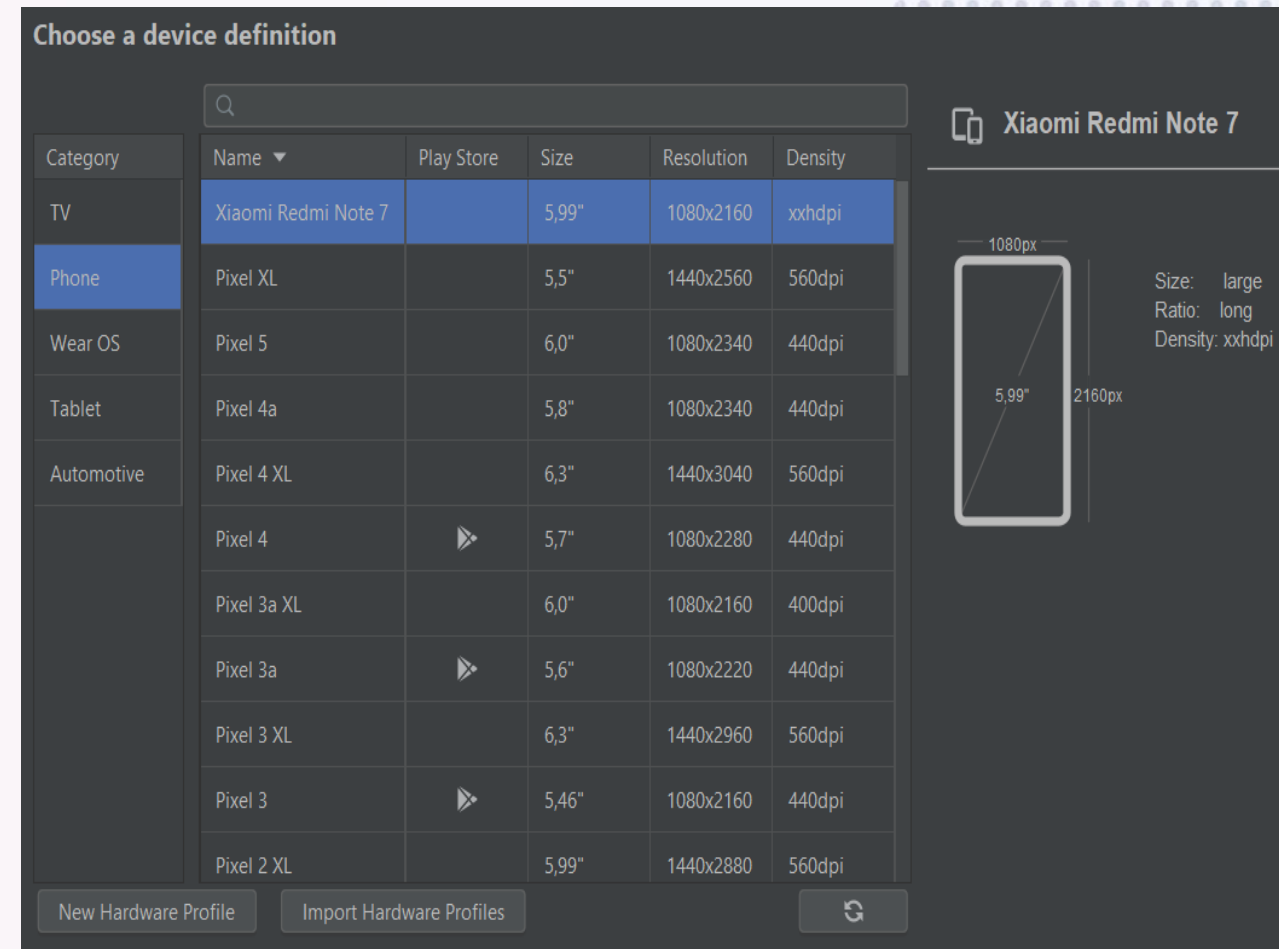
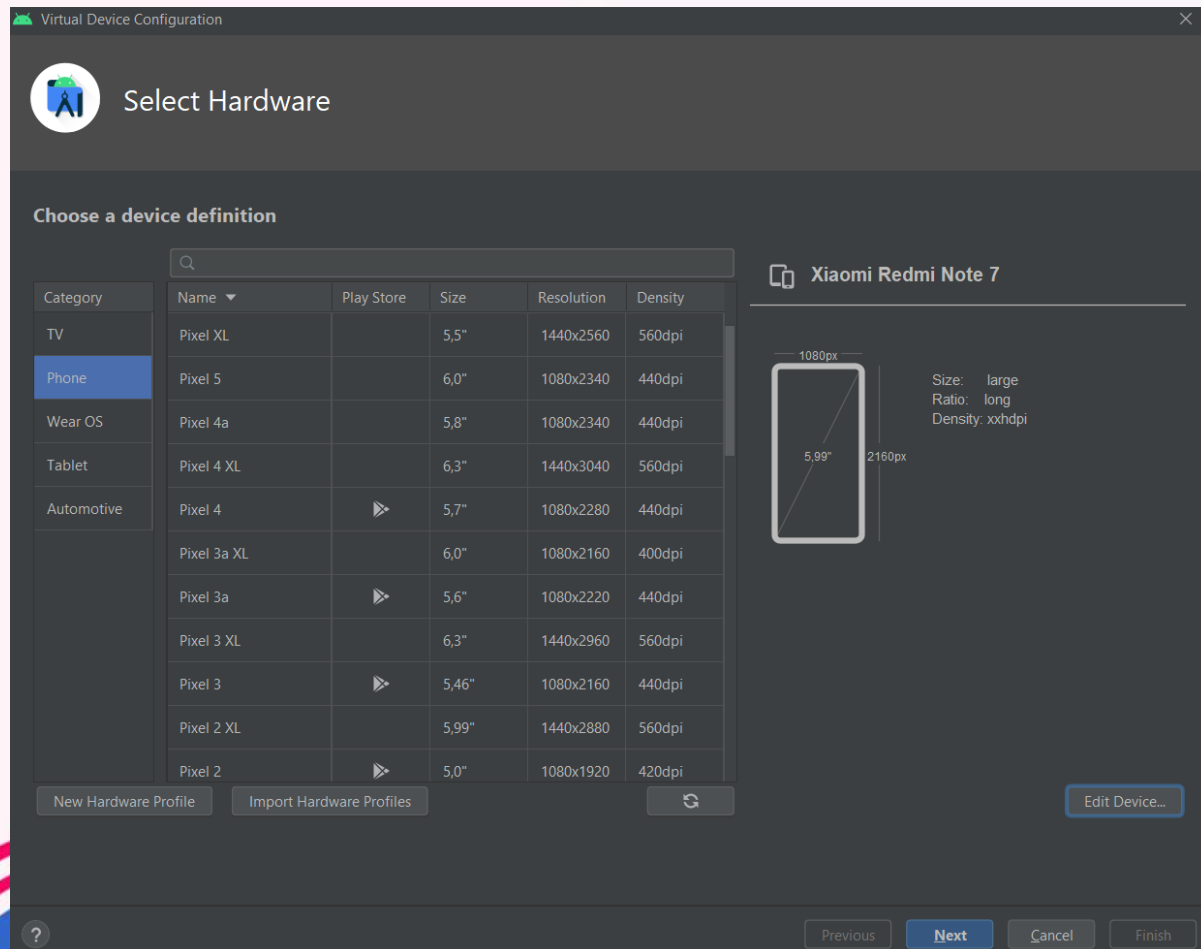
Haga clic en "Siguiente" a continuación para ir a la página siguiente "Seleccione una imagen del sistema". Seleccione la versión que desee. Por ejemplo, Oreo [Destino: Android 8.1]. Haga clic en "Siguiente" para "Verificar configuración."

Nombra el AVD, algo así como " Redmi Note 5 Oreo "





# 1. Depuración sobre emulador (AVD)



Click <<Next>>





# 1. Depuración sobre emulador (AVD)


### Select a system image

Recommended x86 Images Other Images

Release Name	API Level ▼	ABI	Target
<a href="#">R Download</a>	30	x86	Android 11.0 (Google APIs)
<a href="#">Q Download</a>	29	x86	Android 10.0 (Google APIs)
<a href="#">Pie Download</a>	28	x86	Android 9.0 (Google APIs)
<b>Oreo</b>	27	x86	Android 8.1 (Google APIs)
<a href="#">Oreo Download</a>	26	x86	Android 8.0 (Google APIs)
<a href="#">Nougat Download</a>	25	x86	Android 7.1.1 (Google APIs)
<a href="#">Nougat Download</a>	24	x86	Android 7.0 (Google APIs)
<a href="#">Marshmallow Download</a>	23	x86	Android 6.0 (Google APIs)
<a href="#">Lollipop Download</a>	22	x86	Android 5.1 (Google APIs)

Questions on API level?  
[See the API level distribution chart](#)

### Oreo



API Level  
**27**

Android  
**8.1**


Google Inc.


System Image  
**x86**



We recommend these images because they run the fastest and support Google APIs.

### Verify Configuration

AVD Name

 Xiaomi Redmi Note 7 5.99 1080x2160 xxhdpi [Change...](#)

 Oreo Android 8.1 x86 [Change...](#)

Startup orientation  

Emulated Performance Graphics:

Device Frame ☐ Enable Device Frame

Click <<**Finish**>>

Click <<**Next**>>






# 1. Depuración sobre emulador (AVD)

Select a system image

Recommended x86 Images Other Images

Release Name	API Level ▼	ABI	Target
<a href="#">R Download</a>	30	x86	Android 11.0 (Google APIs)
<a href="#">Q Download</a>	29	x86	Android 10.0 (Google APIs)
<a href="#">Pie Download</a>	28	x86	Android 9.0 (Google APIs)
<a href="#">Oreo Download</a>	27	x86	Android 8.1 (Google APIs)
<a href="#">Oreo Download</a>	26	x86	Android 8.0 (Google APIs)
<a href="#">Nougat Download</a>	25	x86	Android 7.1.1 (Google APIs)
<a href="#">Nougat Download</a>	24	x86	Android 7.0 (Google APIs)
<a href="#">Marshmallow Download</a>	23	x86	Android 6.0 (Google APIs)
<a href="#">Lollipop Download</a>	22	x86	Android 5.1 (Google APIs)

Oreo



API Level  
**27**

Android  
**8.1**

Google Inc.

System Image  
**x86**

We recommend these images because they run the fastest and support Google APIs.

Questions on API level?  
See the [API level distribution chart](#)

Licenses

▼ android-sdk-license

↓ Google APIs Intel x86 Atom System

Leer...!  
Aceptar Terminos y condiciones

☐ Decline ☒ Accept

Previous **Next**

Click <<**Next**>>

Este proceso puede tardar  
entre 10 a 20 minutos.

Click <<**Finish**>>

Descargar la imagen al emulador  
Click <<**Next**>> para verificar







## 2. Depuración sobre Dispositivo real

Mientras se descarga e instala la imagen del S.O y se continua con la instalación del emulador (AVD), revisaremos la opción de usar nuestro dispositivo móvil para realizar las pruebas directamente sobre el dispositivo. En ese caso se tendrían dos opciones, **el teléfono y el emulador**.

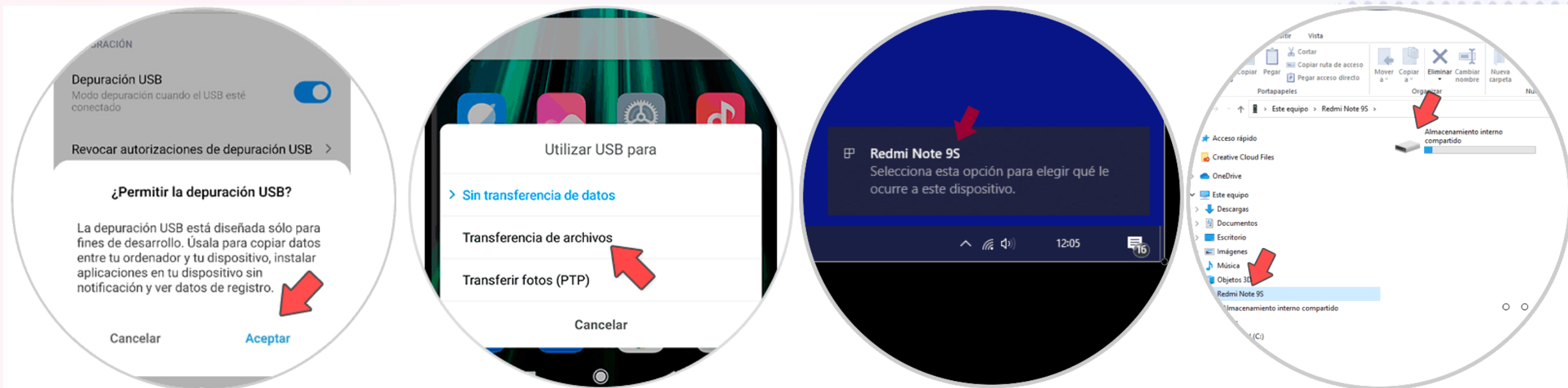
**Opciones de desarrollador:** Cada teléfono o Tablet tiene sus **propio esquema de configuración** como programador, consúltelo en internet o en la ficha técnica de su dispositivo.

Ejemplo: Xiaomi Redmi Note 9s





## 2. Depuración sobre Dispositivo real



Para conectar nuestro teléfono con el PC, debemos usar el cable USB compatible de carga que viene con el teléfono o tablet. Una vez conectado, veremos que en la pantalla del teléfono aparecen diferentes opciones. Para poder gestionar sus datos pulsamos en la opción de “Transferencia de archivos”.

De este manera ya se podrá conectar el teléfono al PC y usar para ejecutar el programa desde el IDE o pasarle el **archivo APK** e instalarlo.

Dependiendo del tipo de dispositivo es posible que requiera **un driver**. Sin embargo, la mayoría es **plug and play**.

<https://www.solvetic.com/tutoriales/article/8396-como-conectar-xiaomi-redmi-note-9s-a-pc-depuracion/>





## 2. Depuración sobre Dispositivo real

### Troubleshoot device connection issues.

This guide helps you troubleshoot issues connecting and running your app on a physical device.

View the device setup instructions in a browser.

- 1 Connect your device over USB**

Verify that your operating system is able to see your device—it may appear as a USB device. Check whether your device appears in the list.

[Rescan devices](#)

✓ Android Studio detected 12 device(s).

**Found 1 Android device(s) with possible problems:**  
eed8165b

  - Device is waiting for you to grant permission for USB debugging

**Found 11 USB device(s) not recognized as Android devices:**  
ADB Interface  
Concentrador USB genérico  
Concentrador raíz USB (USB 3.0) (2x)  
Dispositivo compuesto USB (2x)  
Dispositivo de entrada USB  
HP IR Camera  
HP Wide Vision FHD Camera  
Realtek Bluetooth 4.2 Adapter  
Redmi Note 9S

**Install device drivers.** If you want to connect a device for testing, then you need to install the drivers for the device.

If your computer does not detect your device, disconnect and reconnect your device.

[Next →](#)

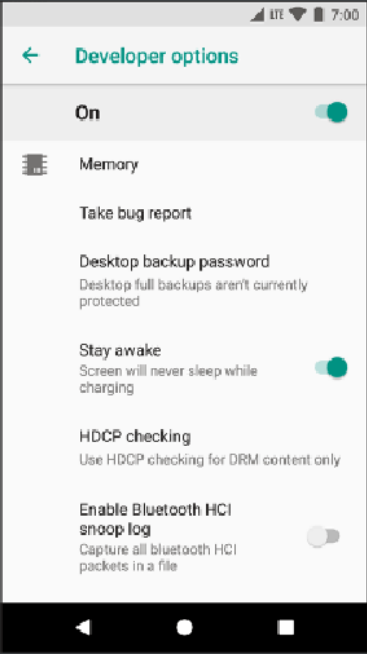
### Assistant Connection Assistant

For ADB to recognize your device as a target for deploying debuggable APKs, you must first enable USB debugging in the on-device developer options.

Depending on the version of Android you're using, proceed as follows:

- On Android 8.0 and higher, go to **Settings > System > About phone** and tap **Build number** seven times.
- On Android 4.2 through 7.1.2, go to **Settings > About phone** and tap **Build number** seven times.

Return to the main **Settings** menu to find **Developer options** at the bottom. In the **Developer options** menu, scroll down and enable **USB debugging**.



[← Previous](#) [Next →](#)

### Assistant Connection Assistant

### Troubleshoot device connection issues.

This guide helps you troubleshoot issues connecting and running your app on a physical device.

View the device setup instructions in a browser.

- 3 Restart the ADB (Android Debug Bridge) server**

If you do not see your device, and you followed the previous steps, click the button below to stop and restart the local ADB server. This also causes ADB to search for new devices.

If your device appears below after restarting the ADB server, try deploying your app to the device. If you continue to experience issues, read about additional steps you can try to resolve USB connection issues.

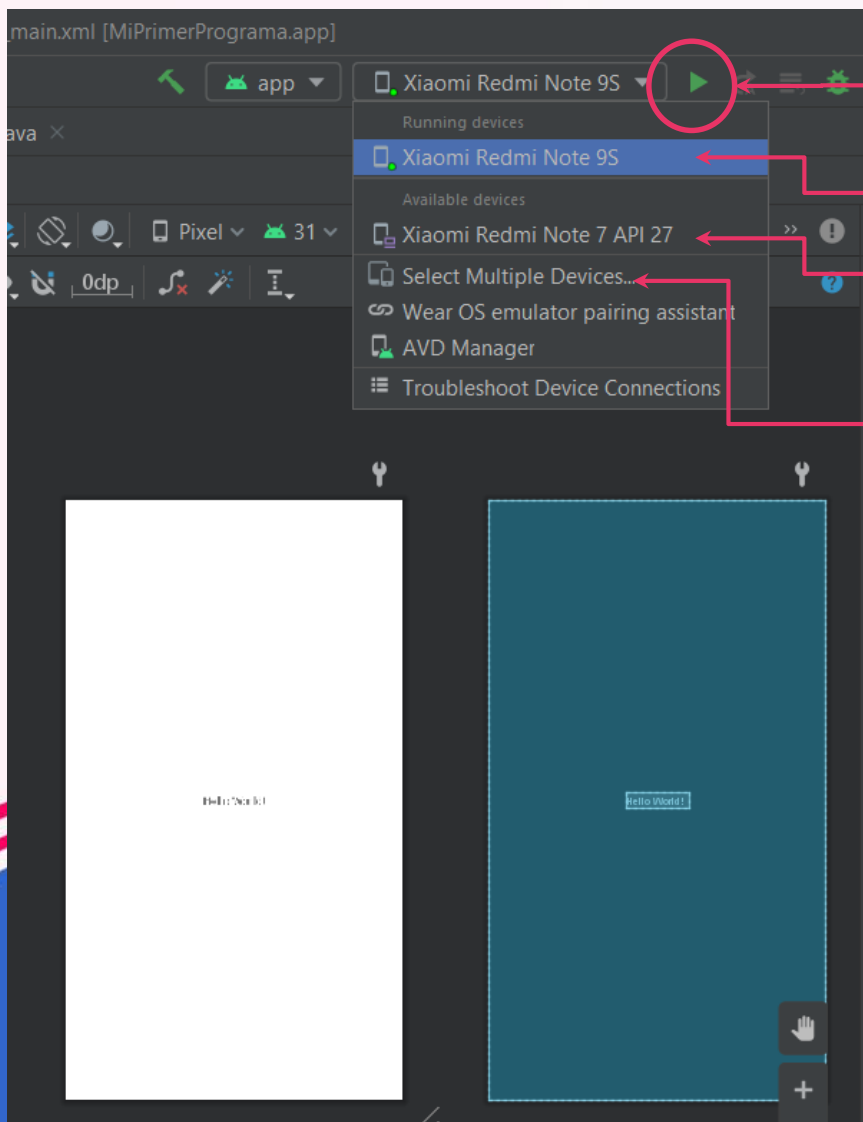
[Restart ADB server](#)

✓ At least one Android device was detected.  
xiaomi-redmi\_note\_9s-ee8165b  
API 29

[← Previous](#) [Finish →](#)



## 2. Depuración sobre Dispositivo real

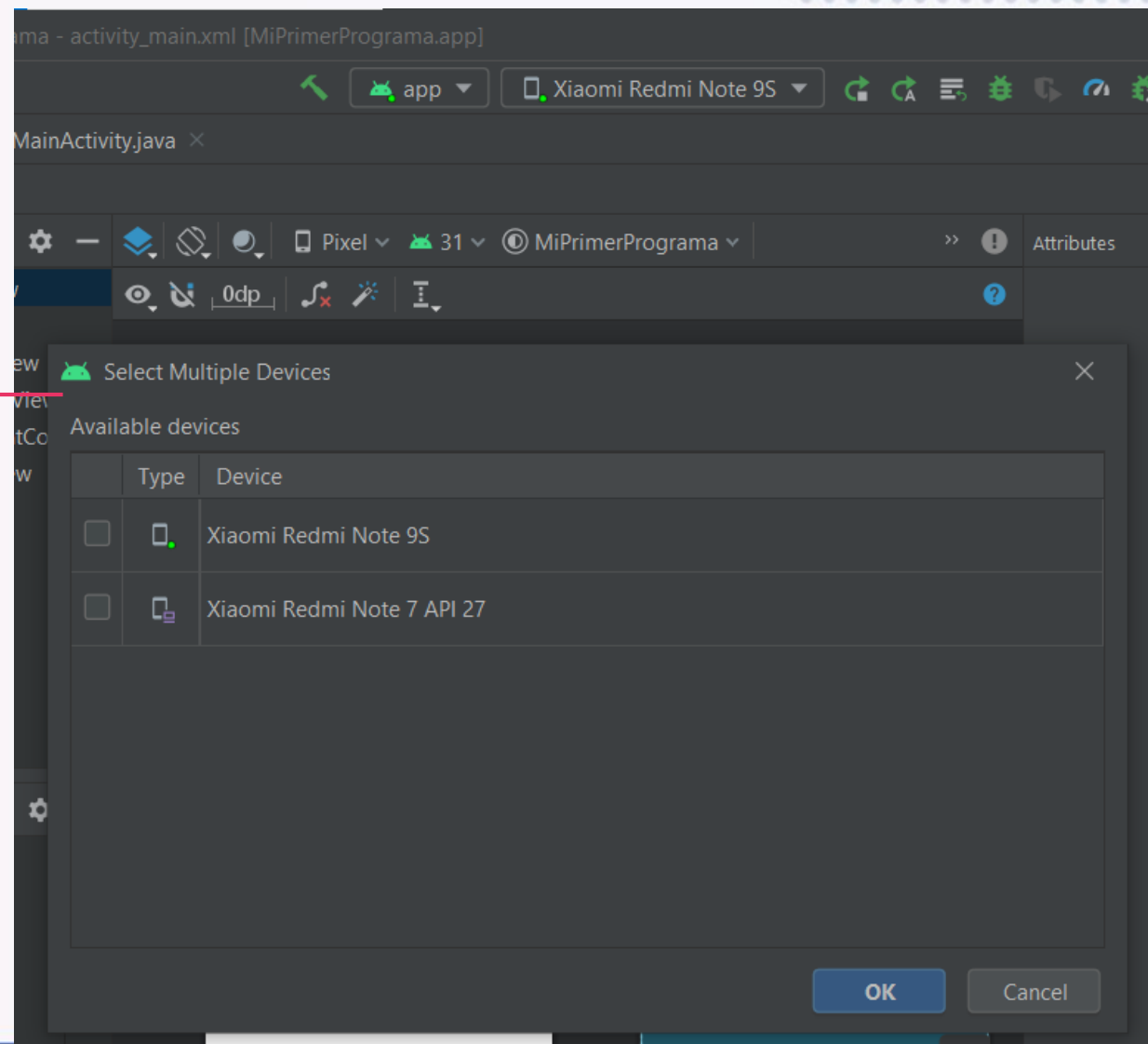


**Ejecutar.**

**AVD Manager.**

**Mobile Device.**

**Multiple Devices.**







El futuro digital  
es de todos

MinTIC

# 3. Ejecución App: MiPrimerPrograma

Type	Name	Play Store	Resolution	API	Target	CPU/ABI	Size on Disk	Actions
	Xiaomi Redmi Note 7 API 27		1080 × 2160: xxhdpi	27	Android 8.1 (Google APIs)	x86	550 MB	

Android Studio interface showing the project structure and the design view of the app.

The interface includes the following components:

- Project View:** Shows the project structure, including the `activity_main.xml` file.
- Design View:** Displays the visual representation of the app's layout, showing a blue background with a white text area.
- Component Tree:** Lists the components used in the layout, including `Ab TextView` and `ConstraintLayout`.
- Attributes Panel:** Shows the attributes for the selected component, including `layout_width` and `layout_height`.

The `activity_main.xml` file is selected in the Project View, and the Design View is active. The Component Tree shows the `Ab TextView` component with the text "Hello World!".

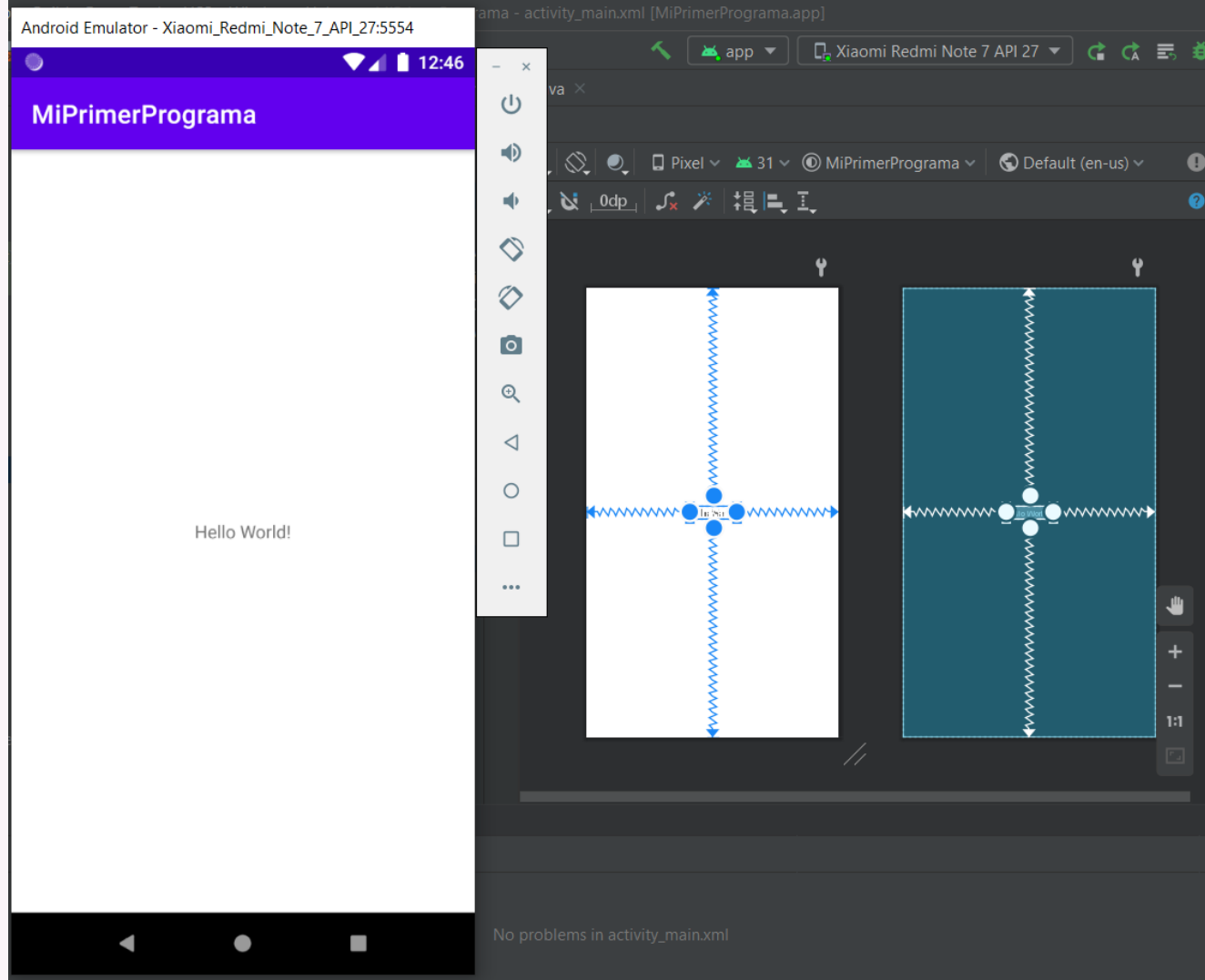
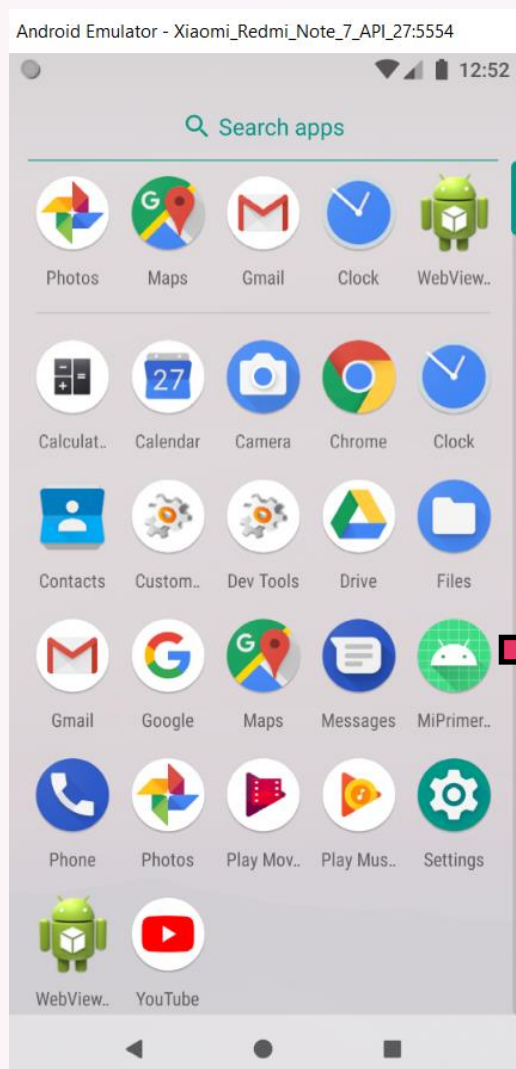
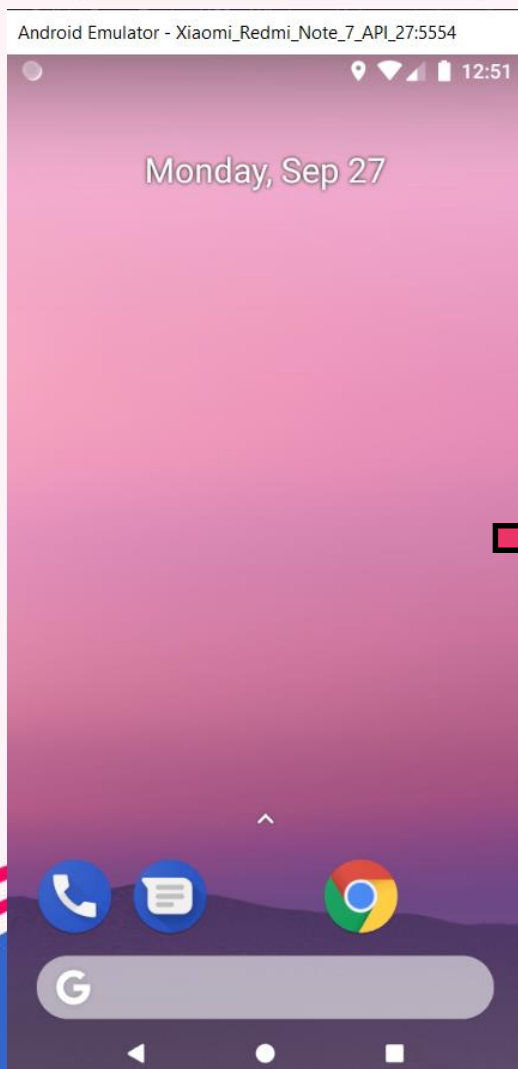
Ejecutar.

AVD Manager.

Mision  
TIC 2022



### 3. Ejecución App: MiPrimerPrograma



El emulador muestra:  
un elemento TextView con el texto "Hello, World!"



# 1. Descarga del APK

**Recomendación:** Procurar iniciar el **IDE Android Studio** antes de comenzar cada clase, ya que según las características de los equipos puede tardar varios minutos en estar listo.

De igual manera después de iniciado, hacer lo mismo con el **emulador (AVD)**.

**El APK:** Es el archivo ejecutable de nuestra aplicación móvil.

Al momento de ejecutar la aplicación (Run), se crea un archivo ejecutable con extensión **APK**. Si se esta usando el emulador (AVD) o directamente un Debug o Debugging sobre el dispositivo móvil este archivo se descarga automáticamente.

Sin embargo, también es posible instalarlo en nuestro dispositivo directamente, buscando el archivo **APK** en la carpeta del directorio **app/build/outputs/apk** de nuestro proyecto. (**Procurar que antes este compilado**)

Pero, primero se debe autorizar el teléfono para permitir esta descarga:

Buscar en configuración y seleccionar la opción: **Aplicaciones** y marcamos la opción **“Origen desconocido (Permitir la instalación de aplicaciones no pertenecientes al mercado)”**.

Se descarga usando el **cable USB** ó enviándolo al **correo** e instalándolo desde allí.



# Preguntas

