**REPORT**
**Codelab 1: ¡Hola ESCOM!**
Subject: Desarrollo de Aplicaciones Moviles Nativas
Group : 7CV1
Student : Julio Cesar Hernández Reyes
Teacher : Velez Saldaña Ulises


---

# 1. Introduction

In this practical i'll learn how to install Android Studio, the Android development environment. Also create and run my first Android app, Hello World, on an emulator and on a physical device.
https://developer.android.com/codelabs/android-training-hello-world#0
What i'll learn:

- How to install and use the Android Studio IDE.
- How to use the development process for building Android apps.
- How to create an Android project from a template.
- How to add log messages to your app for debugging purposes.

# 2. Development

## 2.1. Hola ESCOM

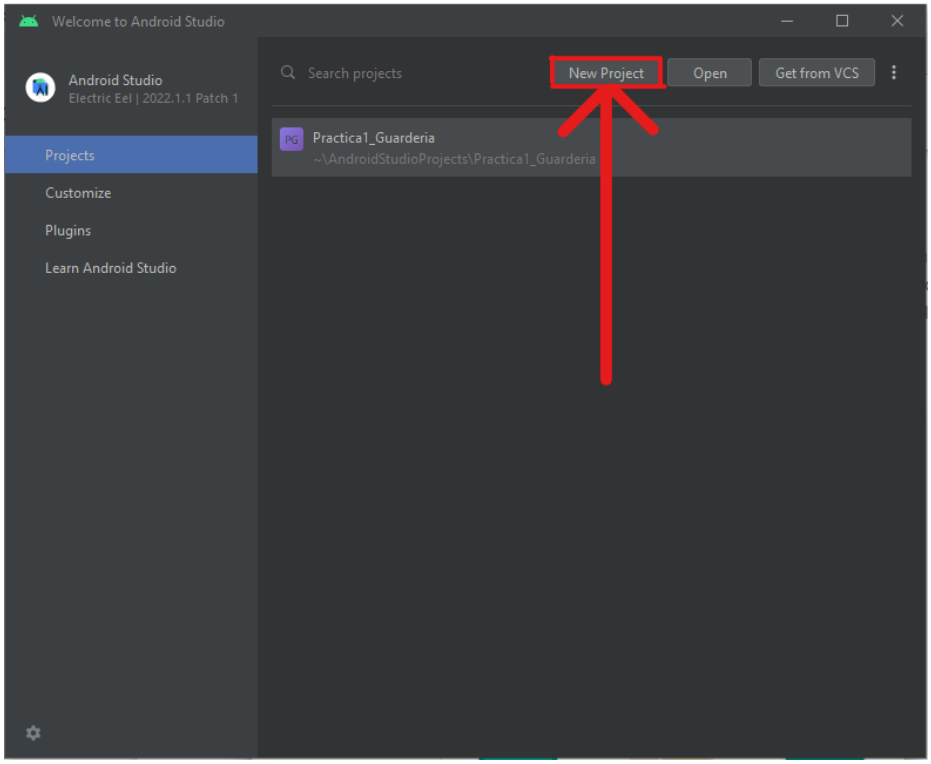
### 2.1.1. Task 1: Install Android Studio

1. I Navigated to the Android developers site and followed the instructions to download and install Android Studio.
2. Then i accepted the default configurations for all steps, and i ensured that all components were selected for installation.
3. After finishing the install, the Setup Wizard downloaded and installed some additional components including the Android SDK.
4. When the download was complete, Android Studio started.

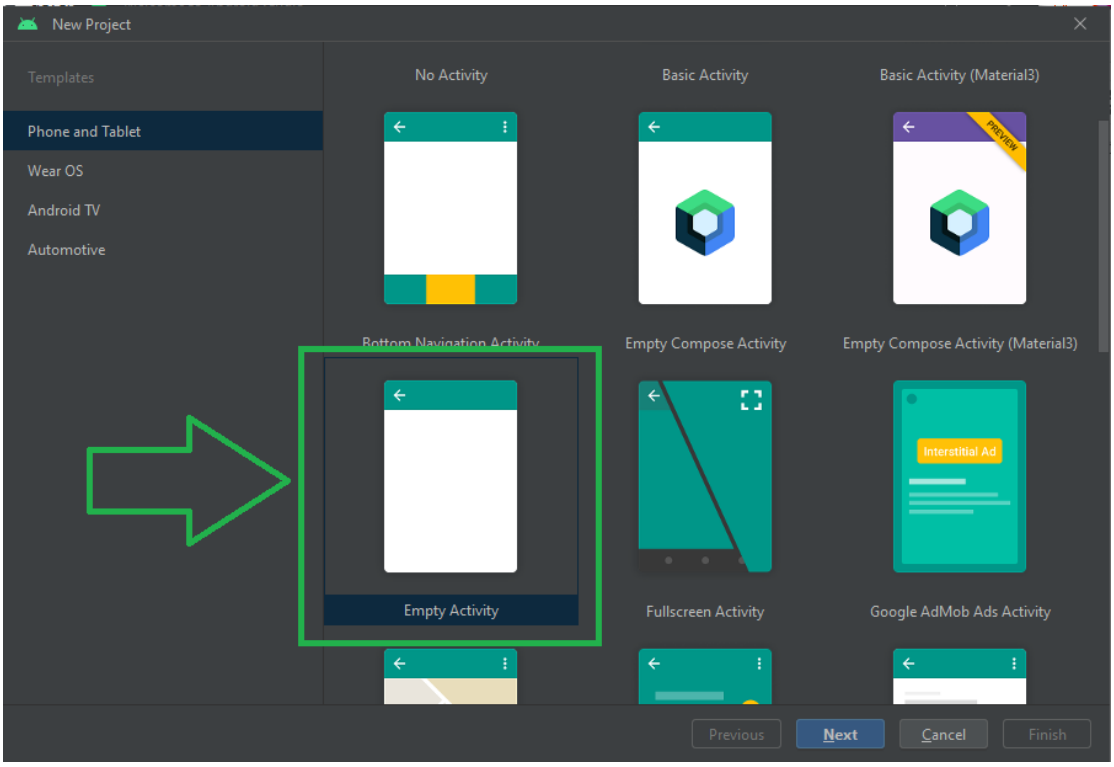

### 2.1.2. Task 2: Create the Hola ESCOM app

Create the app project

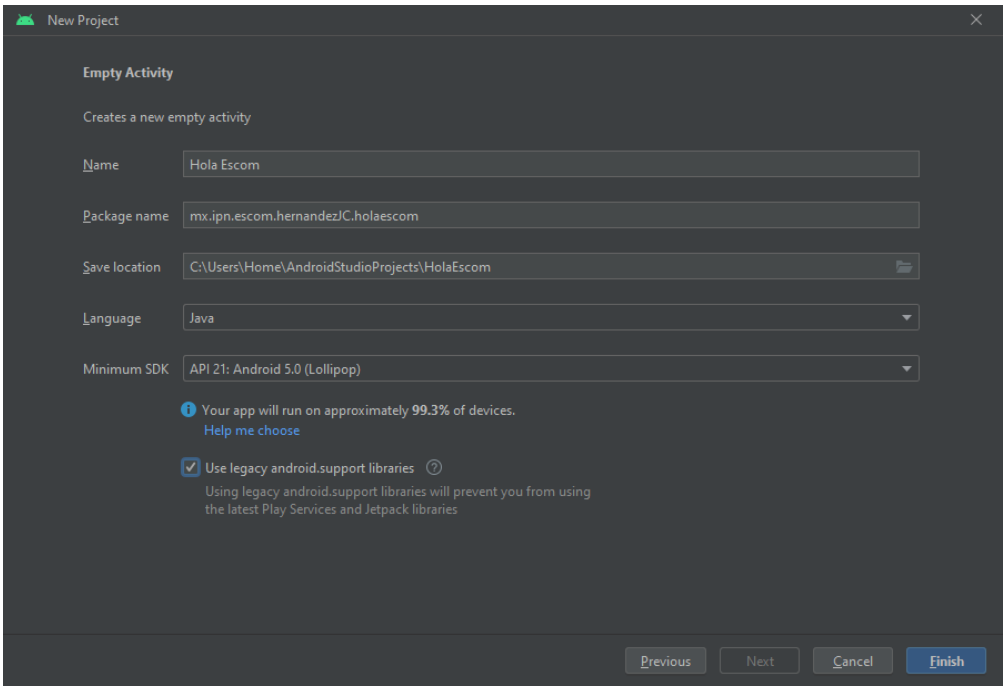
1. Open Android Studio if it is not already opened.
2. In the main Welcome to Android Studio window, click New Project.



3. The New Project window appears, prompting you to choose an Activity. An Activity is a single, focused thing that the user can do. It is a crucial component of any Android app. An Activity typically has a layout associated with it that defines how UI elements appear on a screen. Android Studio provides Activity templates to help you get started. For the Hello World project, choose Empty Activity as shown below, and click Next.




4. In the next window, enter Hello World for the Name.
5. Accept the default com.example.helloworld for the Package name, or create a unique package name. If you are not planning to publish your app, you can accept the default. Be aware that changing the package name of your app later is extra work.
6. Verify that the default Save location is where you want to store your Hello World app and other Android Studio projects, or change it to your preferred directory.
7. Ensure that the selected language is Java; if it is not, use the dropdown menu to set it.
8. Ensure that API 21: Android 5.0 (Lollipop) is set as the Minimum SDK; if it is not, use the dropdown menu to set it.
9. Check the option to Use legacy android.support libraries (Note: newer codelabs will leave this unchecked). As of this writing, these settings make your Hello World app compatible with 98 % of Android devices active on the Google Play store.

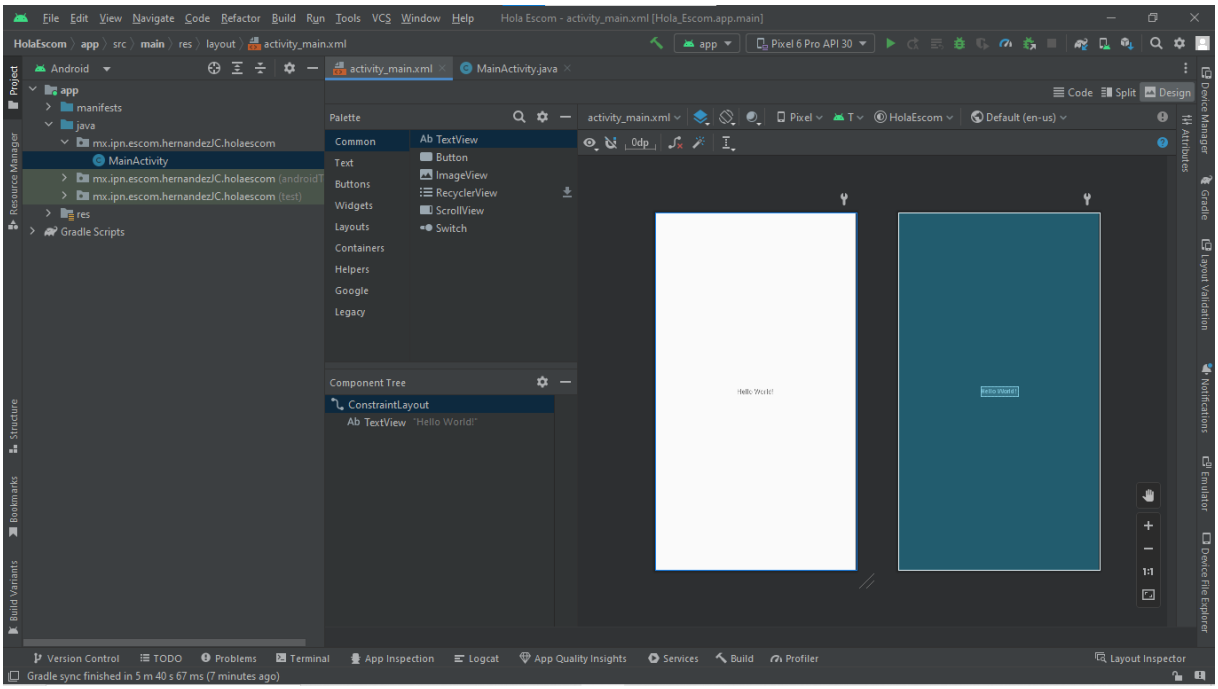



10. Click Finish.

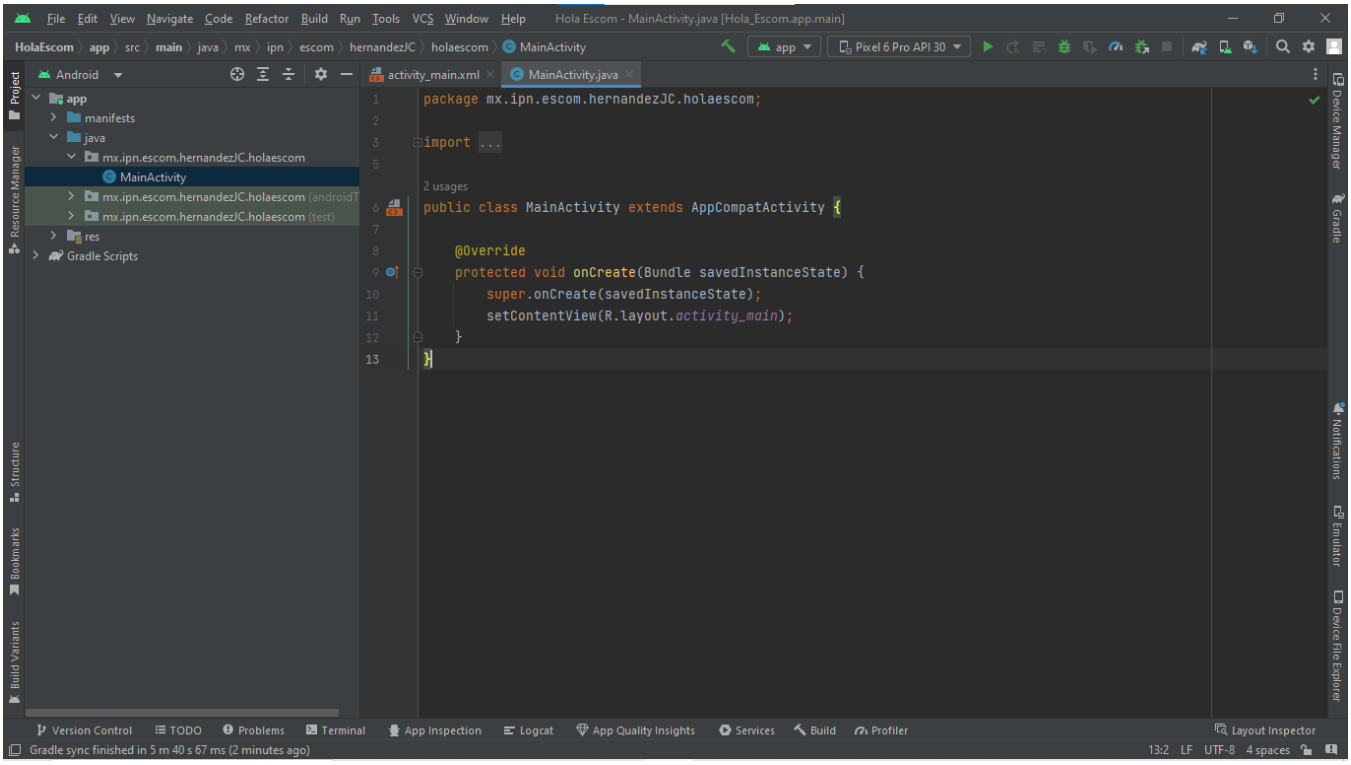
The Android Studio editor appears. Follow these steps:

1. Click the activity_main.xml tab to see the layout editor.

2. Click the layout editor Design tab, if not already selected, to show a graphical rendition of the layout as shown below.

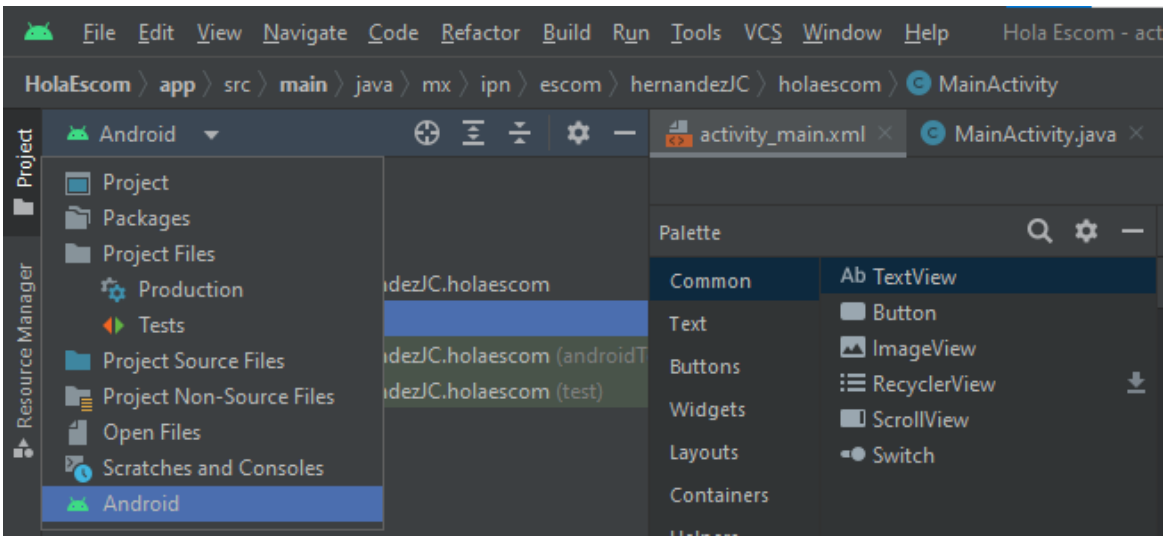


3. Click the MainActivity.java tab to see the code editor as shown below.




Explore the Project >Android pane

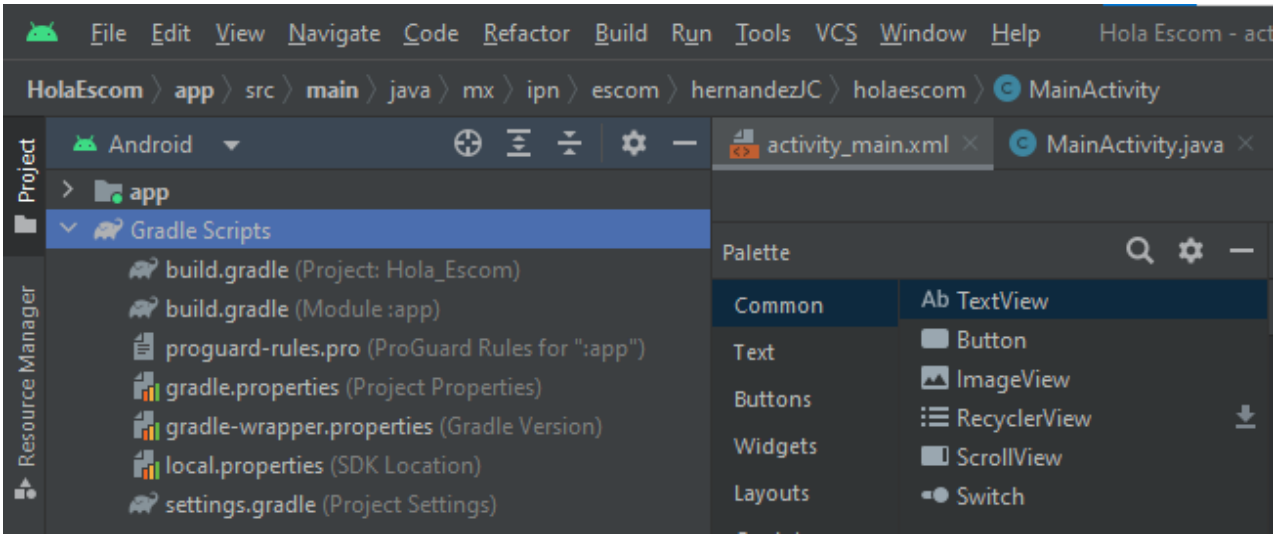
1. If not already selected, click the Project tab in the vertical tab column on the left side of the Android Studio window. The Project pane appears.
2. To view the project in the standard Android project hierarchy, choose Android from the drop-down menu at the top of the Project pane, as shown below.



Explore the Gradle Scripts folder

The Gradle build system in Android Studio makes it easy to include external binaries or other library modules to your build as dependencies.

When you first create an app project, the Project >Android pane appears with the Gradle Scripts folder expanded as shown below.

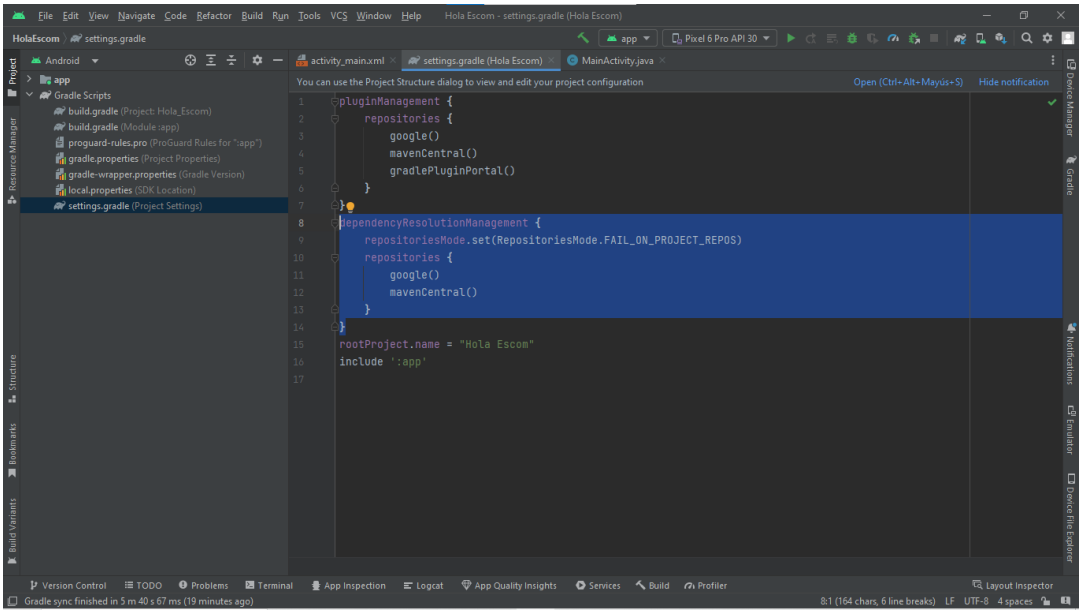



Follow these steps to explore the Gradle system:

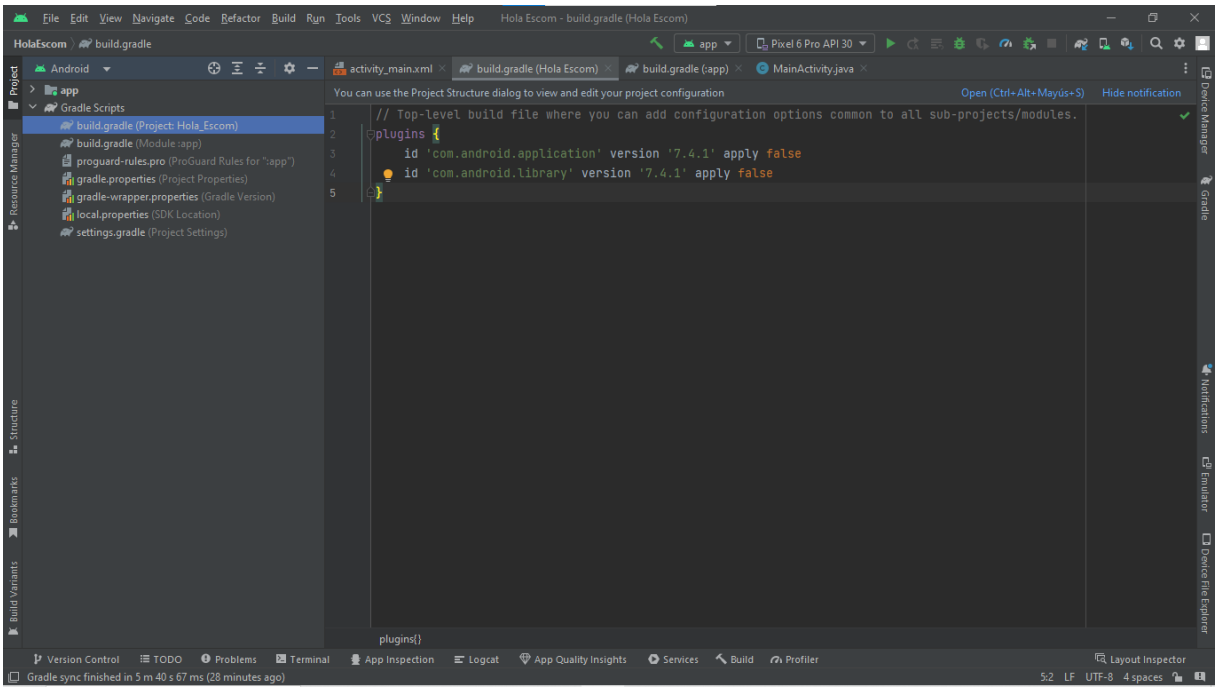
If the Gradle Scripts folder is not expanded, click the arrow to expand it.

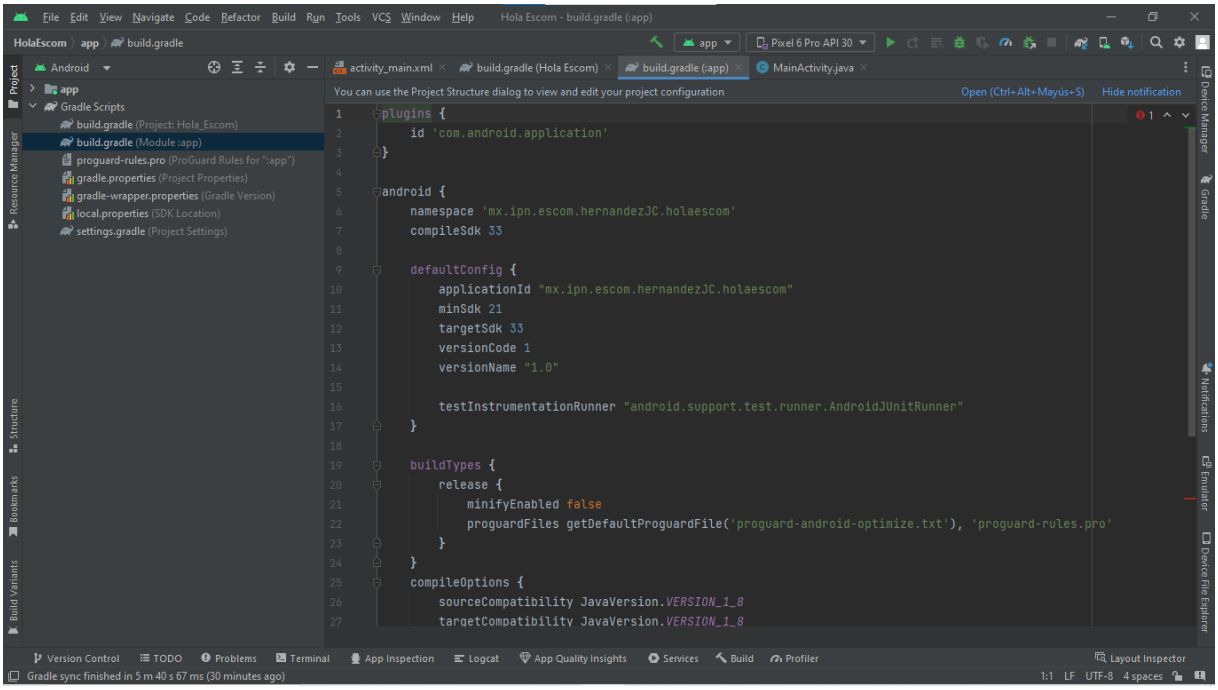
1. This folder contains all the files needed by the build system.

2. Look for the settings.gradle (Project Settings) file.



3. Look for the build.gradle (Project: Hello_World) file.



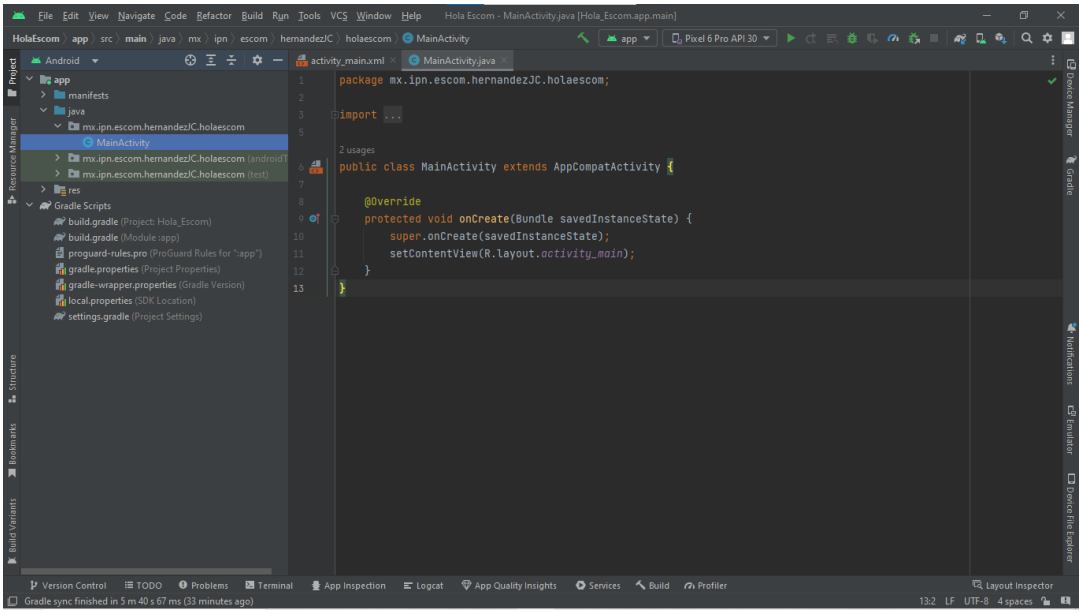




4. Click the arrow to close Gradle Scripts.

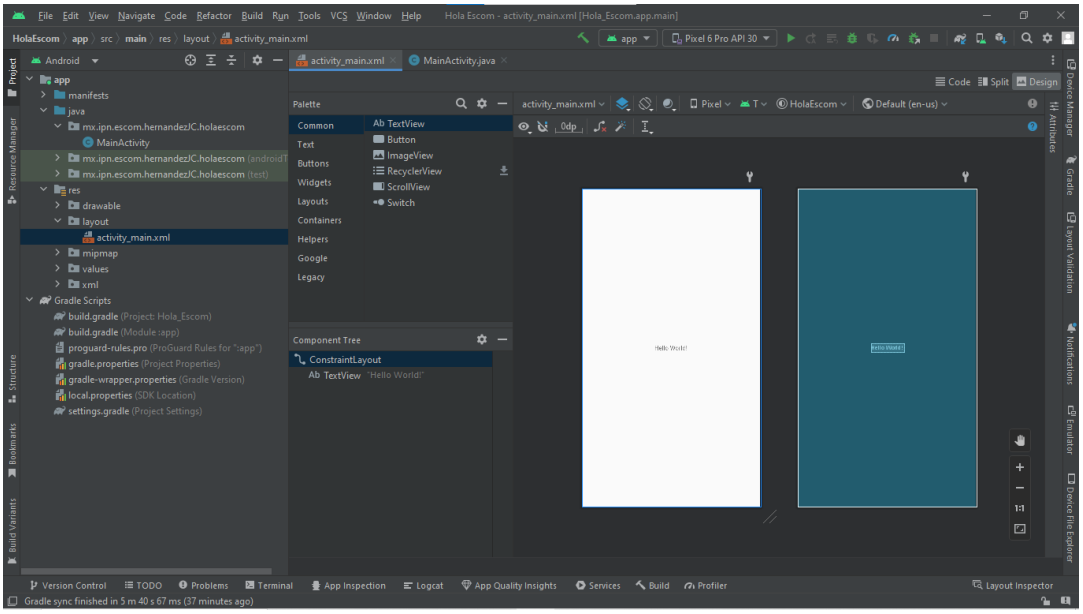
Explore the app and res folders

All code and resources for the app are located within the app and res folders.

1. Expand the app folder, the java folder, and the com.example.helloworld folder to see the MainActivity java file. Double-clicking the file opens it in the code editor.

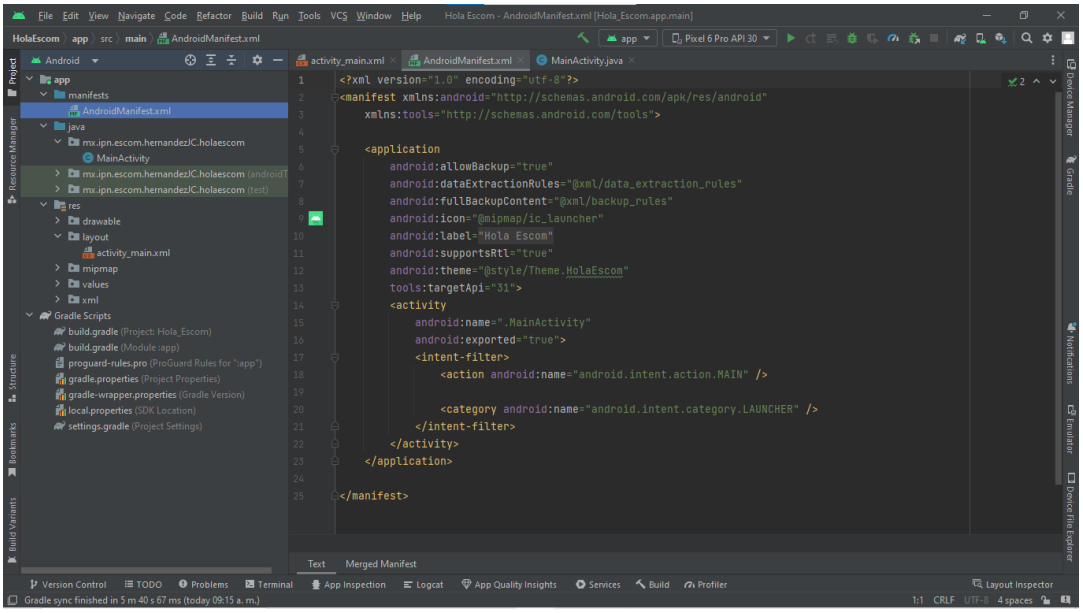



2. Expand the res folder and the layout folder, and double-click the activity_main.xml file to open it in the layout editor.



Explore the manifests folder
The manifests folder contains files that provide essential information about your app to the Android system, which the system must have before it can run any of the app's code.

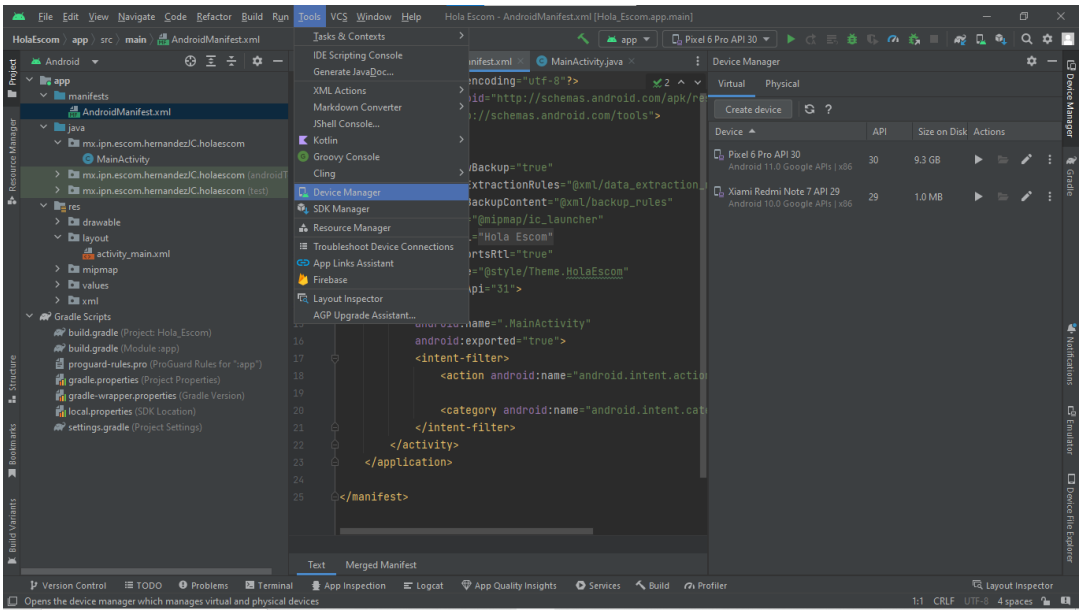
1. Expand the manifests folder.
2. Open the AndroidManifest.xml file.



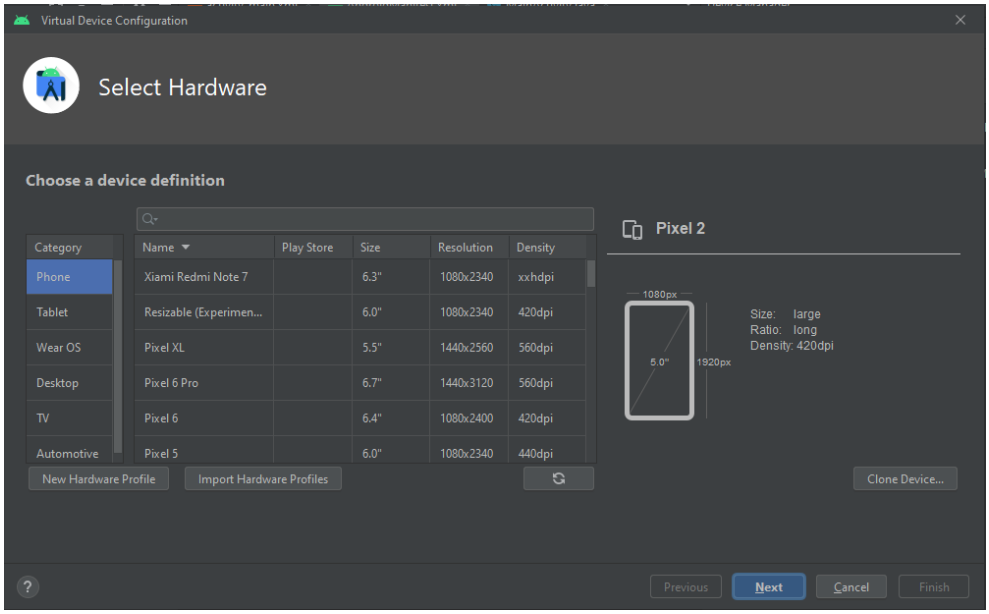

### 2.1.3. Task 3:Use a virtual device(emulator)

Create an Android virtual device (AVD)
In order to run an emulator on your computer, you have to create a configuration that describes the virtual device.

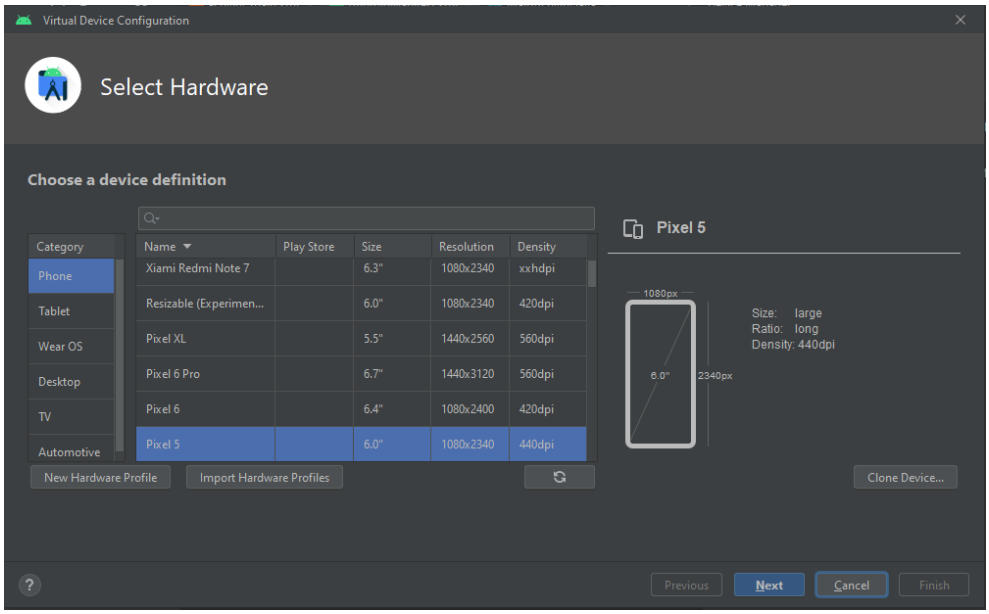
1. In Android Studio, select Tools >Device Manager, or click the Device Manager icon AVD Manager Icon in the toolbar. The Device Manager pane appears. If you've already created virtual devices, the pane shows them (as shown in the figure below); otherwise you see a blank list.



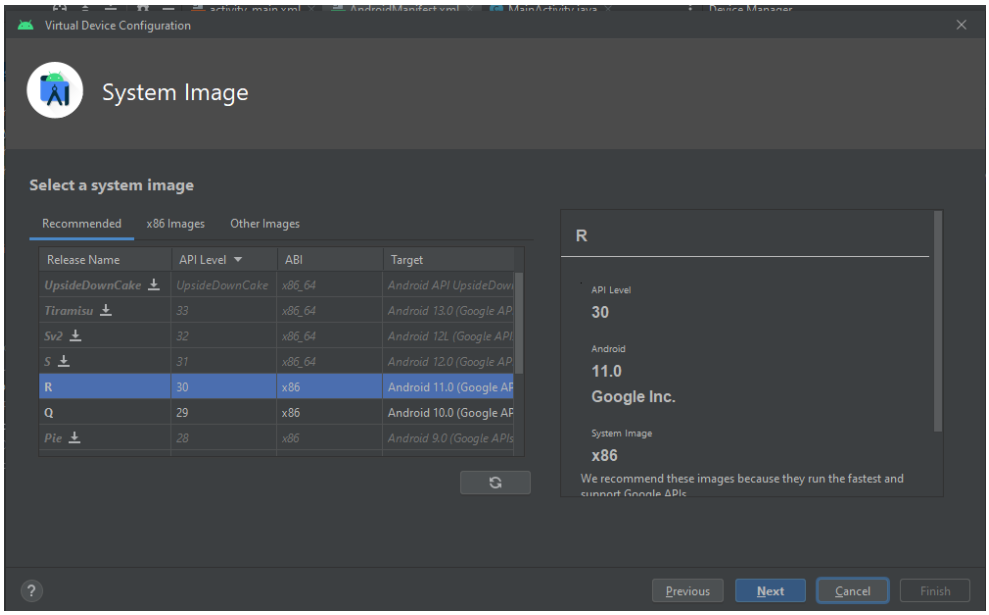

2. Click the Create Device button. The Select Hardware window appears showing a list of pre configured hardware devices. For each device, the table provides a column for its diagonal display size (Size), screen resolution in pixels (Resolution), and pixel density (Density).



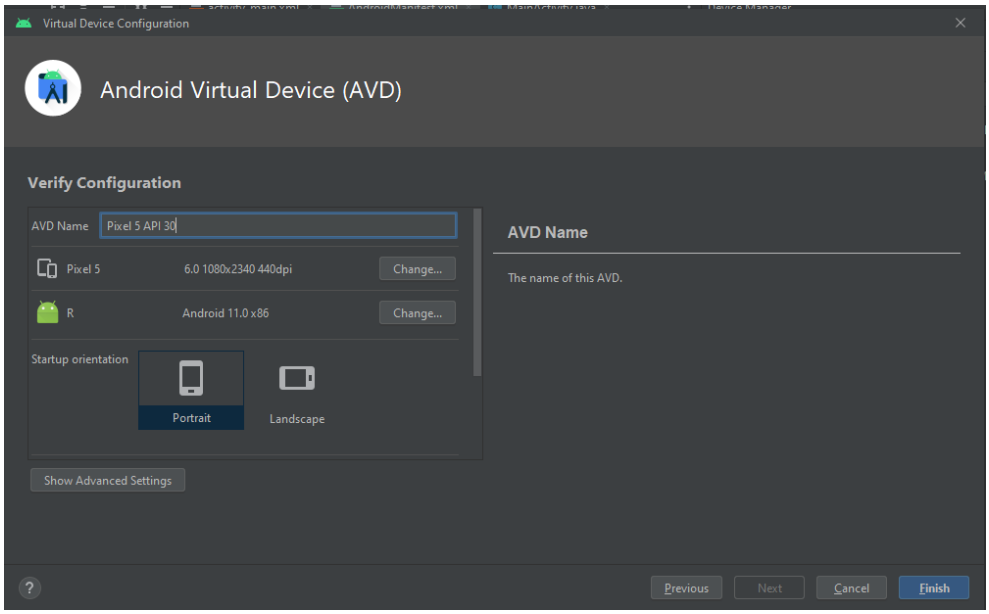

3. Choose a device such as Pixel 5, and click Next. The System Image screen appears.




4. Click the Recommended tab if it is not already selected, and choose which version of the Android system to run on the virtual device (such as R).

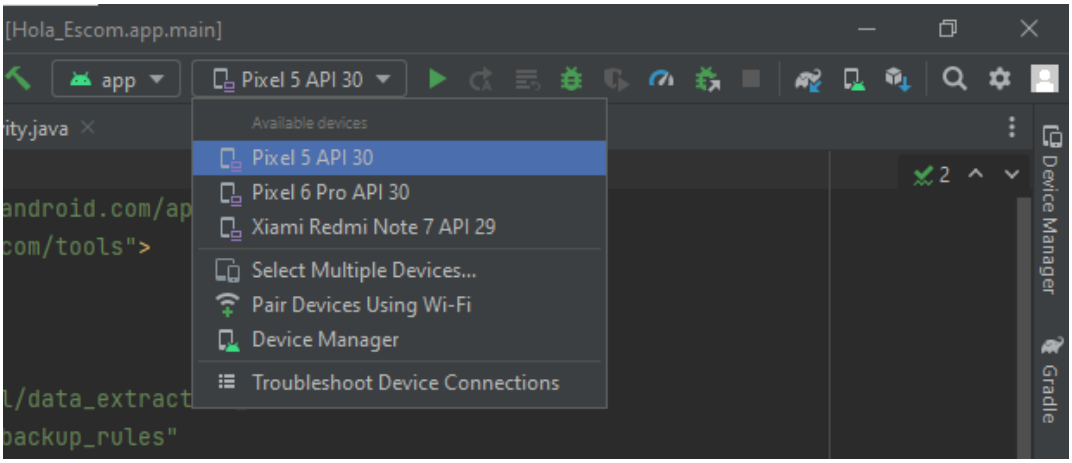



5. After choosing a system image, click Next. The Android Virtual Device (AVD) window appears. You can also change the name of the AVD. Check your configuration and click Finish.

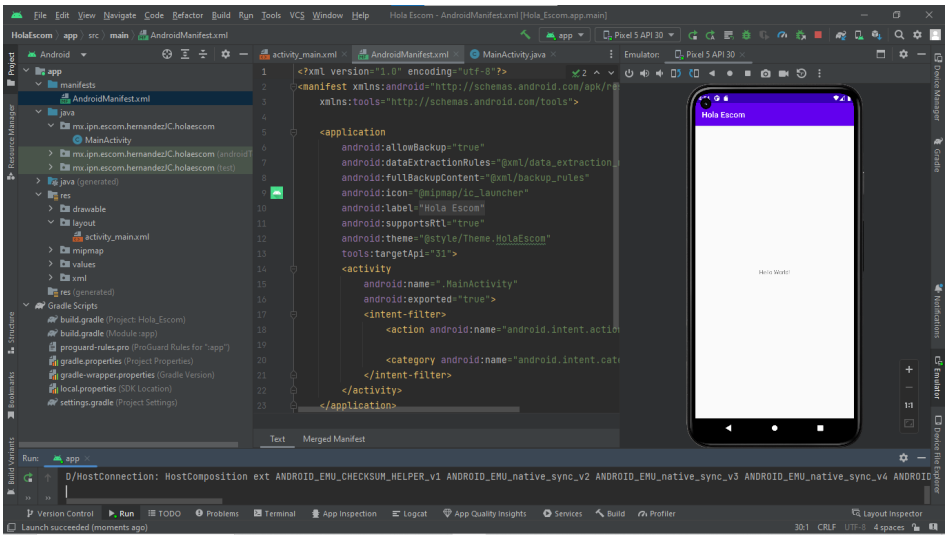



Run the app on the virtual device
In this task, you will finally run your Hello World app.

1. In Android Studio, choose Run >Select Device or click the devices dropdown next to the Run icon choose Run >Run app or click the Run icon in the toolbar.  ic_run.png, Android Studio Run icon in the toolbar and select the virtual device which you just created.

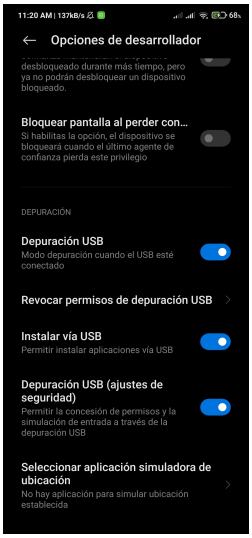



2. Choose Run >Run app or click the Run icon in the toolbar.




### 2.1.4. Task 4: Use a physical device

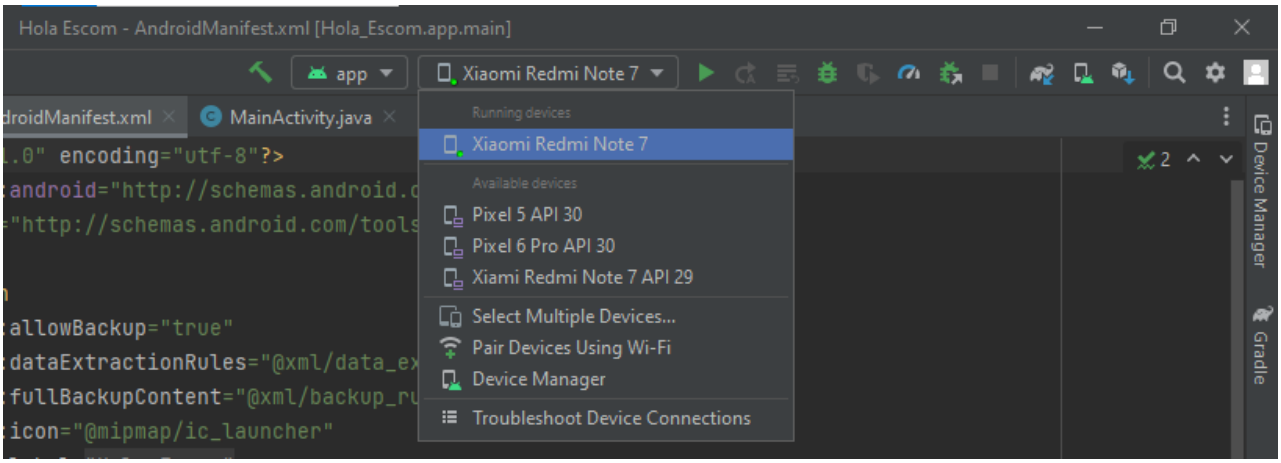
Turn on USB debugging
To let Android Studio communicate with your device, you must turn on USB Debugging on your Android device. This is enabled in the Developer options settings of your device.
On Android 4.2 and higher, the Developer options screen is hidden by default. To show developer options and enable USB Debugging:
1. On your device, open Settings, search for About phone, click on About phone, and tap Build number seven times.
2. Return to the previous screen (Settings / System). Developer options appears in the list. Tap Developer options.
3. Choose USB Debugging.



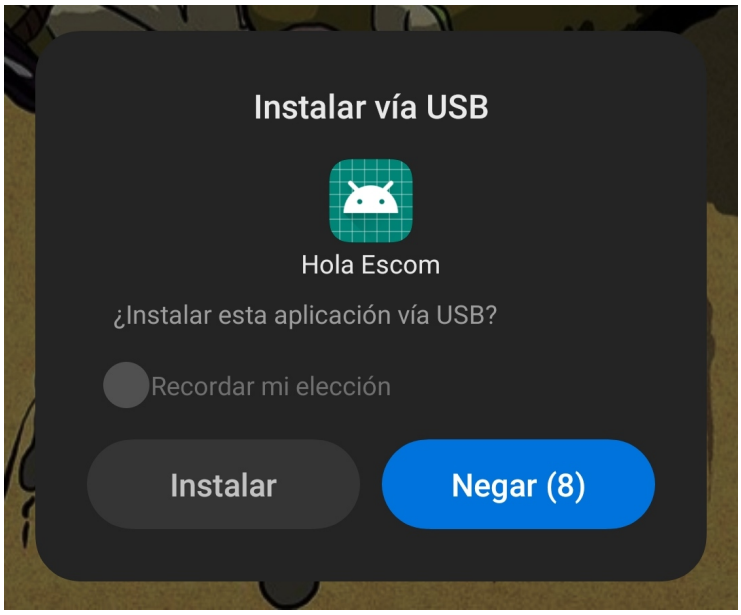
Run your app on a device

Now you can connect your device and run the app from Android Studio.

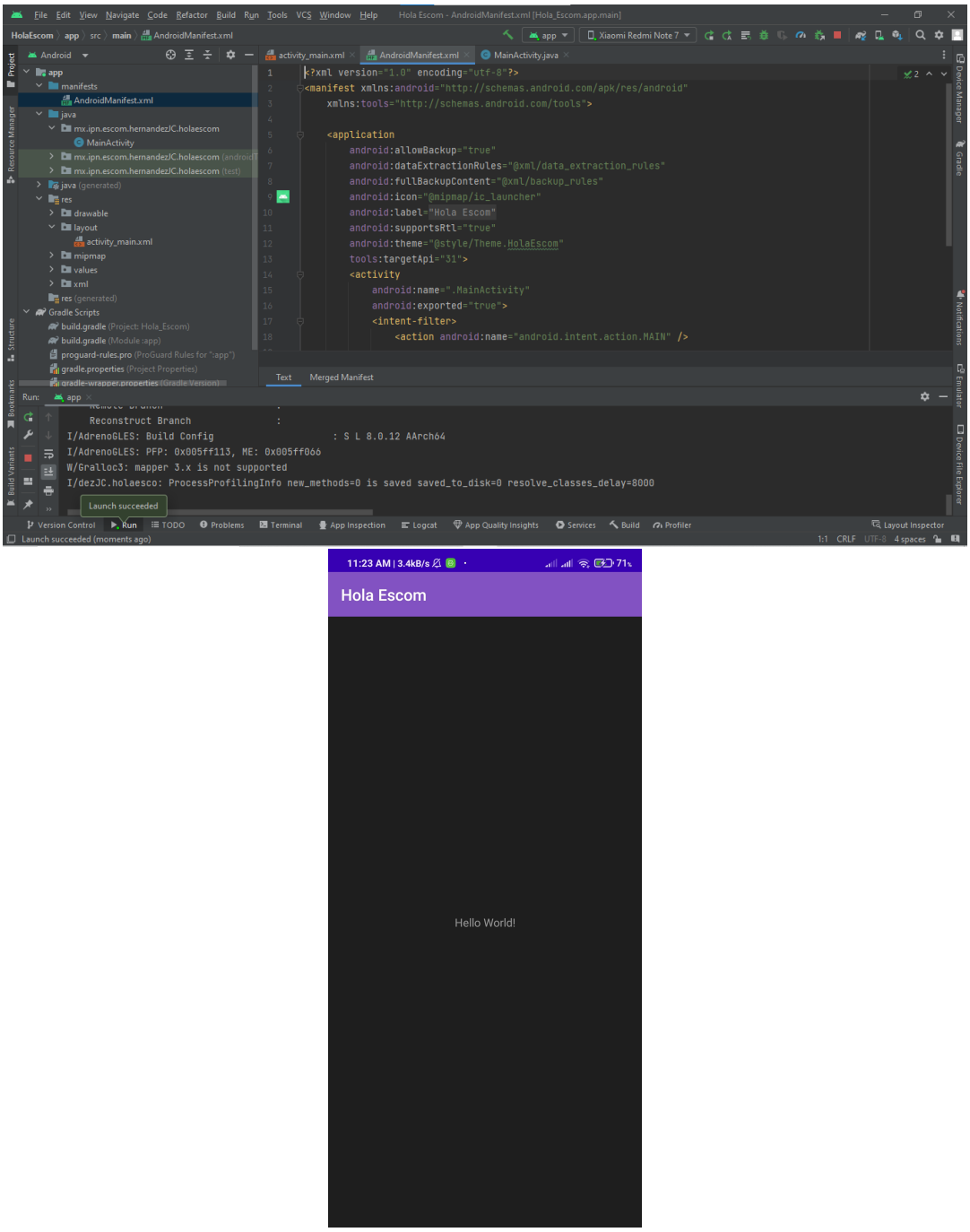
1. Connect your device to your development machine with a USB cable.




2. In Android Studio, choose Run >Select device or click the devices dropdown next to the Run button Android Studio Run icon in the toolbar and select your connected device.




3. Click the Run button in the toolbar.







### 2.1.5. Task 5: Change de app Graddle Configuration

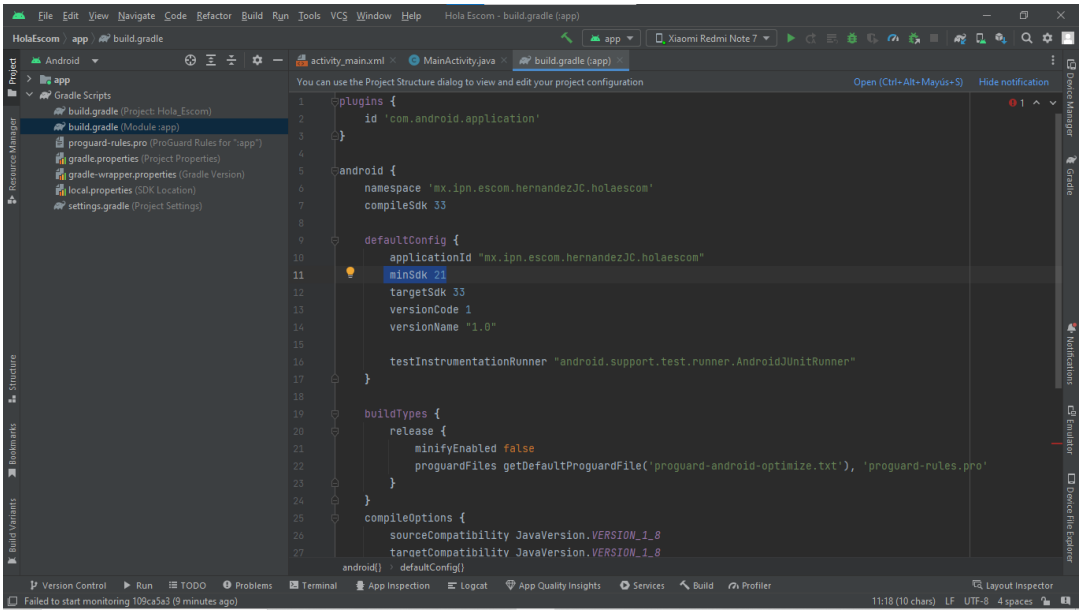
Change the minimum SDK version for the app

Follow these steps:

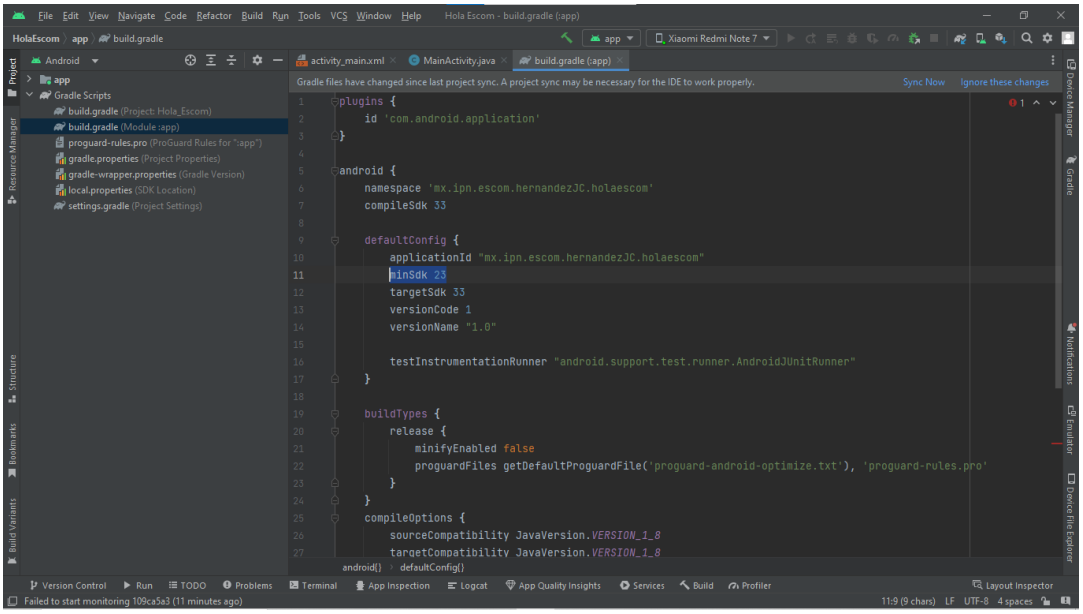
1. Expand the Gradle Scripts folder if it is not already open, and double-click the build.gradle

(Module:Hello_World.app) file. The content of the file appears in the code editor.

2. Within the defaultConfig block, change the value of minSdkVersion to 23.

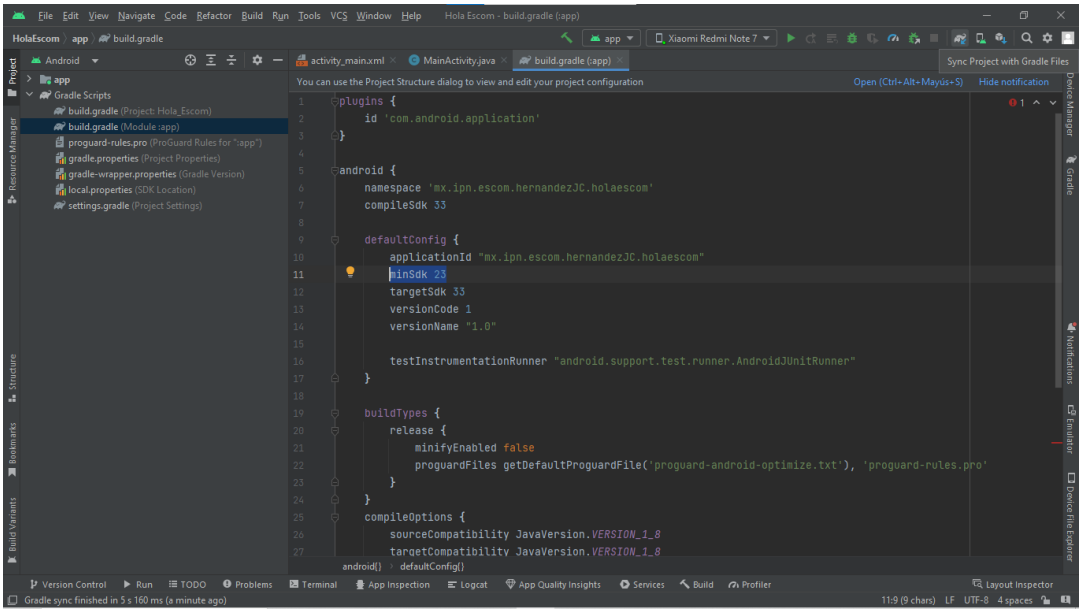



3. The code editor shows a notification bar at the top with the Sync Now link.




Sync the new Gradle configuration When you make changes to the build configuration files in a project, Android Studio requires that you sync the project files so that it can import the build configuration changes and run some checks to make sure the configuration won't create build errors.

To sync the project files, click Sync Now in the notification bar that appears when making a change (as shown in the previous figure), or click the Sync Project with Gradle Files icon click Sync Now in the notification bar that appears when making a change (as shown in the previous figure), or click the Sync Project with Gradle Files icon in the toolbar. ic_gradle_sync.png, Sync Project with Gradle Files in the toolbar.

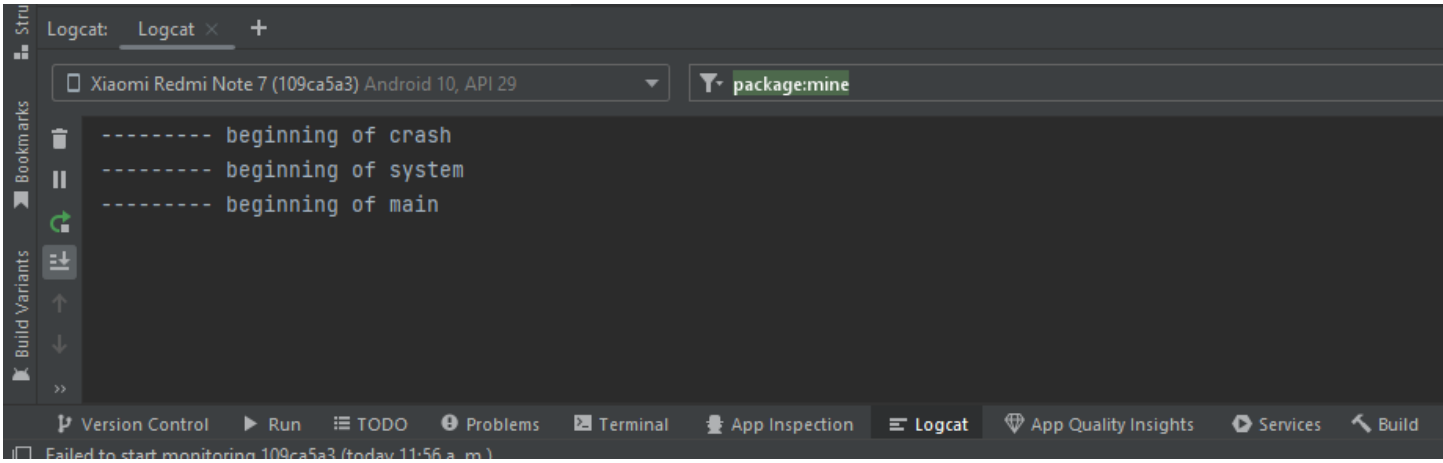



### 2.1.6. Task 6: Add log statements to your app

To see the Logcat pane, click the Logcat tab at the bottom of the Android Studio window as shown in the figure below.

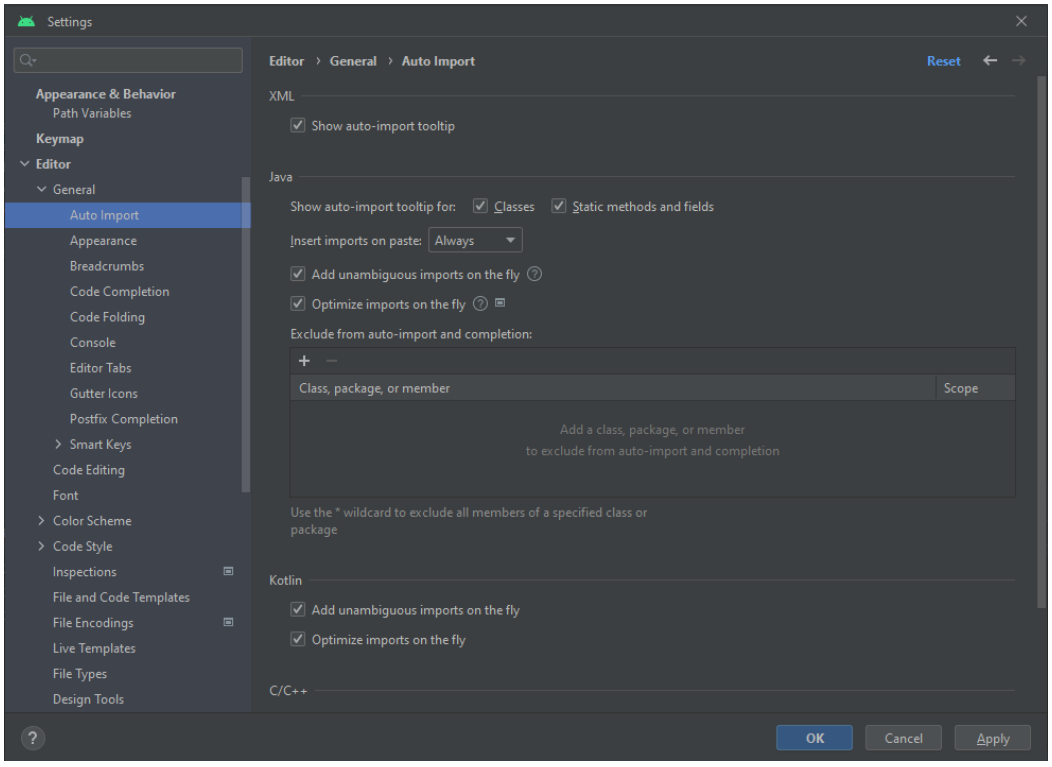
In the figure above:

1. The Logcat tab for opening and closing the Logcat pane, which displays information about your app as it is running. If you add Log statements to your app, Log messages appear here.
2. The Log level menu is set to Verbose (the default), which shows all Log messages. Other settings include Debug, Error, Info, and Warn.




Add log statements to your app

1. Open your Hello World app in Android studio, and open MainActivity.// 2. To add unambiguous imports automatically to your project (such as android.util.Log required for using Log), choose File >Settings in Windows, or Android Studio >Preferences in macOS.
3. Choose Editor >General >Auto Import. Select all checkboxes and set Insert imports on paste to Always.
4. Click Apply and then click OK.




5. In the onCreate() method of MainActivity, add the following statement:

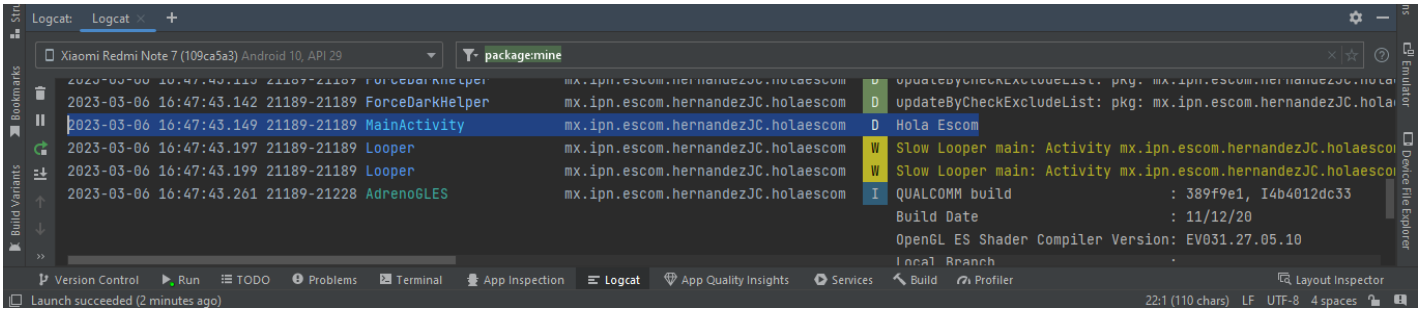
```java
package mx.ipn.escom.hernandezJC.holaescom;

import ...

2 usages
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Log.d( tag: "MainActivity",  msg: "Hola Escom");
    }
}
```

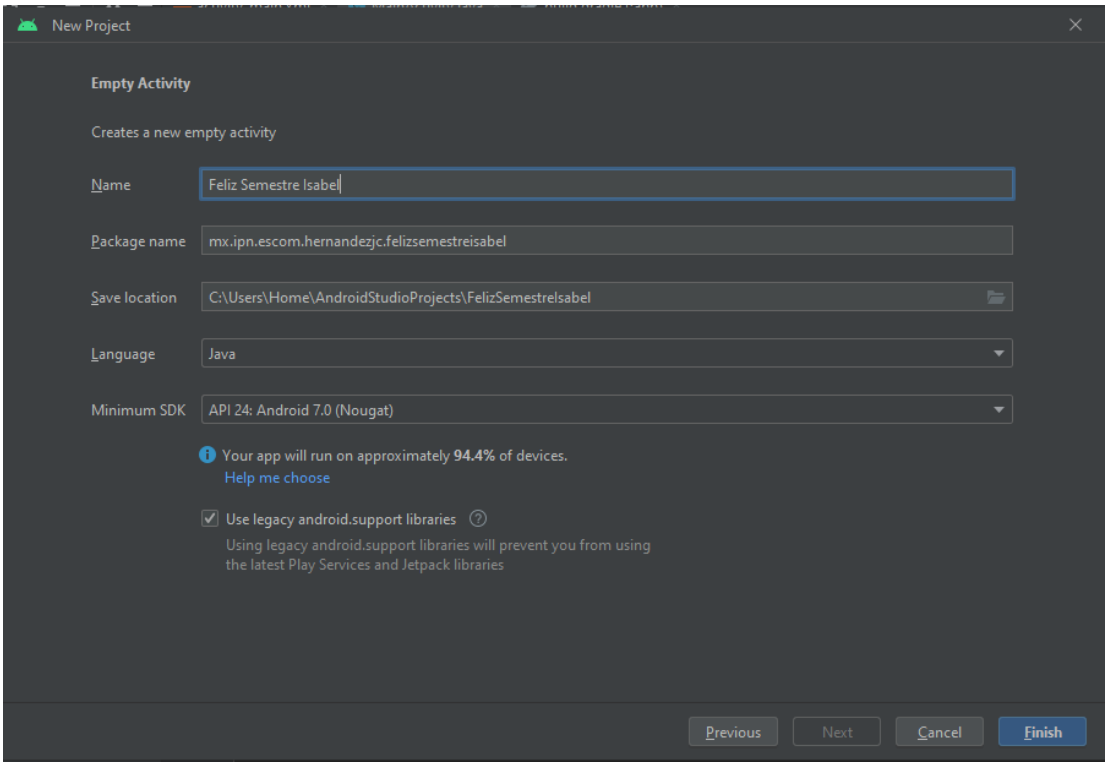
6. If the Logcat pane is not already open, click the Logcat tab at the bottom of Android Studio to open it.
7. Check that the name of the target and package name of the app are correct.
8. Change the Log level in the Logcat pane to Debug (or leave as Verbose since there are so few log messages).
9. Run your app.

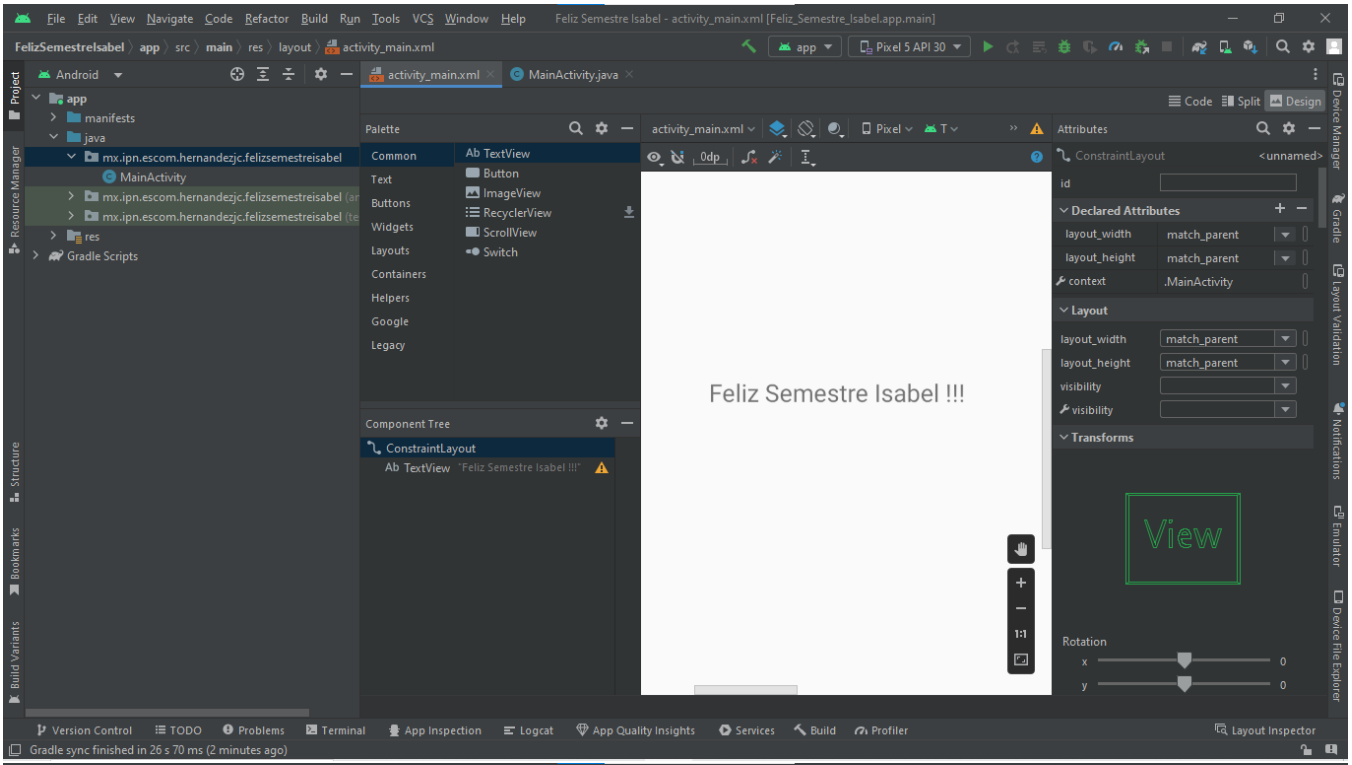


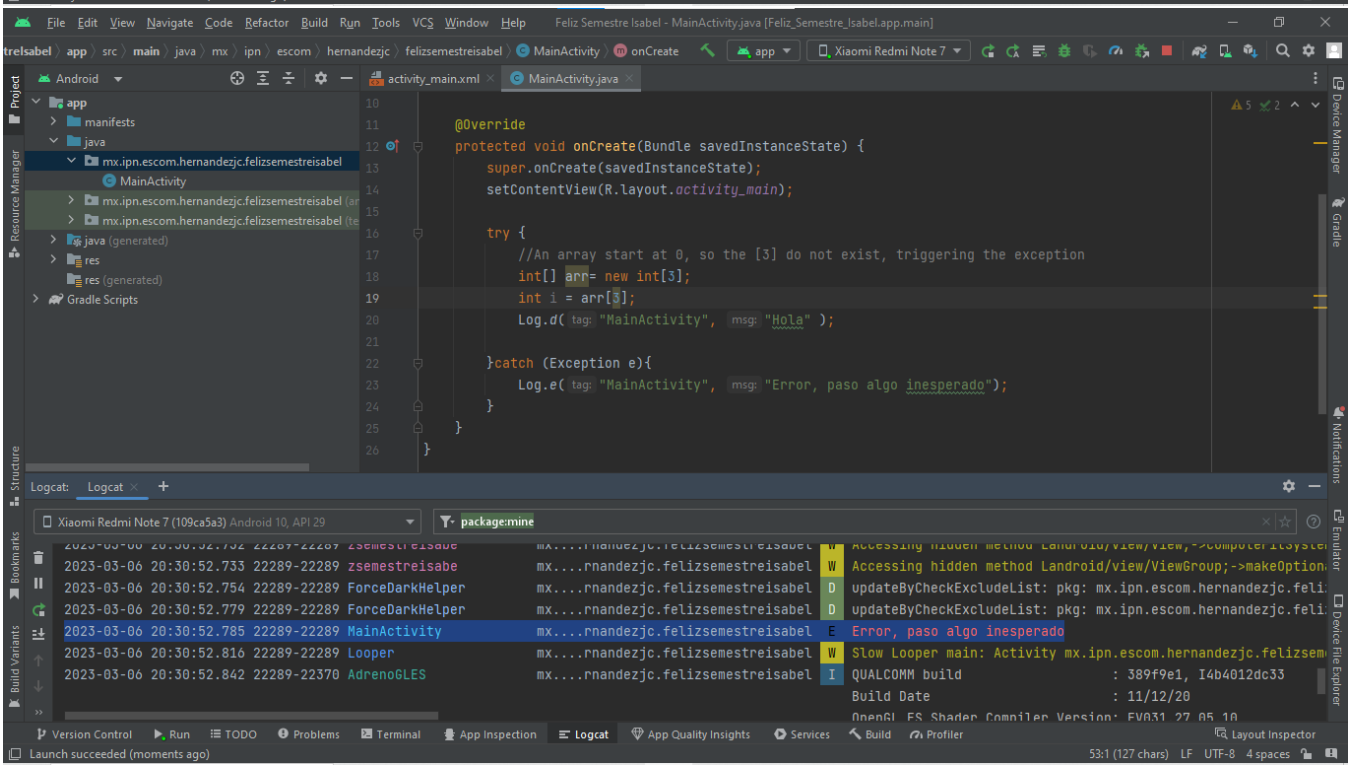

## 2.2. Codding challenge

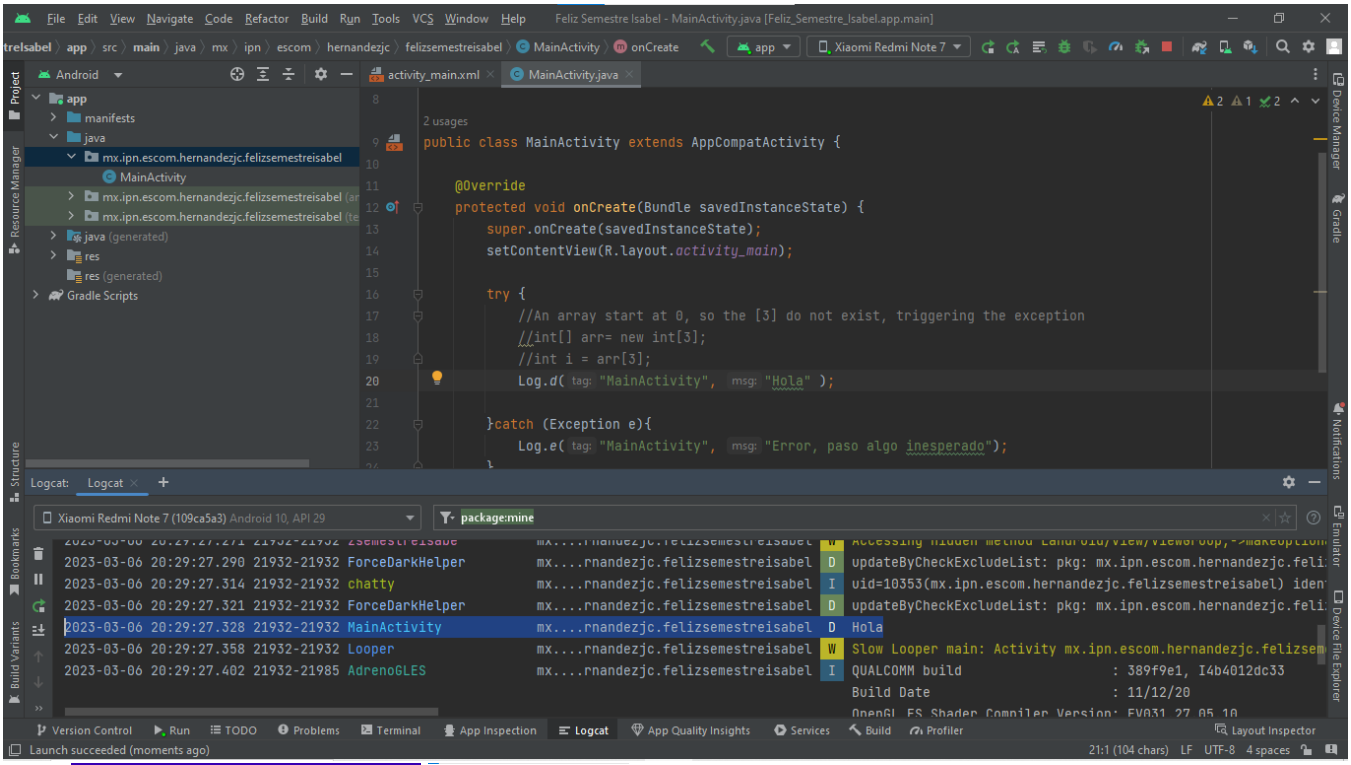
Challenge: Now that you are set up and familiar with the basic development workflow, do the following:

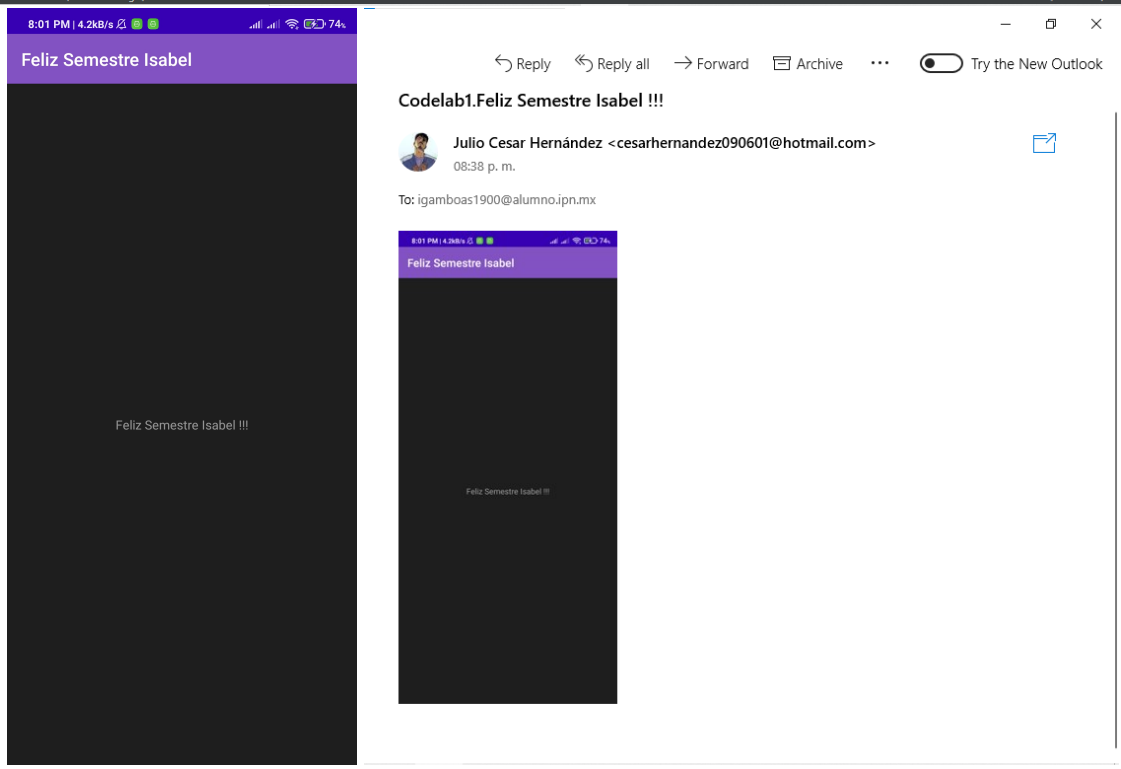
1. Create a new project in Android Studio.
2. Change the "Hello World"greeting to "Feliz Semestre .ªnd the name of someone with a recent birthday.
3. (Optional) Take a screenshot of your finished app and email it to someone whose birthday you forgot.
4. A common use of the Log class is to log Java exceptions when they occur in your program. There are some useful methods, such as Log.e(), that you can use for this purpose. Explore methods you can use to include an exception with a Log message. Then, write code in your app to trigger and log an exception.




Feliz Semestre Isabel !!!
View

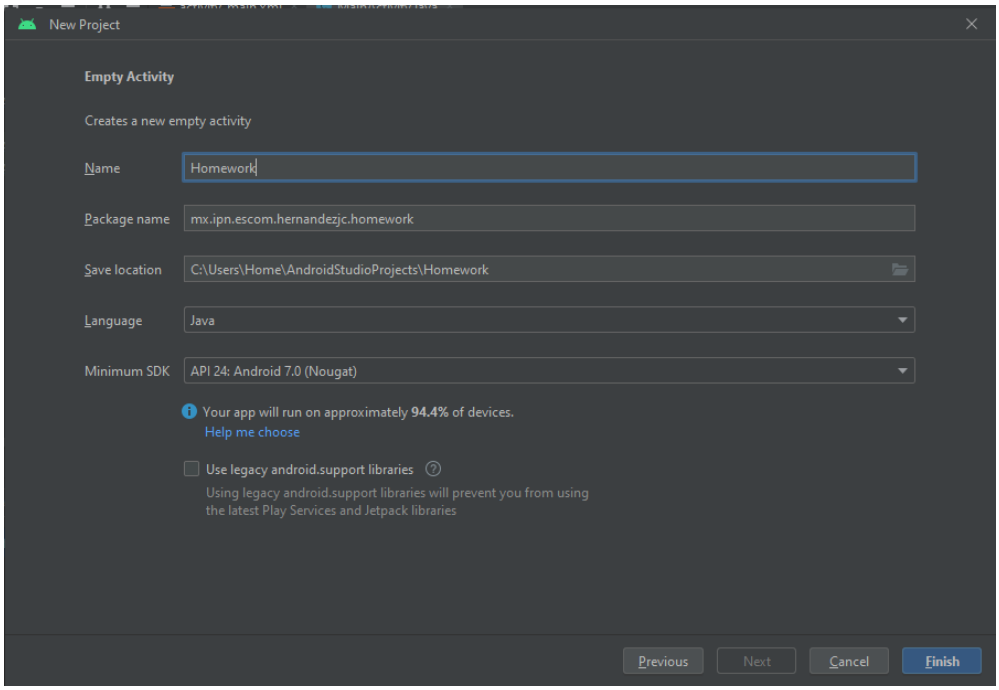

@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
try {
//An array start at 0, so the [3] do not exist, triggering the exception
int[] arr= new int[3];
int i = arr[3];
Log.d( tag: "MainActivity", msg: "Hola" );
}catch (Exception e){
Log.e( tag: "MainActivity", msg: "Error, paso algo inesperado");
}
}
2023-03-06 20:30:52.733 22289-22289 zsemestreisabe mx....rnandezjc.felizsemestreisabel W Accessing hidden method Landroid/view/ViewGroup;->makeOption
2023-03-06 20:30:52.754 22289-22289 ForceDarkHelper mx....rnandezjc.felizsemestreisabel D updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.feli
2023-03-06 20:30:52.779 22289-22289 ForceDarkHelper mx....rnandezjc.felizsemestreisabel D updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.feli
2023-03-06 20:30:52.785 22289-22289 MainActivity mx....rnandezjc.felizsemestreisabel E Error, paso algo inesperado
2023-03-06 20:30:52.816 22289-22289 Looper mx....rnandezjc.felizsemestreisabel W Slow Looper main: Activity mx.ipn.escom.hernandezjc.felizsem
2023-03-06 20:30:52.842 22289-22370 AdrenoGLES mx....rnandezjc.felizsemestreisabel I QUALCOMM build : 389f9e1, I4b4012dc33
Build Date : 11/12/20

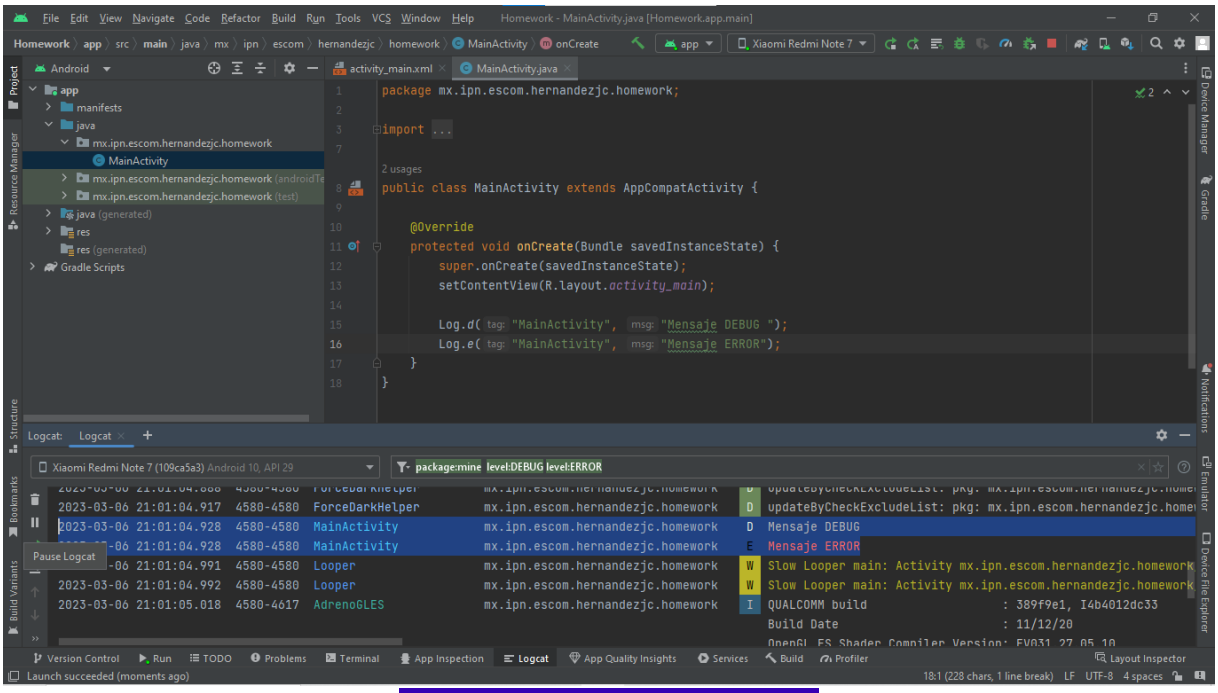

File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
Feliz Semestre Isabel - MainActivity.java [Feliz_Semestre_Isabel.app.main]
Android
app
manifests
java
mx.ipn.escom.hernandezjc.felizsemestreisabel
MainActivity
res
Gradle Scripts
activity_main.xml
MainActivity.java
2 usages
public class MainActivity extends AppCompatActivity {
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
try {
//An array start at 0, so the [3] do not exist, triggering the exception
//int[] arr= new int[3];
//int i = arr[3];
Log.d( tag: "MainActivity", msg: "Hola" );
}catch (Exception e){
Log.e( tag: "MainActivity", msg: "Error, paso algo inesperado");
Logcat
Xiaomi Redmi Note 7 (109ca5a3) Android 10, API 29
package:mine
2023-03-06 20:29:27.290 21932-21932 ForceDarkHelper
2023-03-06 20:29:27.314 21932-21932 chatty
2023-03-06 20:29:27.321 21932-21932 ForceDarkHelper
2023-03-06 20:29:27.328 21932-21932 MainActivity
2023-03-06 20:29:27.358 21932-21932 Looper
2023-03-06 20:29:27.402 21932-21985 AdrenoGLES
updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.feli
uid=10353(mx.ipn.escom.hernandezjc.felizsemestreisabel) iden
updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.feli
Hola
Slow Looper main: Activity mx.ipn.escom.hernandezjc.felizsem
QUALCOMM build : 389f9e1, I4b4012dc33
Build Date : 11/12/20
Version Control
Run
TODO
Problems
Terminal
App Inspection
Logcat
App Quality Insights
Services
Build
Profiler
Layout Inspector
Launch succeeded (moments ago)

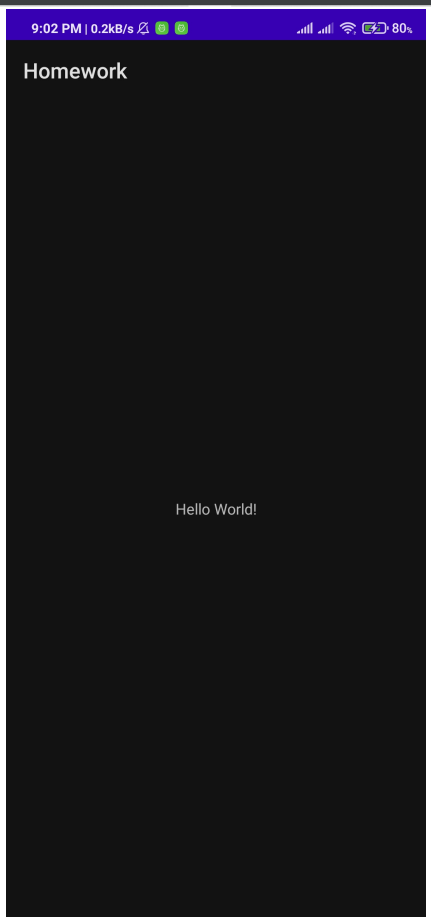

8:01 PM | 4.2kB/s
Feliz Semestre Isabel
Feliz Semestre Isabel !!!
Reply
Reply all
Forward
Archive
Try the New Outlook
Codelab1.Feliz Semestre Isabel !!!
Julio Cesar Hernández <cesarhernandez090601@hotmail.com>
08:38 p. m.
To: igamboas1900@alumno.ipn.mx
Feliz Semestre Isabel
Feliz Semestre Isabel !!!

## 2.3. Homework

Build and run an app -Create a new Android project from the Empty Activity Template.
-Add logging statements for various log levels in onCreate() in the main activity.
-Create an emulator for a device, targeting any version of Android you like, and run the app.
-Use filtering in Logcat to find your log statements and adjust the levels to only display debug or error logging statements.




File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
Homework - MainActivity.java [Homework.app.main]
Homework app src main java mx ipn escom hernandezjc homework MainActivity onCreate
app
Xiaomi Redmi Note 7
Android
activity_main.xml
MainActivity.java
app
manifests
java
mx.ipn.escom.hernandezjc.homework
MainActivity
mx.ipn.escom.hernandezjc.homework (androidTe
mx.ipn.escom.hernandezjc.homework (test)
java (generated)
res
res (generated)
Gradle Scripts
package mx.ipn.escom.hernandezjc.homework;
import ...
2 usages
public class MainActivity extends AppCompatActivity {
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
Log.d( tag: "MainActivity", msg: "Mensaje DEBUG ");
Log.e( tag: "MainActivity", msg: "Mensaje ERROR");
}
}
Logcat:
Logcat
Xiaomi Redmi Note 7 (109ca5a3) Android 10, API 29
package:mine level:DEBUG level:ERROR
2023-03-06 21:01:04.917 4580-4580 ForceDarkHelper mx.ipn.escom.hernandezjc.homework D updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.homework
2023-03-06 21:01:04.928 4580-4580 MainActivity mx.ipn.escom.hernandezjc.homework D Mensaje DEBUG
2023-03-06 21:01:04.928 4580-4580 MainActivity mx.ipn.escom.hernandezjc.homework E Mensaje ERROR
2023-03-06 21:01:04.991 4580-4580 Looper mx.ipn.escom.hernandezjc.homework W Slow Looper main: Activity mx.ipn.escom.hernandezjc.homework
2023-03-06 21:01:04.992 4580-4580 Looper mx.ipn.escom.hernandezjc.homework W Slow Looper main: Activity mx.ipn.escom.hernandezjc.homework
2023-03-06 21:01:05.018 4580-4617 AdrenoGLES mx.ipn.escom.hernandezjc.homework I QUALCOMM build : 389f9e1, I4b4012dc33
Build Date : 11/12/20
Pause Logcat
Version Control Run TODO Problems Terminal App Inspection Logcat App Quality Insights Services Build Profiler Layout Inspector
Launch succeeded (moments ago)
18:1 (228 chars, 1 line break) LF UTF-8 4 spaces


9:02 PM | 0.2kB/s
80%
Homework
Hello World!

Question 1

What is the name of the layout file for the main activity?

activity_main.xml

Question 2

What is the name of the string resource that specifies the application's name?

android:name

Question 3

Which tool do you use to create a new emulator?

Device Manager

Question 4 Assume that your app includes this logging statement:

Log.i("MainActivity", "MainActivity layout is complete"); You see the statement "MainActivity layout is completeïn the Logcat pane if the Log level menu is set to which of the following? (Hint: multiple answers are OK.)

Debug
Info