



# ANDROID NIVEL I LABORATORIO 02

ACTIVIDAD : Practica de Laboratorio Nro. 02

TEMA : Java Code

DOCENTE : Ing. Eric Gustavo Coronel Castillo

DBJETIVO	2
DESCRIPCIÓN	
DESARROLLO	
PASO 01: CREACIÓN DEL PROYECTO	
PASO 02: DISEÑO DEL LAYOUT	
PASO 03: CADENAS	
Paso 04: Código Java	5
PROBLEMAS DE LA APLICACIÓN	
Problema 1	7
Problema 2	7





### **OBJETIVO**

En este segundo laboratorio el objetivo es que el participante se familiarice con el IDE y la escritura de código, tanto XML, como clases y métodos.

# **DESCRIPCIÓN**

Se desarrollara una aplicación muy simple, y luego le aplicaremos una serie de mejoras.

La aplicación consistirá de una única actividad encargada de recoger dos números, A y B, luego calcular su módulo y devolverle el resultado al usuario.

### **DESARROLLO**

## Paso 01: Creación del Proyecto

Crear un nuevo proyecto con las siguientes especificaciones iniciales:

Application Name	Practica02
Company Domain	egcc.uss.pe
Project location	D:\UNI_SIPAN\Ejercicios\Practica02
Minimun SDK	API 16: Android 4.1 (Jelly Bean)

El proyecto debe tener una actividad en blanco con las siguientes especificaciones:

Activity Name	ActividadModulo
Layout Name	actividad_modulo
Title	MODULO





# Paso 02: Diseño del Layout

En el layout agregue los siguientes controles:

Control	ID
TextView	tvA
EditText	etA
TextView	tvB
EditText	etB
TextView	tvResult
Button	btnAction

#### El XML es el siguiente:

```
<LinearLayout
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_alignParentTop="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="New Text"
        android:id="@+id/tvA" />
    <EditText
       android:layout_width="match_parent"
        android:layout height="wrap content"
        android:inputType="number"
        android:ems="10"
        android:id="@+id/etA" />
    <TextView
        android:layout_width="match_parent"
        android:layout height="wrap content"
        android:text="New Text"
        android:id="@+id/tvB" />
```





```
<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/tvResult" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="New Text"
    android:id="@+id/textView3" />

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="New Button"
    android:text="New Button"
    android:id="@+id/btnAction" />

</LinearLayout>
```

#### Paso 03: Cadenas

Se debe definir las siguientes cadenas:

```
<string name="tituloA">Número 1:</string>
<string name="tituloB">Número 2:</string>
```





# Paso 04: Código Java

Es el momento de programar la única actividad que tiene el proyecto.

1. Variables:

```
// Identificador de Clase
private static final String LOG_TAG = "ModActivity";

// Variables privadas
private EditText editTextA;
private EditText editTextB;
private TextView textViewA;
private TextView textViewB;
private TextView result;
private Button action;
```

#### 2. Método modulo

Este método se encarga de encontrar el módulo entre dos números:

```
private int modulo(int a, int b) {
   int r = a % b;
   return r;
}
```

#### 3. Método onCreate

Desde este método se está ejecutando el método progActividad, el objetivo es separar el código del desarrollador en un nuevo método:

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    // Nuevo método
    progActividad();
}

private void progActividad(){
    // En este método se programara la actividad
}
```





4. Mensajes en el LogCat

```
private void progActividad() {
   Log.i(LOG_TAG, "Inicio de progActividad()");

   // resto

Log.i(LOG_TAG, "Fin de progActividad()");
}
```

5. Referencia a los controles del Layout

La programación debe realizarse dentro del método progActividad:

```
textViewA = (TextView) findViewById(R.id.tvA);
textViewB = (TextView) findViewById(R.id.tvB);
editTextA = (EditText) findViewById(R.id.etA);
editTextB = (EditText) findViewById(R.id.etB);
result = (TextView) findViewById(R.id.tvResult);
action = (Button) findViewById(R.id.btnAction);
textViewA.setText(getString(R.string.tituloA));
textViewB.setText(getString(R.string.tituloB));
result.setText("Introduzca valores y pulse el botón.");
```

6. Programar el evento click del botón

```
action.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        String res = "";
        String A = editTextA.getText().toString();
        String B = editTextB.getText().toString();
        int a = Integer.parseInt(A);
        int b = Integer.parseInt(B);
        res = "Resultado = " + modulo(a, b);
        result.setText(res);
    }
});
```

En este momento ya se puede probar el programa.





# PROBLEMAS DE LA APLICACIÓN

#### Problema 1

La aplicación tiene problemas cuando se trata de encontrar el módulo entre cero (0), para solucionar este impase se debe reprogramar el método click del botón como se muestra a continuación:

```
@Override
public void onClick(View view) {
  String res = "";
  String A = editTextA.getText().toString();
  String B = editTextB.getText().toString();
  try {
     int a = Integer.parseInt(A);
     int b = Integer.parseInt(B);
     res = "Resultado = " + modulo(a, b);
  } catch (Exception e) {
     Log.e(LOG_TAG, "Excepción al realizar el módulo de " + A
       + " entre " + B);
     res = "Se produjo una excepción";
     Toast toast = Toast.makeText(getApplicationContext(),
        "Se produjo una excepción.", Toast.LENGTH_LONG);
     toast.show();
  result.setText(res);
```

#### Problema 2

Se debe controlar cuando el usuario ejecuta el botón y no ha ingresado ningún valor.