



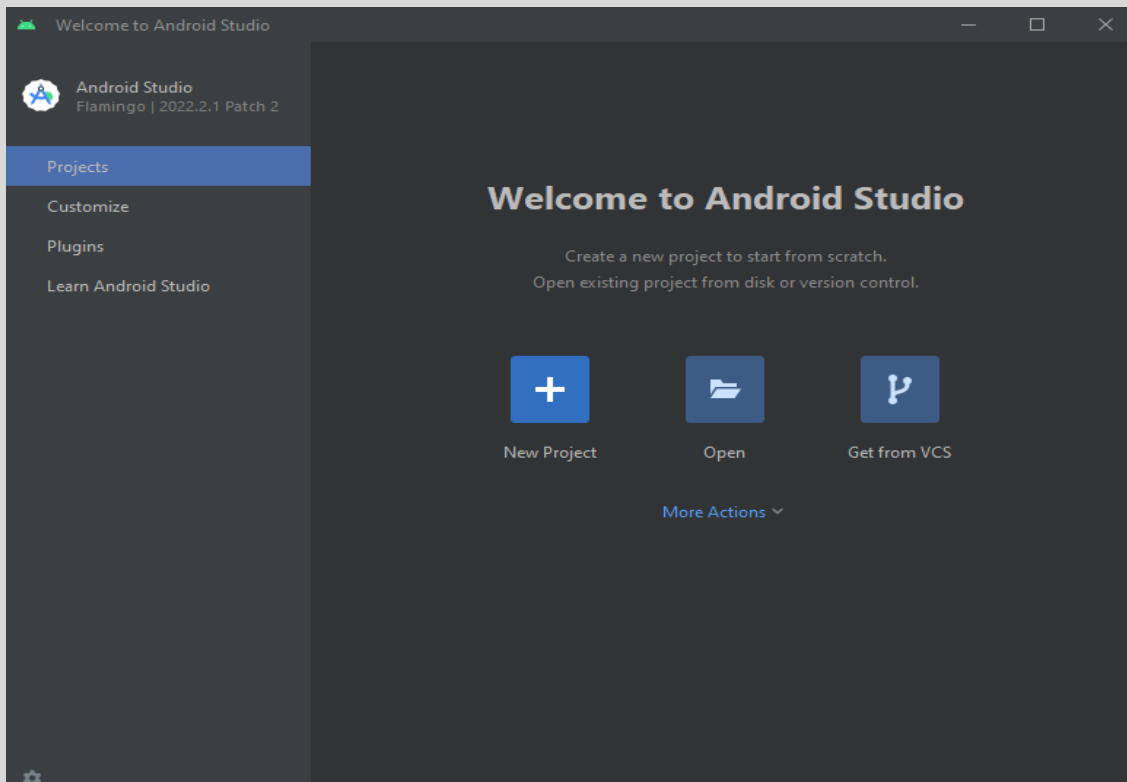
# Mi primera APP - Android Studio/Kotlin

ELIANA YINETH  
LOZANO TRIANA  
eylozano@sena.edu.co

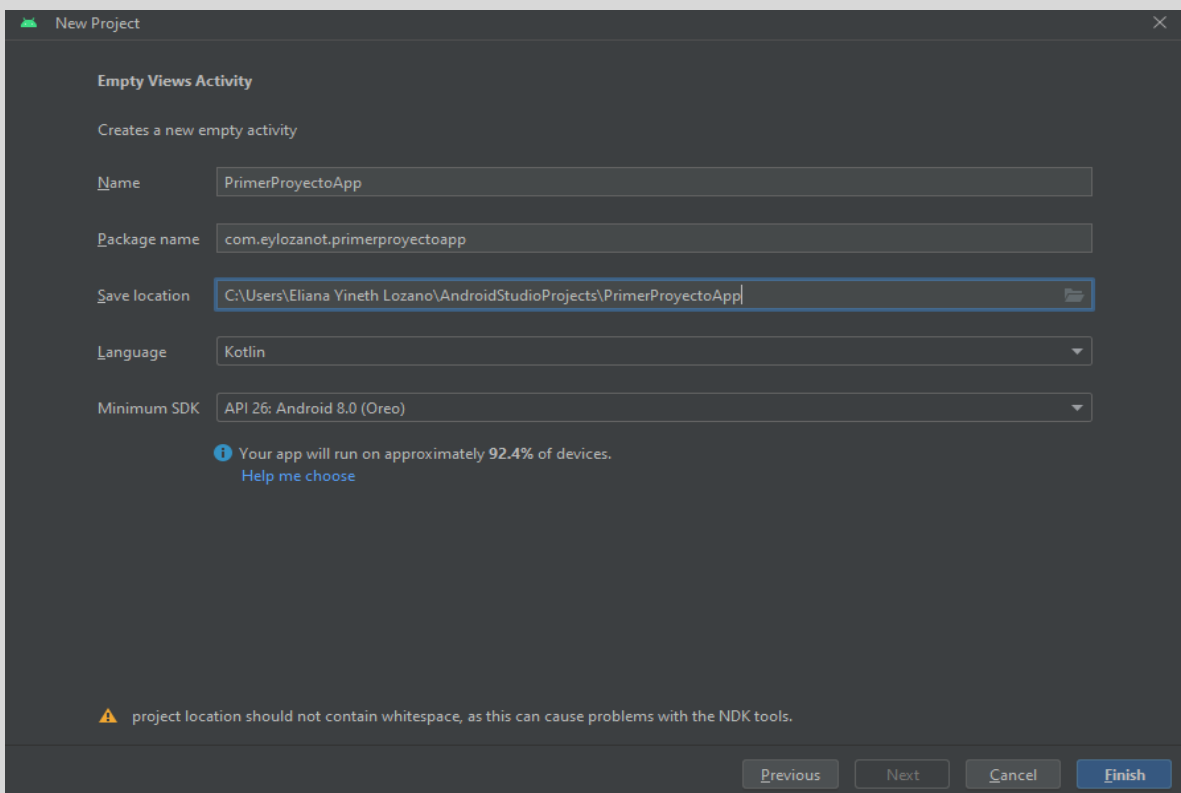
## PRIMER PROYECTO

### Como crear una APP en Android

1. Iniciamos la aplicación de Android Studio y damos clic en New Project.



2. Luego seleccionamos Empty Views Activity y next, daremos nombre a nuestro proyecto: **PrimerProyectoApp**, package name: **com.eylozanot.primerproyectoapp** recordando que este nombre irá en la url de la app, localización, lenguaje de programación: **KOTLIN** y versión de SDK, en esta seleccionaremos la API 26: Android 8.0 (Oreo) y Finish.

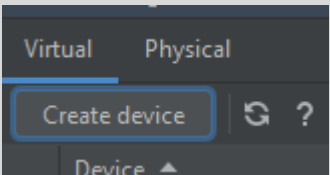
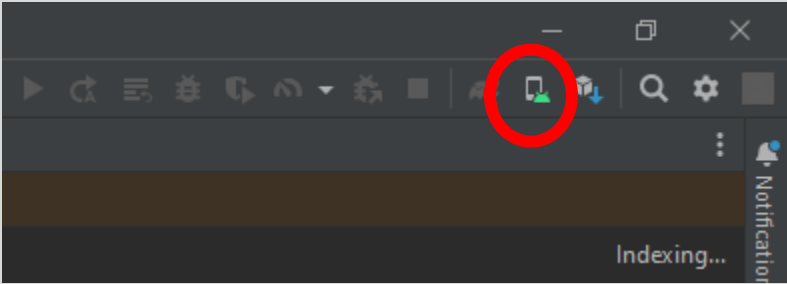




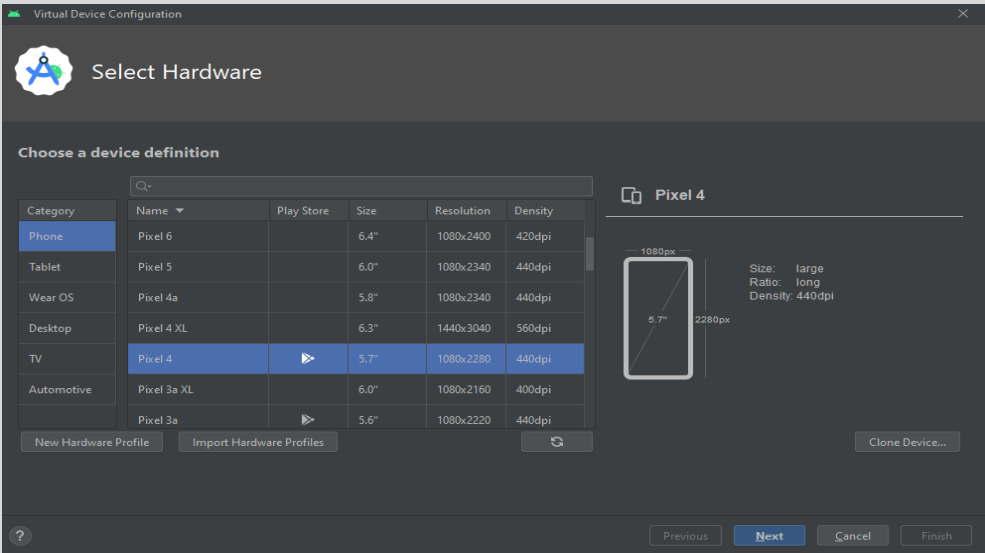
# Mi primera APP - Android Studio/Kotlin

ELIANA YINETH  
LOZANO TRIANA  
eylozano@sena.edu.co

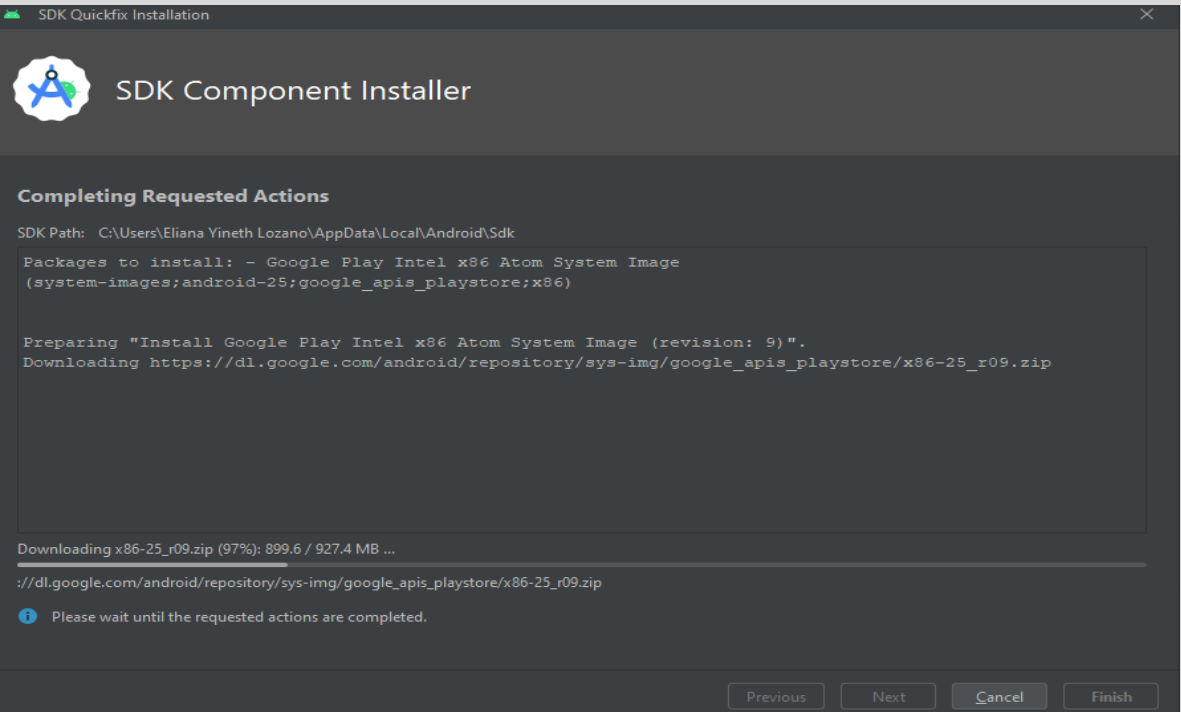
3. Configuraremos nuestro emulador, se asume de que no tenemos configurado uno, por lo que damos clic en Device Manager.



4. Damos clic en Create Device:  
5. Daremos clic en Phone, Pixel 4 y Next.



6. Daremos clic en Oreo como imagen del sistema operativo, aceptar los términos del contrato y Next.



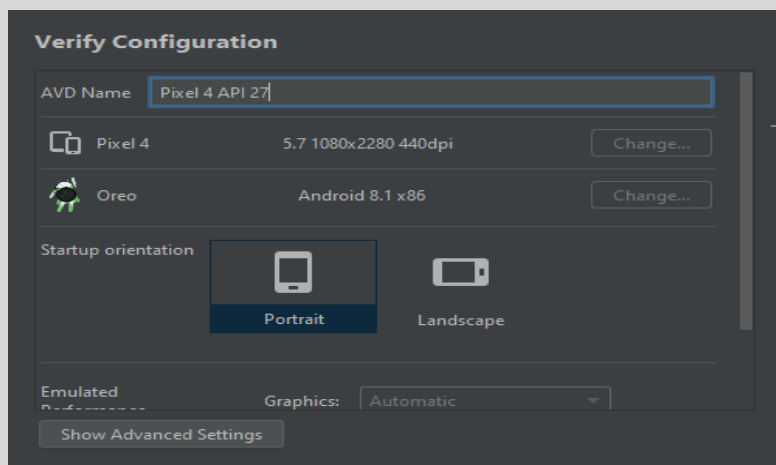


# Mi primera APP - Android Studio/Kotlin

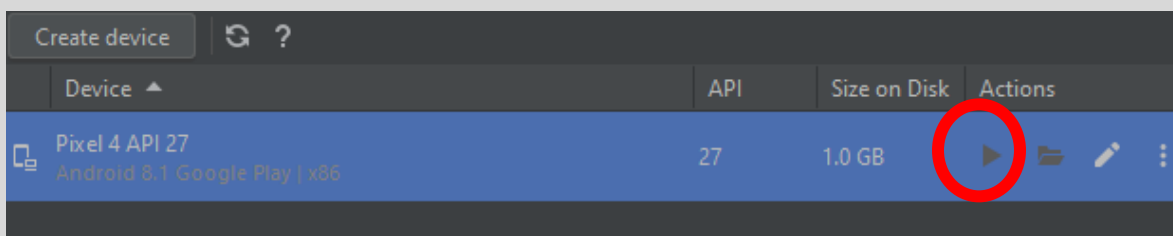
ELIANA YINETH  
LOZANO TRIANA  
eylozano@sena.edu.co

**Nota:** Si los equipos ya tienen un sistema aprovisionado omitamos estos pasos para el emulador.

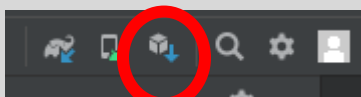
7. Seleccionaremos nuevamente Oreo (el que ya descargamos) y Next.
8. Dejaremos por default la configuración de la siguiente pantalla.



9. Probemos que el emulador quedó debidamente instalado, lo que hacemos es dar clic en el icono de play.



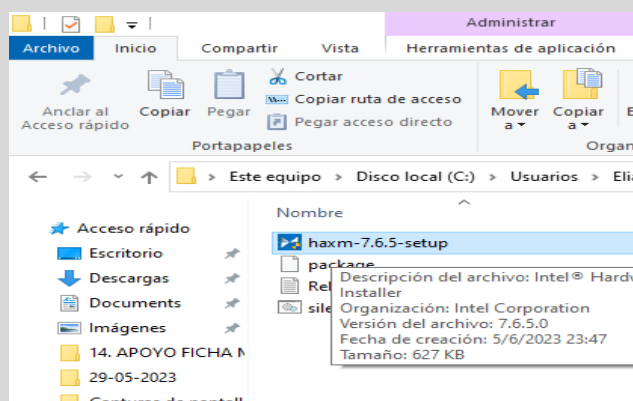
El debería de conectar al emulador después de unos segundos de configuración en caso de que nos aparezca errores como: Hardware acceleration HAXM, lo más común es que se solucione activando la virtualización en la B I O S o que nos dirijamos a SDK Manager para administrar las extensiones o complementos.



Debe de estar seleccionado el Emulator Accelerator si no lo está lo chuleamos y damos Clic en Apply, en caso tal de que, al dar Play para probar nuevamente el emulador, los problemas persistan, cerremos la aplicación y dirijámonos al siguiente directorio:

C:\Users\Eliana Yineth Lozano\AppData\Local\Android\Sdk\extras\intel\Hardware\_Accelerated\_Execution\_Manager

Ajustemos la ruta a la del equipo en uso y ejecutemos la aplicación que se encuentra ahí.





# Mi primera APP - Android Studio/Kotlin

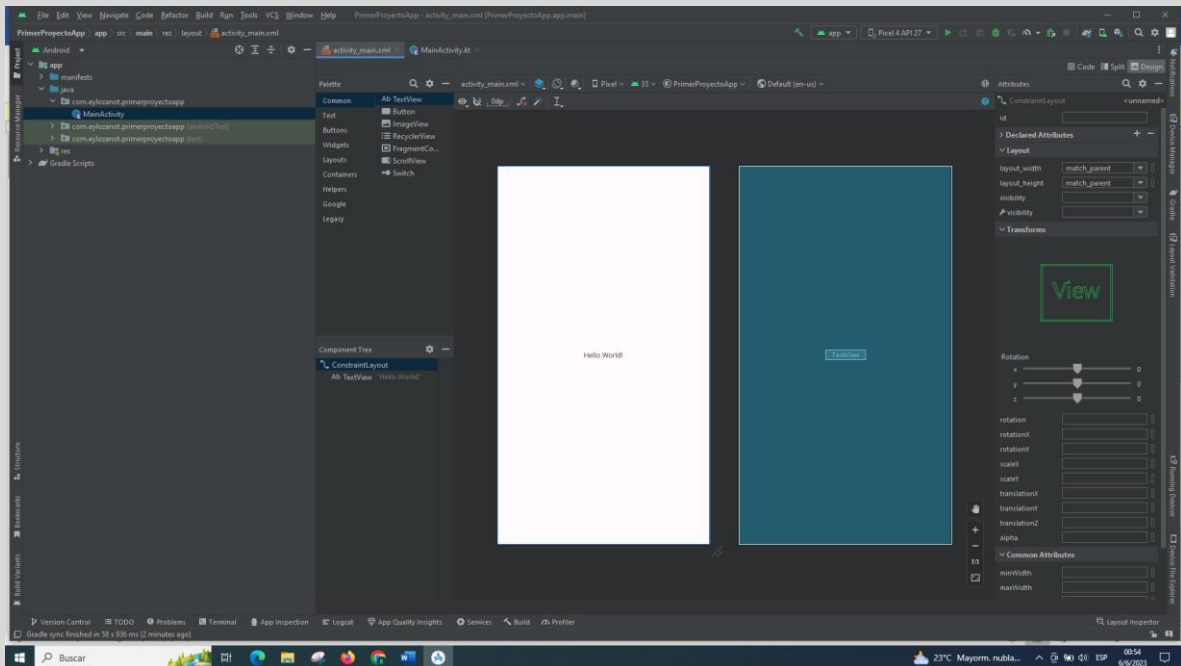
ELIANA YINETH  
LOZANO TRIANA  
eylozano@sena.edu.co

Con esto bastaría para corregir los problemas de emulación.

Sin embargo, tenemos otra opción que es emularla online en el siguiente link:

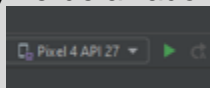
<https://www.apkonline.net/filemanager.php?username=1716342>

10. Ahora tenemos una interfaz gráfica así:

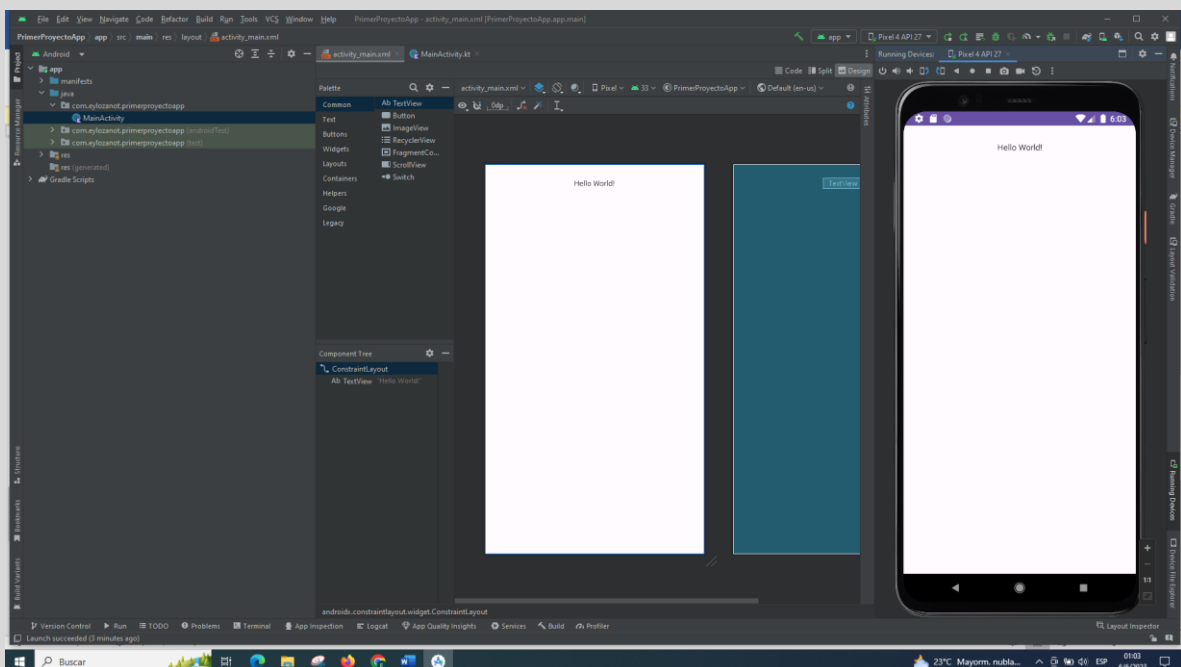


11. Trasladamos hacia arriba nuestro “Hello World”, arrastrándolo.

12. Ahora damos clic en el Play verde al lado del emulador que tenemos creado.



Lo anterior con el fin de dejar corriendo nuestro emulador, sin embargo, lo minimizamos para dejar mas espacio en nuestra pantalla.

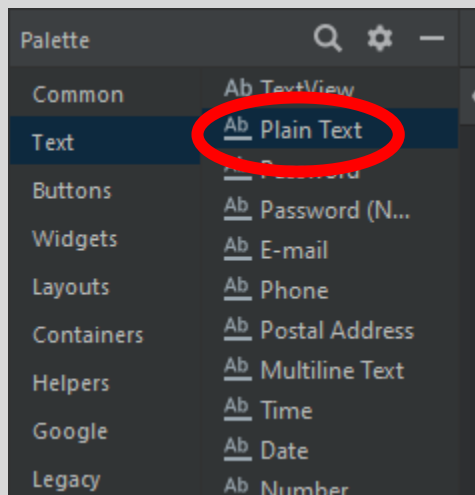




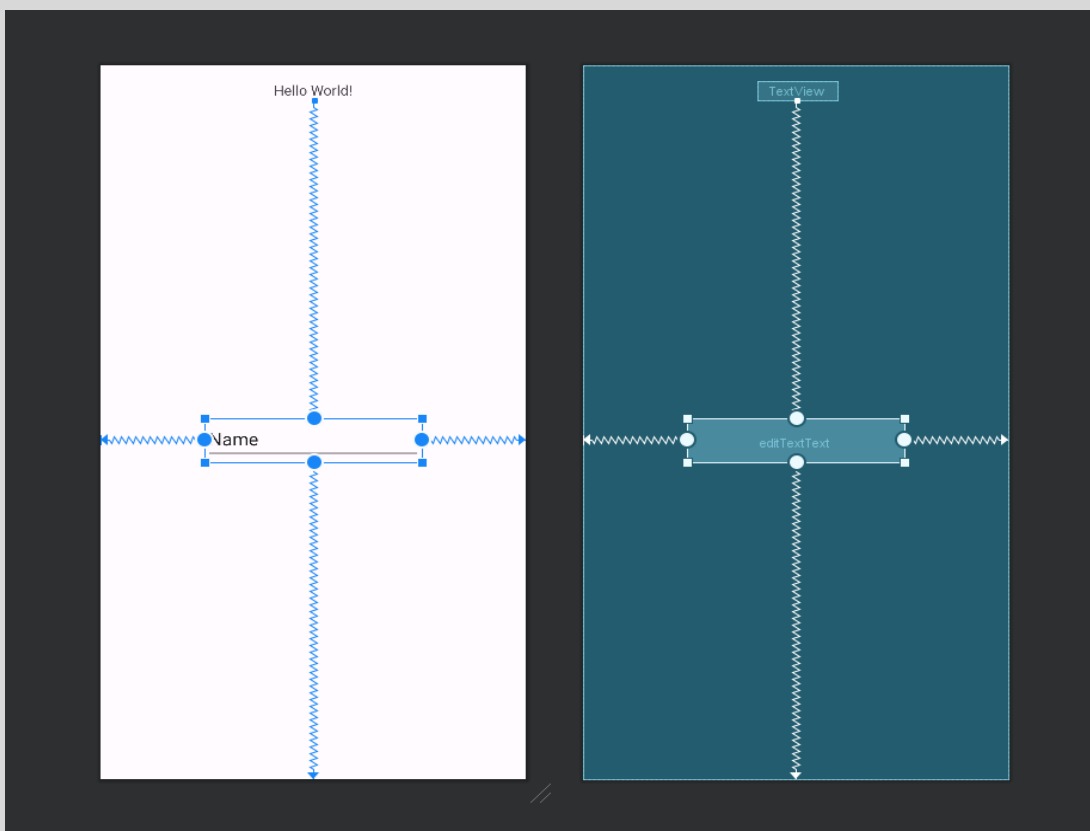
# Mi primera APP - Android Studio/Kotlin

ELIANA YINETH  
LOZANO TRIANA  
eylozano@sena.edu.co

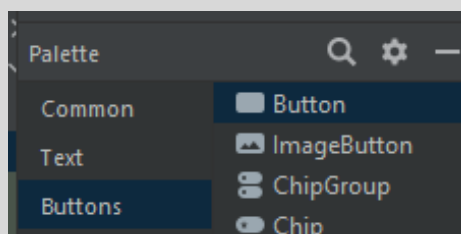
13. Agregaremos un INPUT – Plain Text, el cual encontraremos dando clic en TEXT y lo arrastramos hasta nuestro espacio de trabajo.



14. Nos ubicamos en el árbol de componentes o Component Tree y damos clic en ConstraintLayout que es nuestro gestor de contenido y referenciamos nuestro Input a partir del texto Hello World, desde los puntos azules, esto con el fin de administrar adecuadamente el espacio.



15. Agregaremos un Buttons o botón, el cual encontraremos en el Palette y lo arrastramos hasta nuestro espacio de trabajo.



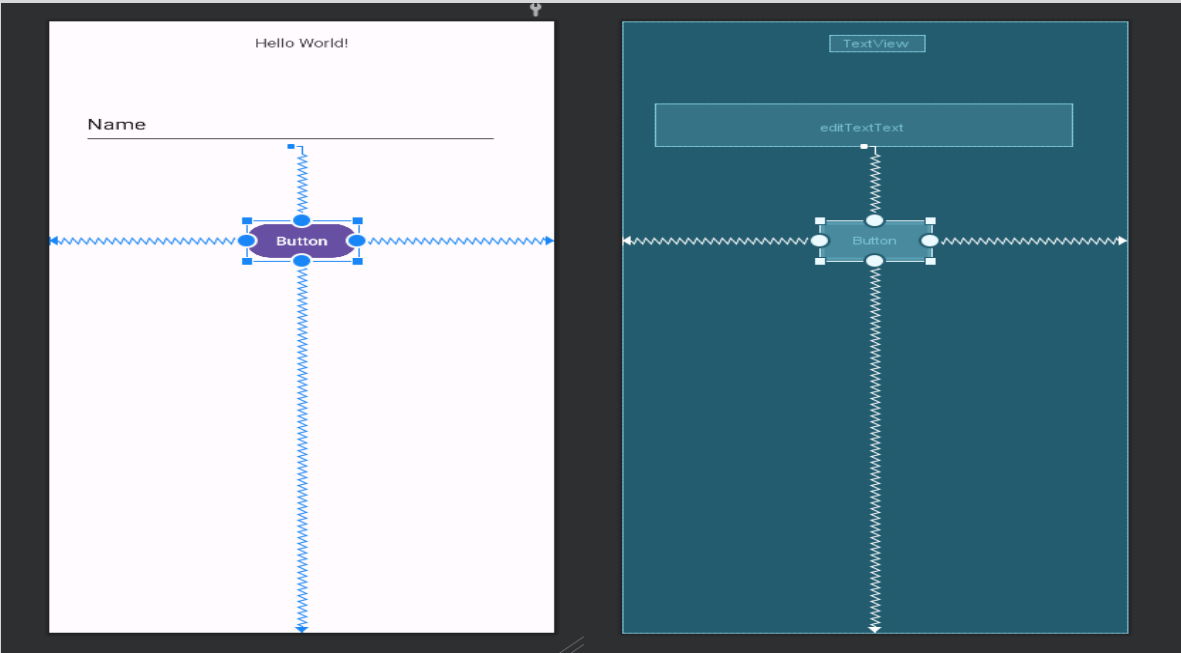
16. De la misma manera referenciaremos, pero al componente superior, es decir



# Mi primera APP - Android Studio/Kotlin

ELIANA YINETH  
LOZANO TRIANA  
eylozano@sena.edu.co

a Name.



17. Ingresaremos a la vista de Split y cambiaremos el Hello World, por algo mas llamativo, en mi caso le cambiaré a “Bienvenid@s Futuros Profesionales!”.

```
<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Bienvenid@s Futuros Profesionales!"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.11" />
```

18. Empezaremos a realizar la identificación y asignación de nombre a nuestros componentes con el fin de que cuando se requieran llamarlos pueda ser fácil ubicarlos, para esto damos clic en Name y así ubicaremos sus propiedades:

```
<EditText
    android:id="@+id/editTextText"
    android:layout_width="339dp"
    android:layout_height="51dp"
    android:ems="10"
    android:inputType="text"
    android:text="Name"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.513"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView"
    app:layout_constraintVertical_bias="0.097" />
```





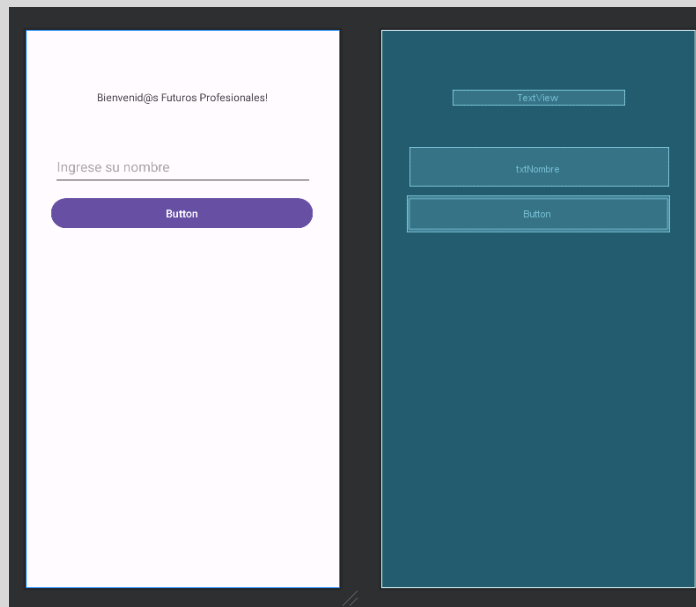
# Mi primera APP - Android Studio/Kotlin

ELIANA YINETH  
LOZANO TRIANA  
eylozano@sena.edu.co

19. El `android:id="@+id/editTextText"` lo cambiaremos a:  
`android:id="@+id/txtNombre"`

```
<EditText  
    android:id="@+id/txtNombre"
```

20. También cambiaremos la propiedad: `android:text="Name"` a:  
`android:hint="Ingrese su nombre"` la propiedad hint lo que hace básicamente es permitirnos tener un texto por debajo en marca de agua, que se borra cuando ingresamos el texto, y va quedando así:

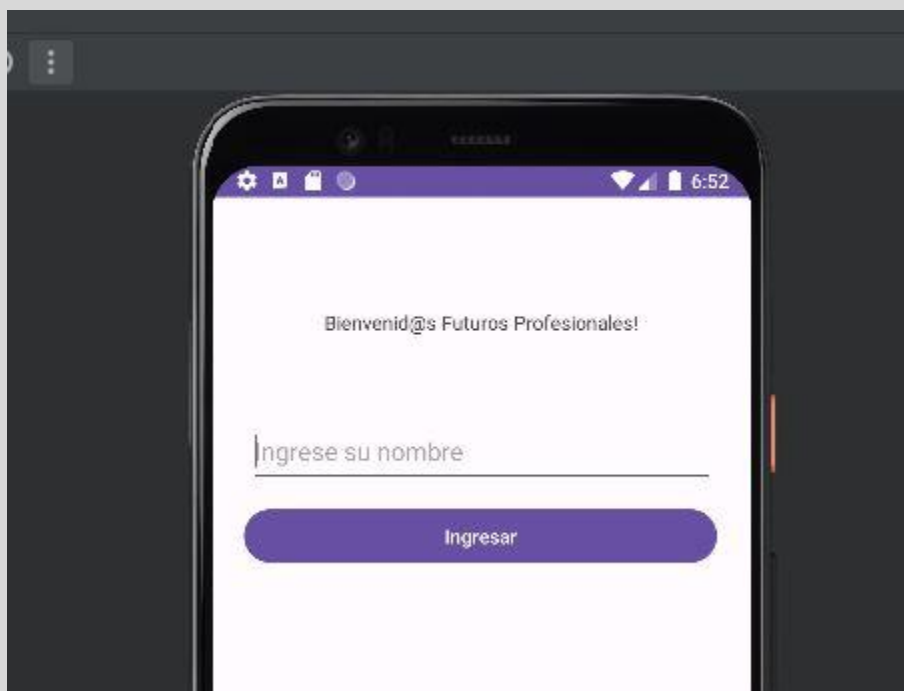


21. Ahora realizaremos el proceso con el Button, cambiando propiedades así:

`android:id="@+id/button3"` a : `android:id="@+id/btnIngresar"`

`android:text="Button"` a: `android:text="Ingresar"`

Y va quedando así:





# Mi primera APP - Android Studio/Kotlin

ELIANA YINETH  
LOZANO TRIANA  
eylozano@sena.edu.co

22. Agreguemos propiedades adicionales al texto de: a “Bienvenid@s Futuros Profesionales!”

- `android:textSize="20sp"` Esto con el fin de agrandar el tamaño de la fuente.
- `android:textColor="@color/material_dynamic_neutral80"` Esto con el fin de cambiar el color de la letra del texto, asignemos el de preferencia.

```
<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Bienvenid@s Futuros Profesionales!"
    android:textSize="20sp"
    android:textColor="@color/material_dynamic_neutral80"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.11" />
```

23. Ahora empecemos a configurar nuestros eventos, para esto nos remitimos a la pestaña MainActivity.kt, y encontramos que ya tenemos referenciado nuestro `Layout.activity_main` dentro de una clase R, que es la que parametriza los diversos componentes.

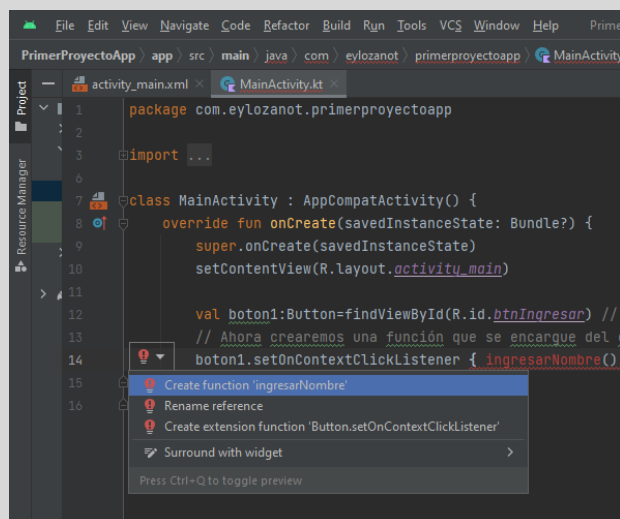
Iremos escribiendo el siguiente código dentro de la Class MainActivity.

```
val boton1:Button=findViewById(R.id.btnIngresar) // Esto con el fin de capturar los ID, con esto ya tendríamos referenciado el botón
```

// Ahora crearemos una función que se encargue del evento de ese botón.

```
boton1.setOnClickListener { ingresarNombre() }
```

// Nos generará un error, daremos clic en el foco o bombillo que se marca rojo y crearemos la función.







# Mi primera APP - Android Studio/Kotlin

ELIANA YINETH  
LOZANO TRIANA  
eylozano@sena.edu.co

```
private fun ingresarNombre() {  
    Toast.makeText(this, "Bienvenid@", Toast.LENGTH_LONG).show()  
}
```

// Lo anterior para capturar el dato, generar un mensaje emergente con el `makeText`, `this` para el contexto de esta clase, el mensaje que queremos, el `LENGTH_LONG` es para que el mensaje sea un poco más largo.

Entonces el código va quedando así:

```
package com.eylozanot.primerproyectoapp
```

```
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.widget.Button  
import android.widget.EditText  
import android.widget.Toast
```

```
class MainActivity : AppCompatActivity() {  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_main)
```

```
        val boton1:Button=findViewById(R.id.btnIngresar)  
        boton1.setOnClickListener { ingresarNombre () }
```

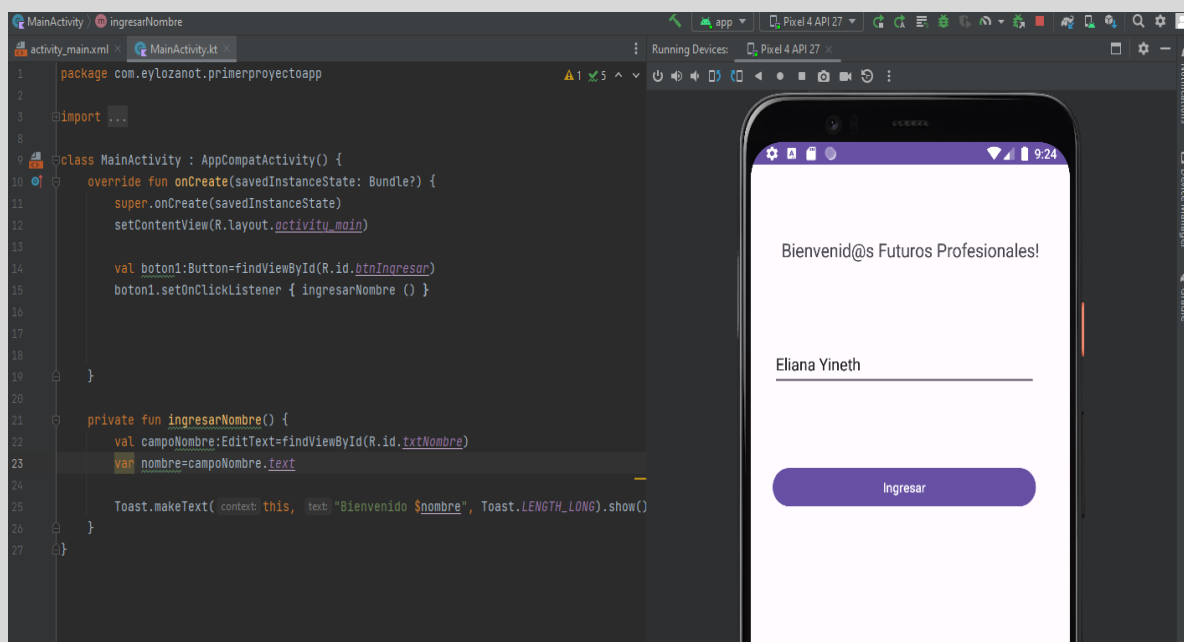
```
    }
```

```
    private fun ingresarNombre() {  
        val campoNombre:EditText=findViewById(R.id.txtNombre)  
        var nombre=campoNombre.text
```

```
        Toast.makeText(this, "Bienvenido $nombre", Toast.LENGTH_LONG).show()
```

```
    }
```

```
}
```





# Mi primera APP - Android Studio/Kotlin

ELIANA YINETH  
LOZANO TRIANA  
eylozano@sena.edu.co

