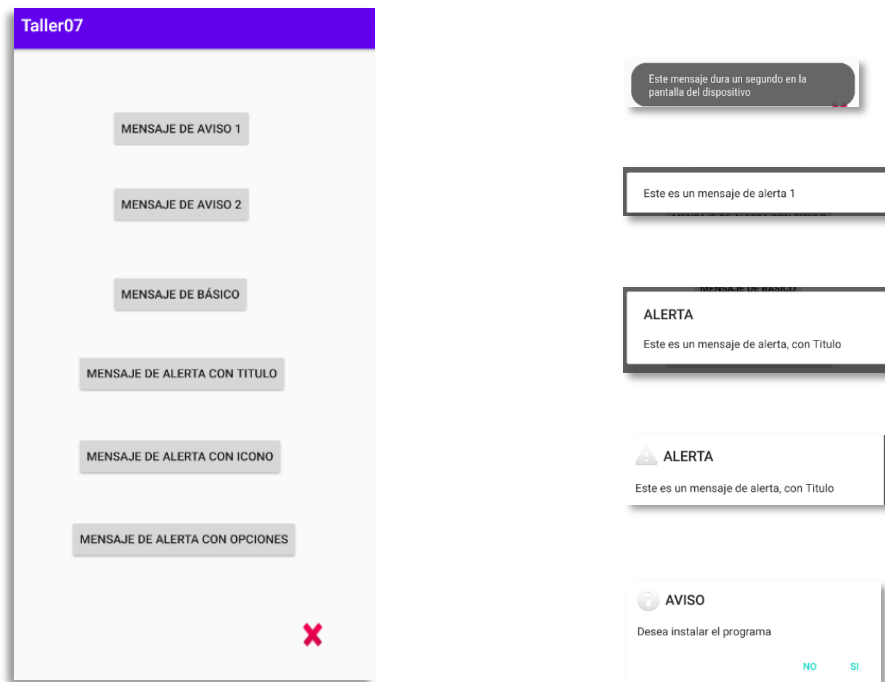


Taller 7: Mensajes y Notificaciones

REQUERIMIENTOS

- ✓ Crear un Proyecto “Empty Activity” (Lollipop 5.0)
- ✓ Insertar los siguientes componentes:
- ✓ 6 Button
- ✓ **Implementar un programa que permita conocer el uso de avisos y mensajes de alertas.**
- ✓ **Realizar un ejemplo del usos de mensajes de alerta X**
- ✓ Agregar un AVD (Virtual Device) para correr la app móvil (optimo)
- ✓ Compilar y Revisar Errores.
- ✓

PANTALLAS



PROCEDIMIENTO

Implementacion con los eventos `setOnClickListener(new View.OnClickListener).....`

Declarar variables propias

Button btn1,btn2,btn3,btn4,btn5,btn6,btn7;

Asociar variables a componentes gráficos

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    btn1 = (Button) findViewById(R.id.button);
    btn2 = (Button) findViewById(R.id.button2);
    btn3 = (Button) findViewById(R.id.button3);
    btn4 = (Button) findViewById(R.id.button4);
    btn5 = (Button) findViewById(R.id.button5);
    btn6 = (Button) findViewById(R.id.button6);
    btn7 = (Button) findViewById(R.id.button7);

```

Poner el código para cada botón antes de las 2 llaves }}
Usar eventyos propios para cada Boton setOnClickListener

BOTON 1

```

btn1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Toast.makeText(getApplicationContext(), "Este mensaje dura medio segundo en la
pantalla del dispositivo", Toast.LENGTH_SHORT).show();
    }
});

```

BOTON 2

```

btn2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Toast.makeText(getApplicationContext(), "Este mensaje dura un segundo en la
pantalla del dispositivo", Toast.LENGTH_SHORT).show();
    }
});

```

BOTON 3

```

btn3.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        AlertDialog.Builder alerta = new AlertDialog.Builder(MainActivity.this);

        alerta.setMessage("Este es un mensaje de alerta 1");
        alerta.show();
    }
});

```

```
}  
});
```

BOTON 4

```
btn4.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        AlertDialog.Builder alerta = new AlertDialog.Builder(MainActivity.this);  
  
        alerta.setTitle("ALERTA");  
        alerta.setMessage("Este es un mensaje de alerta, con Titulo");  
        alerta.show();  
  
    }  
});
```

BOTON 5

```
btn5.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
  
        AlertDialog.Builder alerta = new AlertDialog.Builder(MainActivity.this);  
  
        alerta.setTitle("ALERTA");  
        alerta.setMessage("Este es un mensaje de alerta, con Titulo")  
            .setIcon(android.R.drawable.ic_dialog_alert);  
        alerta.show();  
  
    }  
});
```

BOTON 6

```
btn6.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
  
        AlertDialog.Builder alerta = new AlertDialog.Builder(MainActivity.this);  
  
        alerta.setTitle("AVISO");  
        alerta.setMessage("Desea instalar el programa")  
            .setIcon(android.R.drawable.ic_dialog_info)  
  
            .setCancelable(false)
```

```

        .setPositiveButton("SI", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {

            }
        })
        .setNegativeButton("NO", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {

            }
        });
        alerta.show();
    }
});

```

BOTON 7 CERRAR X

```

btn7.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        AlertDialog.Builder alerta = new AlertDialog.Builder(MainActivity.this);

        alerta.setTitle("AVISO");
        alerta.setMessage("Desea salir de la aplicación")
            .setIcon(android.R.drawable.ic_dialog_alert)

        .setCancelable(false)
        .setPositiveButton("SI", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {

                System.exit(0);

            }
        })
        .setNegativeButton("NO", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {

            }
        });
        alerta.show();
    }
});

```

```
}  
});
```

```
////notificación snack  
Snackbar.make(view, "Replace with your own action", Snackbar.LENGTH_LONG)  
    .setAction("Action", null).show();
```

Taller 8: Estructuras de Control Condicionales "IF" (MANEJO DE SPINNER)

REQUERIMIENTOS

- ✓ Crear un Proyecto "Empty Activity" (Lollipop 5.0)
- ✓ Insertar los siguientes componentes:
 - 5 TextView (labels)
 - 2 plainText (edit tex - number)
 - 1 Spinner (grupo Container)
 - 3 Buttons
- ✓ **Implementar un programa que permita realizar las operaciones indicadas, seleccionada desde un desplegable (spinner).**
- ✓ **Explicar y conocer el uso del componente spinner**
- ✓ **Hacer énfasis en las estructuras de control-condiconal**
- ✓ **Usar avisos o mensajes de alerta para controlar errores.**
- ✓ Agregar un AVD (Virtual Device) para correr la app móvil (optimo)
- ✓ Compilar y Revisar Errores.

PANTALLAS

Taller08

OPERACIONES

Valor 1:

#. #

Valor 2:

#. #

Seleccione la Operación:

Sumar

CALCULAR

Resultado:

RESET

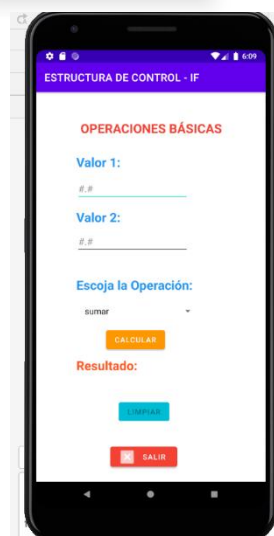
Escoja la Operación:

sumar

restar

multiplicar

dividir



PROCEDIMIENTO

Declarar variables propias

```
Button btn1,btn2, btnsalir;  
TextView lbl1, lbl2 ,lbl3 ,lbl4 , lblresultado;  
EditText txt1, txt2;  
Spinner spn1; @Override
```

Asociar variables a componentes gráficos

```
//titulo  
getSupportActionBar().setTitle("ESTRUCTURA DE CONTROL - IF");
```

//LOGICA VS GRAFICA

```
btn1=(Button)findViewById(R.id.button);  
btn2=(Button)findViewById(R.id.button2);  
btnsalir =(Button)findViewById(R.id.button3);  
  
lbl1 =(TextView)findViewById(R.id.textView);  
lbl2 =(TextView)findViewById(R.id.textView2);  
lbl3 =(TextView)findViewById(R.id.textView3);  
lbl4 =(TextView)findViewById(R.id.textView4);  
lblresultado =(TextView)findViewById(R.id.textView5);  
  
txt1=(EditText)findViewById(R.id.editTextNumber);  
txt2=(EditText)findViewById(R.id.editTextNumber2);  
  
spn1 = (Spinner)findViewById(R.id.spinner);
```

//cargar datos en el spinner

```
String [] operaciones = {"sumar", "restar", "multiplicar", "dividir"};  
ArrayAdapter <String> adapter = new  
ArrayAdapter<String>(MainActivity.this,R.layout.support_simple_spinner_dropdown_item,operaciones);  
spn1.setAdapter(adapter);
```

Implementacion con los eventos `setOnClickListener(new View.OnClickListener).....`

BOTON 1

```
btn1.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {
```

```

float x1,x2,result;

if( (txt1.getText().toString().equals("")) || (txt2.getText().toString().equals("")) )
{
    Toast.makeText(MainActivity.this, "Existen campos vacios!!", Toast.LENGTH_SHORT).show();
    lblresultado.setText("Resultado:");
}
else
{
    x1 = Float.valueOf(txt1.getText().toString());
    x2 = Float.valueOf(txt2.getText().toString());

    //operacion suma
    if(spn1.getSelectedItem().toString().equals("sumar"))
    {
        resul = x1+x2;
        lblresultado.setText("Resultado: "+resul);

        //deshabilito
        txt1.setEnabled(false);
        txt2.setEnabled(false);
        btn1.setEnabled(false);
        btn2.setEnabled(true);
    }

    //operacion resta
    if(spn1.getSelectedItem().toString().equals("restar"))
    {
        resul = x1-x2;
        lblresultado.setText("Resultado: "+resul);

        //deshabilito
        txt1.setEnabled(false);
        txt2.setEnabled(false);
        btn1.setEnabled(false);
        btn2.setEnabled(true);
    }

    //operacion multiplicacion
    if(spn1.getSelectedItem().toString().equals("multiplicar"))
    {
        resul = x1*x2;
        lblresultado.setText("Resultado: "+resul);

        //deshabilito
        txt1.setEnabled(false);
        txt2.setEnabled(false);
        btn1.setEnabled(false);
        btn2.setEnabled(true);
    }

    //operacion division
    if(spn1.getSelectedItem().toString().equals("dividir"))
    {
        if(txt2.getText().toString().equals("0"))
        {
            Toast.makeText(MainActivity.this, "Imposible operar con divisor 0", Toast.LENGTH_SHORT).show();

```

```

        lblresultado.setText("");
    }
    else
    {
        resul = x1/x2;
        lblresultado.setText("Resultado: "+resul);

        //deshabilito
        txt1.setEnabled(false);
        txt2.setEnabled(false);
        btn1.setEnabled(false);
        btn2.setEnabled(true);

    }

}

}

}

}

});

```

BOTON 2 – limpiar

```

btn2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        txt1.setText("");
        txt2.setText("");
        lblresultado.setText("Resultado:");

        txt1.setEnabled(true);
        txt1.requestFocus();
        txt1.setEnabled(true);
        btn1.setEnabled(true);

        btn2.setEnabled(false);

    }
});

```

BOTON 2 – SALIR

```

//boton salir
btnsalir.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

```

```
AlertDialog.Builder alerta = new AlertDialog.Builder(MainActivity.this);
alerta.setTitle("Aviso");
alerta.setMessage("Desea Cerrar la Aplicación");
alerta.setIcon(android.R.drawable.ic_dialog_alert);
alerta.setCancelable(false);
alerta.setPositiveButton("Si", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int which) {
        System.exit(0);
    }
});

alerta.setNegativeButton("No", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int which) {

    }
});

alerta.show();

}
});
```

Taller 9: Estructuras de Control Alternativas “switch” (*manejo de radiobuttons*)

REQUERIMIENTOS

- ✓ Crear un Proyecto “Empty Activity” (Lollipop 5.0)
- ✓ Insertar los siguientes componentes:
 - 5 TextView (labels)
 - 2 plainText (edit tex)
 - 1 GroupRadio
 - 4 Radio Buttons
 - 2 Buttons
- ✓ **Implementar un programa que permita realizar las operaciones indicadas, seleccionada desde los RadioButtons.**
- ✓ **Explicar y conocer el uso del componente RadioButtons**
- ✓ **Hacer énfasis en las estructuras de control-alternativas**
- ✓ **Usar avisos o mensajes de alerta para controlar errores.**
- ✓ Agregar un AVD (Virtual Device) para correr la app móvil (optimo)
- ✓ Compilar y Revisar Errores.

PANTALLAS

The screenshot shows a mobile application interface titled "Taller09". The main heading is "OPERACIONES" in red. Below it, there are two input fields labeled "Valor 1:" and "Valor 2:", each with a placeholder "#.#". Underneath these is a label "Marque la Operación_" followed by four radio button options: "Suma", "Resta", "Multiplicación", and "División". At the bottom of the input section is a grey button labeled "CALCULAR". Below the calculator section, there is a red label "Resultado:" and a grey button labeled "RESET".

PROCEDIMIENTO

IMPORTANTE

//poner los componentes indicados

Agregar primero el RadioGroup y dentro los RadioButton

Al RadioGroup hay q ponerle un ID manual

Declarar variables propias

```
Button btn1,btn2;  
EditText txt1,txt2;  
TextView lbl1,lbl2,lbl3,lbl4,lblrespuesta;  
RadioButton rbtn1, rbtn2, rbtn3, rbtn4;  
RadioGroup rbng1;
```

int op;

@Override

Asociar variables a componentes gráficos

//titulo

```
getSupportActionBar().setTitle("ESTRUCTURA DE CONTROL - IF");
```

//LOGICA VS GRAFICA

```
btn1 = (Button) findViewById(R.id.button);  
btn2 = (Button) findViewById(R.id.button2);  
  
txt1 = (EditText) findViewById(R.id.editTextTextPersonName);  
txt2 = (EditText) findViewById(R.id.editTextTextPersonName2);  
  
lbl1= (TextView) findViewById(R.id. textView);  
lbl2= (TextView) findViewById(R.id. textView2);  
lbl3= (TextView) findViewById(R.id. textView5);  
lbl4= (TextView) findViewById(R.id. textView3);  
lblrespuesta = (TextView) findViewById(R.id. textView4);  
  
rbtn1 = (RadioButton) findViewById(R.id. radioButton);  
rbtn2 = (RadioButton) findViewById(R.id. radioButton2);  
rbtn3 = (RadioButton) findViewById(R.id. radioButton3);  
rbtn4 = (RadioButton) findViewById(R.id. radioButton4);  
  
rbng1 = (RadioGroup) findViewById(R.id.rg);
```

```
txt1.requestFocus();
btn2.setEnabled(false);
rbng1 = (RadioGroup) findViewById(R.id.rg);
```

//no olvidar los inicios por defecto de algunos componentes

Implementacion con los eventos `setOnClickListener(new View.OnClickListener).....`

CLICK RadioButton ***Capturar el click en los radioButons**

```
rbtn1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        op=1;

        // Toast.makeText(getApplicationContext(),"Op =" + op, Toast.LENGTH_SHORT).show();
    }
});
```

```
rbtn2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        op=2;

        //Toast.makeText(getApplicationContext(),"Op =" + op, Toast.LENGTH_SHORT).show();
    }
});
```

```
rbtn3.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        op=3;

        //Toast.makeText(getApplicationContext(),"Op =" + op, Toast.LENGTH_SHORT).show();
    }
});
```

```
rbtn4.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        op=4;

        //Toast.makeText(getApplicationContext(),"Op =" + op, Toast.LENGTH_SHORT).show();
    }
});
```

```
});
```

BOTON 1

```
btn1.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
  
        if(txt1.getText().toString().isEmpty() || txt2.getText().toString().isEmpty())  
        {  
            Toast.makeText(getApplicationContext(), "Existen Campos Vacios", Toast.LENGTH_SHORT).show();  
        }  
        else {  
  
            float num1 = Float.parseFloat(txt1.getText().toString());  
            float num2 = Float.parseFloat(txt2.getText().toString());  
            switch (op) {  
                case 1: {  
  
                    float r = num1+num2;  
                    lblrespuesta.setText("Resultado: "+ r);  
  
                    txt1.setEnabled(false);  
                    txt2.setEnabled(false);  
                    btn1.setEnabled(false);  
                    btn2.setEnabled(true);  
                    rbtn1.setEnabled(false);  
                    rbtn2.setEnabled(false);  
                    rbtn3.setEnabled(false);  
                    rbtn4.setEnabled(false);  
  
                    break;  
                }  
  
                case 2: {  
  
                    float respuesta= num1-num2;  
                    lblrespuesta.setText("Resultado: "+ respuesta);  
  
                    txt1.setEnabled(false);  
                    txt2.setEnabled(false);  
                    btn1.setEnabled(false);  
                    btn2.setEnabled(true);  
                    rbtn1.setEnabled(false);  
                    rbtn2.setEnabled(false);  
                    rbtn3.setEnabled(false);  
                    rbtn4.setEnabled(false);  
  
                    break;  
                }  
  
                case 3: {  
  
                    float respuesta=num1*num2;  
                    lblrespuesta.setText("Resultado: "+ respuesta);
```

```

        txt1.setEnabled(false);
        txt2.setEnabled(false);
        btn1.setEnabled(false);
        btn2.setEnabled(true);
        rbtn1.setEnabled(false);
        rbtn2.setEnabled(false);
        rbtn3.setEnabled(false);
        rbtn4.setEnabled(false);

        break;
    }

    case 4: {

        if( Float.parseFloat(txt2.getText().toString()) == 0.0 )
        {
            Toast.makeText(getApplicationContext(), "División para cero", Toast.LENGTH_SHORT).show();
            lblrespuesta.setText("Resultado: ");
        }
        else {

            float respuesta = num1 / num2;
            lblrespuesta.setText("Resultado: " + respuesta);
            txt1.setEnabled(false);
            txt2.setEnabled(false);
            btn1.setEnabled(false);
            btn2.setEnabled(true);
            rbtn1.setEnabled(false);
            rbtn2.setEnabled(false);
            rbtn3.setEnabled(false);
            rbtn4.setEnabled(false);

        }
        break;
    }

    default: {
        Toast.makeText(getApplicationContext(), "Escoja la Operación", Toast.LENGTH_SHORT).show();
        break;
    }
}
}
});

```

BOTON 2 – limpiar

```

btn2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        txt1.setText("");
        txt2.setText("");
        lblrespuesta.setText("Resultado:");
    }
});

```

```

txt1.setEnabled(true);
txt1.requestFocus();
txt2.setEnabled(true);

btn1.setEnabled(true);
btn2.setEnabled(false);
rbtn1.setEnabled(true);
rbtn2.setEnabled(true);
rbtn3.setEnabled(true);
rbtn4.setEnabled(true);

rbtn1.setChecked(false);
rbtn2.setChecked(false);
rbtn3.setChecked(false);
rbtn4.setChecked(false);

op=0;

}
});

```

BOTON 2 – SALIR

```

//boton salir
btnsalir.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        AlertDialog.Builder alerta = new AlertDialog.Builder(MainActivity.this);
        alerta.setTitle("Aviso");
        alerta.setMessage("Desea Cerrar la Aplicación");
        alerta.setIcon(android.R.drawable.ic_dialog_alert);
        alerta.setCancelable(false);
        alerta.setPositiveButton("Si", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                System.exit(0);
            }
        });

        alerta.setNegativeButton("No", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {

            }
        });
    }
});

```

```
alerta.show();
```

```
    }  
});
```

Taller 10: Estructuras de Control Repetitivas “for, while, do while” *(manejo de checkbox)*

REQUERIMIENTOS

- ✓ Crear un Proyecto “Empty Activity” (Lollipop 5.0)
- ✓ Insertar los siguientes componentes:
 - 8 TextView (labels)
 - 2 plainText (edit tex)
 - 4 CheckBox
 - 2 Buttons
- ✓ *Implementar un programa que permita realizar las operaciones indicadas, seleccionada desde los CheckBox.*
- ✓ *Explicar y conocer el uso del componente CheckBox*
- ✓ *Hacer énfasis en las estructuras de control-repetitivas*
- ✓ *Usar avisos o mensajes de alerta para controlar errores.*
- ✓ Agregar un AVD (Virtual Device) para correr la app móvil (optimo)
- ✓ Compilar y Revisar Errores.

PANTALLAS

The screenshot shows a mobile application titled 'Taller10'. The interface is divided into two main sections: 'CÁLCULOS' and 'OPERACIONES'. Under 'CÁLCULOS', there are two input fields labeled 'Valor A:' and 'Valor B:', each with a placeholder text '# entero'. Under 'OPERACIONES', there are four checkboxes, each followed by a label and a red 'R:' indicator. The checkboxes are: 'Potencia: A^B', 'Potencia: B^A', 'Factorial: A!', and 'Factorial: B!'. At the bottom of the screen, there are two buttons: 'CALCULAR' and 'RESET'.

PROCEDIMIENTO

IMPORTANTE

Declarar variables propias

*****//Crear variables propias

TextView lbl1, lbl2, lbl3, lbl4, lblresp1, lblresp2, lblresp3, lblresp4;

Button btn1, btn2;

EditText txt1, txt2;

CheckBox chb1, chb2, chb3, chb4;

@Override

Asociar variables a componentes gráficos

//titulo

getSupportActionBar().setTitle("ESTRUCTURA DE CONTROL - IF");

//LOGICA VS GRAFICA

lbl1 = (TextView) findViewById(R.id.textView);

lbl2 = (TextView) findViewById(R.id.textView2);

lbl3 = (TextView) findViewById(R.id.textView3);

lbl4 = (TextView) findViewById(R.id.textView4);

lblresp1 = (TextView) findViewById(R.id.textView5);

lblresp2 = (TextView) findViewById(R.id.textView6);

lblresp3 = (TextView) findViewById(R.id.textView7);

lblresp4 = (TextView) findViewById(R.id.textView8);

btn1 = (Button) findViewById(R.id.button);

btn2 = (Button) findViewById(R.id.button2);

txt1 = (EditText) findViewById(R.id.editTextTextPersonName);

txt2 = (EditText) findViewById(R.id.editTextTextPersonName2);

chb1 = (CheckBox) findViewById(R.id.checkBox);

chb2 = (CheckBox) findViewById(R.id.checkBox2);

chb3 = (CheckBox) findViewById(R.id.checkBox3);

chb4 = (CheckBox) findViewById(R.id.checkBox4);

//no olvidar los inicios por defecto de algunos componentes

Implementacion con los eventos **setOnClickListener(new View.OnClickListener).....**

BOTON 1

```
btn1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        if (txt1.getText().toString().isEmpty() || txt2.getText().toString().isEmpty()) {
            Toast.makeText(getApplicationContext(), "Existen Campos Vacios",
            Toast.LENGTH_SHORT).show();
        }
        else
        {

            if( chb1.isChecked())
            {
                ///// Exponente A:B
                int base,exponente;
                base = Integer.parseInt(txt1.getText().toString());
                exponente = Integer.parseInt(txt2.getText().toString());

                int acum1=1;
                for(int i=1; i<=exponente; i++)
                {
                    acum1=acum1*base;
                }
                lblresp1.setText("R: "+acum1);

            }

            if( chb2.isChecked())
            {
                ///// Exponente B:A
                int base2,exponente2;
                base2 = Integer.parseInt(txt1.getText().toString());
                exponente2 = Integer.parseInt(txt2.getText().toString());

                int acum2=1;
                for(int i=1; i<=base2; i++)
                {
                    acum2=acum2*exponente2;
                }
                lblresp2.setText("R: "+acum2);

            }

        }
    }
});
```

```

if( chb3.isChecked())
{
    ///// Factorial A
    int numero;
    numero = Integer.parseInt(txt1.getText().toString());

    int cont=1;
    int acum3=1;
    while(cont<= numero)
    {
        acum3 = acum3*cont;
        cont++;
    }
    lblresp3.setText("R: "+acum3);

}

if( chb4.isChecked())
{
    //// Factorial b

    int numero2;
    numero2 = Integer.parseInt(txt2.getText().toString());

    int cont=1;
    int acum4=1;
    do
    {
        acum4 = acum4*cont;
        cont++;
    }
    while(cont<= numero2);

    lblresp4.setText("R: "+acum4);

}

if (chb1.isChecked()== false && chb2.isChecked()== false && chb3.isChecked()== false
&& chb4.isChecked()== false )
{
    Toast.makeText(getApplicationContext(), "Marque Operación a realizar",
    Toast.LENGTH_SHORT).show();
}

```

```
    } //fin primer if
```

```
    }  
    });
```

BOTON 2 – limpiar

```
btn2.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
  
        txt1.setText("");  
        txt2.setText("");  
        txt1.requestFocus();  
  
        lblresp1.setText("R:");  
        lblresp2.setText("R:");  
        lblresp3.setText("R:");  
        lblresp4.setText("R:");  
  
        chb1.setChecked(false);  
        chb2.setChecked(false);  
        chb3.setChecked(false);  
        chb4.setChecked(false);  
    }  
});
```

BOTON 2 – SALIR

```
//boton salir  
btnsalir.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
  
        AlertDialog.Builder alerta = new AlertDialog.Builder(MainActivity.this);  
        alerta.setTitle("Aviso");  
        alerta.setMessage("Desea Cerrar la Aplicación");  
        alerta.setIcon(android.R.drawable.ic_dialog_alert);  
        alerta.setCancelable(false);  
        alerta.setPositiveButton("Si", new DialogInterface.OnClickListener() {  
            @Override  
            public void onClick(DialogInterface dialog, int which) {  
                System.exit(0);  
            }  
        });  
    }  
});
```

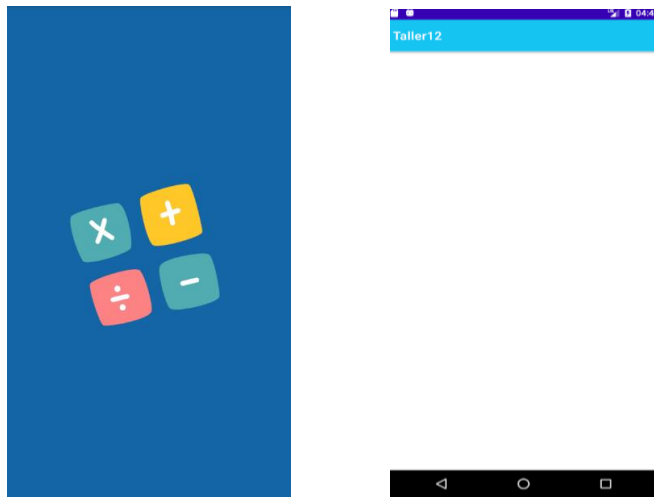
```
    }  
    });  
  
    alerta.setNegativeButton("No", new DialogInterface.OnClickListener() {  
        @Override  
        public void onClick(DialogInterface dialog, int which) {  
  
        }  
    });  
  
    alerta.show();  
  
    }  
});
```

Taller 11: Splash (Pantalla de Inicio)

REQUERIMIENTOS

- ✓ Crear un Proyecto "Empty Activity" (Lollipop 5.0)
- ✓ Crear +2 Activity(Layout)
- ✓ Insertar los siguientes componentes MainActivity1:
 - 1 ImageView
 - 1 TextView (labels)
- ✓ MainActivity2 :
 - 1 TextView (labels)
 - 1 Button Salir
- ✓ **Implementar un aplicación que permite mostrar un splash de bienvenida aproximadamente por 2 segundos y luego cargue la pantalla principal de la APP**
- ✓ Agregar un AVD (Virtual Device) para correr la app móvil (optimo)
- ✓ Compilar y Revisar Errores.

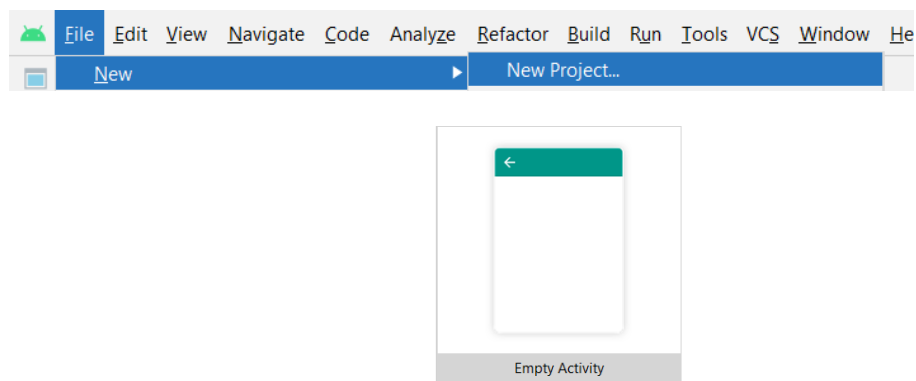
PANTALLAS



PROCEDIMIENTO

Crear el MainActivity2

File > New > Project



En MainActivity1

Declarar variables propias

Asociar variables a componentes gráficos

@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main1);  
}
```


//funcion para quitar Action Bar

```
getSupportActionBar().hide;
```

//Evento para el Timer

```
new Handler().postDelayed(new Runnable() {  
    @Override  
    public void run() {  
  
        Intent menu = new Intent(MainActivity4.this, MainActivity.class);  
        startActivity(menu);  
        finish();  
    }  
},2000);
```

////////IMPORTANTE////////

En el caso de que el Layout splash no sea el primer Activity se debe proseguir de la siguiente manera:

1. Entrar al archivo **AndroidManifest.xml**; buscar la activity con la que quiere arrancar la app (splash)
2. Cortar el siguiente código que le corresponde a la **MainActivity1**

<intent-filter>

```
<action android:name="android.intent.action.MAIN" />
```

```
<category android:name="android.intent.category.LAUNCHER" />
```

</intent-filter>

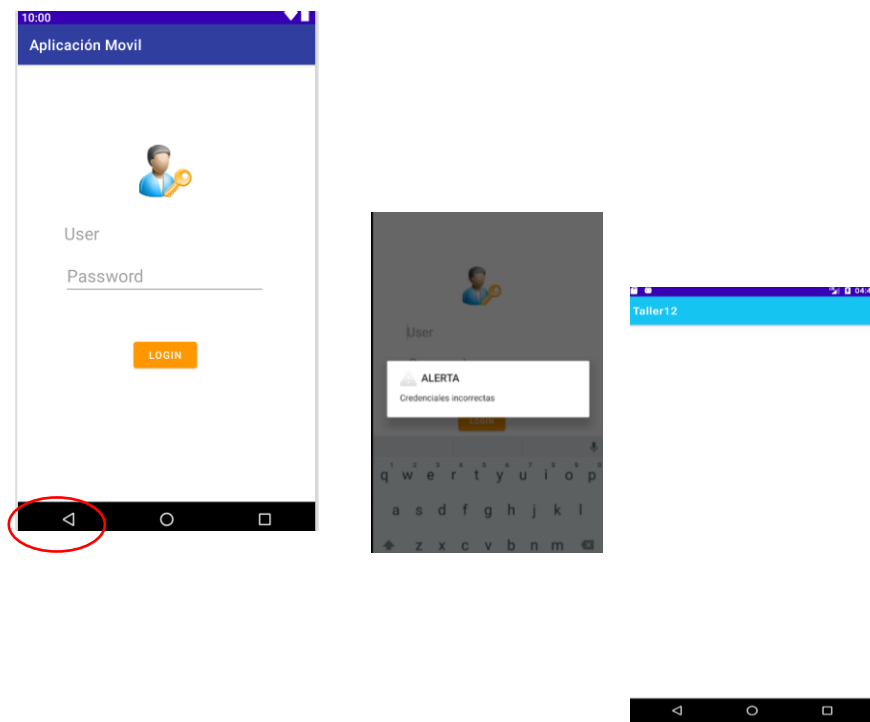
1. Y colocarlo dentro de la **activity** con la que desea inciar
2. <activity android:name=".MainActivity4"> ****aqui*** </activity>

Taller 12: Autenticación (Login y Password)

REQUERIMIENTOS

- ✓ Crear un Proyecto "Empty Activity" (Lollipop 5.0)
- ✓ Insertar los siguientes componentes **MainActivity**:
 - 1 ImageView (Widgets)
 - 2 plainText (edit tex)
 - 1 Buttons
- ✓ Crear **MainActivity2** :
 - 1 TextView (labels)
- ✓ *Implementar una aplicación que permita validar el ingreso con sus credenciales antes de ir a la pantalla principal.*
- ✓ *Explicar y conocer el propiedades de Barra de Acción*
- ✓ *Usar avisos o mensajes de alerta para controlar errores.*
- ✓ *Progamar el evento atrás de la barra de Navegación ◀*
- ✓ Agregar un AVD (Virtual Device) para correr la app móvil (optimo)
- ✓ Compilar y Revisar Errores.
- ✓

PANTALLAS



PROCEDIMIENTO

MainActivity1 (Login)

Declarar variables propias

```
Button btn1;  
EditText txt1;  
EditText txt2;  
@Override
```

Asociar variables a componentes gráficos

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main2);
```

//ACCIONES PARA EL ACTION BAR

```
getSupportActionBar().setTitle(" ACCESO"); //titulo en la barra de acción  
getSupportActionBar().setDisplayHomeAsUpEnabled(true); // opcional muestra icono  
getSupportActionBar().setIcon(R.mipmap.ic_launcher); //opcional muestra icono  
//ver informacion CAMBIAR ICONO DE LA APLICACIÓN
```

//LOGICA VS GRAFICA

```
btn1 = (Button) findViewById(R.id.button);  
txt1 = (EditText) findViewById(R.id.editTextTextPersonName);  
txt2 = (EditText) findViewById(R.id.editTextTextPassword);
```

```
//programamos el boton de LOGIN, controlamos credenciales y mostramos Mensajes  
informativos (antes de las 2 llaves )}}
```

```
btn1.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
  
        if((txt1.getText().toString().equals("user")) && (txt2.getText().toString().equals("123")))  
        {  
            Intent pantallamenu = new Intent(MainActivity.this, MainActivity2.class);  
            startActivity(pantallamenu);  
            finish();  
        }  
        else  
        {  
            AlertDialog.Builder alerta = new AlertDialog.Builder(MainActivity.this);  
            alerta.setTitle("ALERTA");  
            alerta.setMessage("Credenciales incorrectas")  
                .setIcon(android.R.drawable.ic_dialog_alert);  
            alerta.show();  
  
            txt1.setText("");  
            txt2.setText("");  
            txt1.requestFocus();  
        }  
    }  
});
```

//programamos el boton de ATRÁS ◀ (barra de navegación).

1. Click derecho en la zona de Código, **Generate-Override Methods** el metodo de nombre **onBackPressed()**
2. Una vez creado el método para el evento programamos

```
@Override
public void onBackPressed() {
    //super.onBackPressed(); esta línea se deshabilita
    //ponemos el código para cerrar
```

//codigo1

```
MainActivity.super.onDestroy();
android.os.Process.killProcess(android.os.Process.myPid());
```

//alternativa

```
System.Exit(0)
```

//SUPER CIERRE---CUANDO HAY MUCHOS LAYOUTS

```
Intent salir = new Intent(Intent.ACTION_MAIN);
salir.addCategory(Intent.CATEGORY_HOME);
salir.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
startActivity(salir);
System.exit(0);
```

//o usar el AlertDialog.Builder para confirmar el cierre

```
AlertDialog.Builder alerta = new AlertDialog.Builder(MainActivity.this);
    alerta.setTitle("Aviso");
    alerta.setMessage("Desea Cerrar la Aplicación");
    alerta.setIcon(android.R.drawable.ic_dialog_alert);
    alerta.setCancelable(false);
    alerta.setPositiveButton("Si", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            System.exit(0);
        }
    });

    alerta.setNegativeButton("No", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {

        }
    });

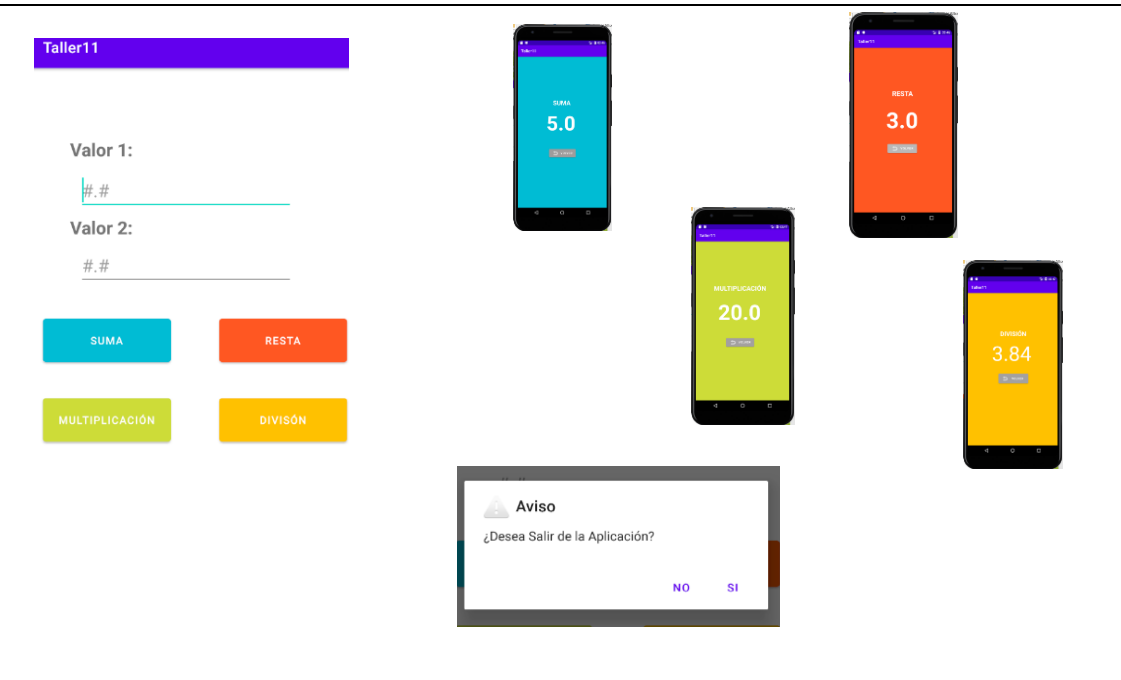
    alerta.show();
}
```

Taller 13: Uso de Varios Screen (Activity, Layout, pantallas)

REQUERIMIENTOS

- ✓ Crear un Proyecto “Empty Activity” (Lollipop 5.0)
- ✓ Insertar los siguientes componentes **MainActivity**:
 - 2 TextView (labels)
 - 2 plainText (edit tex)
 - 4 Buttons
- ✓ Crear **MainActivity2**, **MainActivity3**, **MainActivity4**, **MainActivity5**:
 - 2 TextView (labels)
 - 1 Buttons
- ✓ Implementar un programa que permita realizar las operaciones ; en pantallas diferentes (Activity)
- ✓ Explicar y conocer el manejo de Activity (ida/regreso)
- ✓ Hacer énfasis en el manejo de eventos (botón ATRAS).
- ✓ Pasos de valores entre Activitys.
- ✓ Usar avisos o mensajes de alerta para controlar errores.
- ✓ Agregar un AVD (Virtual Device) para correr la app móvil (optimo)
Compilar y Revisar Errores.

PANTALLAS



PROCEDIMIENTO

****// Pantalla Principal MainActivity
Poner los componentes requeridos

****//Crear los demás Activity
File – New – Activity – Empty Activity

Poner los componentes Necesario (2 Textview, 1 Button, Color de Background)

*****// Crear variables propias en la **MainActivity**

```
Button btn1,btn2,btn3,btn4;  
TextView lbl1,lbl2;  
EditText txt1,txt2;
```

*****// Asociar variables a los componentes **MainActivity**

@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);
```

```
    getSupportActionBar.setTitle("MENU PRINCIPAL")
```

```
    btn1 = (Button) findViewById(R.id.button);  
    btn2 = (Button) findViewById(R.id.button2);  
    btn3 = (Button) findViewById(R.id.button3);  
    btn4 = (Button) findViewById(R.id.button4);
```

```
    lbl1 = (TextView) findViewById(R.id.textView5);  
    lbl2 = (TextView) findViewById(R.id.textView6);
```

```
    txt1 = (EditText) findViewById(R.id.editTextTextPersonName);  
    txt2 = (EditText) findViewById(R.id.editTextTextPersonName2);
```

```
    txt1.setText("");  
    txt2.setText("");
```

//Programar los botones según la operación para que llame al Layout correspondiente
Y pasarle los valores del MainActivity

//Boton1 SUMA al Layout **MainActivity2**

```
btn1.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
  
        if( txt1.getText().toString().isEmpty() || txt2.getText().toString().isEmpty())  
        {  
            Toast.makeText( getApplicationContext() ,"Existen Campos Vacio", Toast.LENGTH_SHORT).show();  
        }  
        else  
        {
```

```
            //Llamado a MainActivity2
```

```
            Intent pantallasuma = new Intent(MainActivity.this, MainActivity2.class);
```

```
            //Paso de valores al MainActivity2
```

```
            pantallasuma.putExtra("prm1", txt1.getText().toString());  
            pantallasuma.putExtra("prm2", txt2.getText().toString());
```

```

        //Inciar MainActivity2
        startActivity(pantallasuma);

        //Cerrar MainActivity Origen
        finish();
    }
}
});

//Boton2 RESTA al Layout MainActivity3
btn2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        if( txt1.getText().toString().isEmpty() || txt2.getText().toString().isEmpty())
        {
            Toast.makeText( getApplicationContext() ,"Existen Campos Vacio",
            Toast.LENGTH_SHORT).show();
        }
        else
        {

            Intent pantallaresta = new Intent(MainActivity.this, MainActivity3.class);
            pantallaresta.putExtra("prm1", txt1.getText().toString());
            pantallaresta.putExtra("prm2", txt2.getText().toString());
            startActivity(pantallaresta);
            finish();
        }
    }
});

```

```

//Boton3 MULTIPLICAR al Layout MainActivity4
btn3.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        if( txt1.getText().toString().isEmpty() || txt2.getText().toString().isEmpty())
        {
            Toast.makeText( getApplicationContext() ,"Existen Campos Vacio",
            Toast.LENGTH_SHORT).show();
        }
        else
        {

            Intent pantallamulti = new Intent(MainActivity.this, MainActivity4.class);
            pantallamulti.putExtra("prm1", txt1.getText().toString());
            pantallamulti.putExtra("prm2", txt2.getText().toString());
            startActivity(pantallamulti);
            finish();
        }
    }
});

```

```
});
```

```
//Boton4 DIVIDIR al Layout MainActivity5
```

```
btn4.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
  
        if( txt1.getText().toString().isEmpty() || txt2.getText().toString().isEmpty() ||  
Float.parseFloat(txt2.getText().toString()) == 0.0)  
        {  
            Toast.makeText( getApplicationContext(), "Comprobar: Campos Vacios/Division para cero",  
Toast.LENGTH_SHORT).show();  
        }  
        else  
        {  
            Intent pantalladv = new Intent(MainActivity.this, MainActivity5.class);  
            pantalladv.putExtra("prm1", txt1.getText().toString());  
            pantalladv.putExtra("prm2", txt2.getText().toString());  
  
            startActivity(pantalladv);  
            finish();  
        }  
    }  
});
```

```
*****  
MainActivity2 - "Suma"  
MainActivity3 - "Resta"  
MainActivity4 - "Multiplicación"  
MainActivity5 - "División"
```

```
*****  
****Crear variables propias
```

```
Button btn1;  
TextView ressuma;
```

```
****Asociar Variables a Componentes
```

```
btn1 = (Button) findViewById(R.id.button5);  
ressuma = (TextView) findViewById(R.id.textView7);
```

```
****Captura valores enviados desde la Activity Principal y realizar la operación
```

```
String num1 = getIntent().getStringExtra("prm1");  
String num2 = getIntent().getStringExtra("prm2");  
@Override
```

```
Evento OnCreate
```

```
float res = Float.parseFloat(num1) + Float.parseFloat(num2);  
ressuma.setText(res + "");
```

REGRESAR AL LAYOUT ANTERIOR – METODO 1

*****Programar el botón VOLVER de cada Layout

```
btn1.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
  
        Intent pantallamenu = new Intent(MainActivity2.this, MainActivity.class);  
        startActivity(pantallamenu);  
        finish();  
  
    }  
});
```

REGRESAR AL LAYOUT ANTERIOR – METODO 2

*****Programar el botón atrás ◀ barra de navegación

- 1.Click derecho en la zona de Código(parte final antes de las }}, Generate-Override Methods el metodo de nombre onBackPressed()
- 2.Una vez creado el método para el evento programamos

```
@Override  
public void onBackPressed() {  
    Intent pantallamenu = new Intent(MainActivityX.this, MainActivityY.class);  
    startActivity(pantallamenu);  
    finish();  
}
```

REGRESAR AL LAYOUT ANTERIOR – METODO 3

***** icono atrás en el <- ActionBar

OJO: QUITAR finish(); del código Intent.

En el archivo AndroidManifest.xml, dirigirse al activity en la que quieres programar el regreso(<-)
Poner el parámetro `android:parentActivityName=".actividad a donde ir"`

Ejemplo 2 al menu

```
<activity android:name=".MainActivity2"  
android:parentActivityName=".MainActivity"></activity>
```

Ejemplo 3 al menu

```
<activity android:name=".MainActivity3"  
android:parentActivityName=".MainActivity"></activity>
```

//SUPER CIERRE---CUANDO HAY MUCHOS LAYOUTS

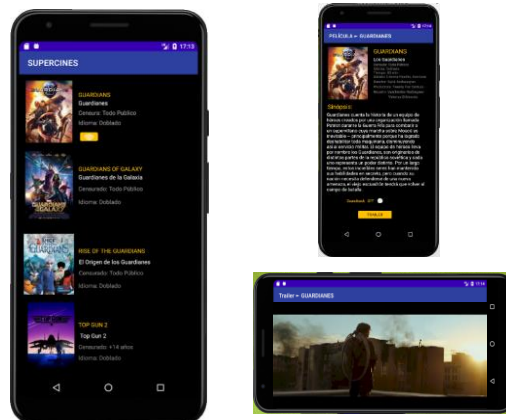
```
Intent salir = new Intent(Intent.ACTION_MAIN);  
salir.addCategory(Intent.CATEGORY_HOME);  
salir.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);  
startActivity(salir);  
System.exit(0);
```

Taller 14: App Multimedia

REQUERIMIENTOS

- ✓ Crear un Proyecto "Empty Activity" (Lollipop 5.0)
- ✓ Crear 3 Activity(Layout)
- ✓ Insertar los siguientes componentes **MainActivity**:
 - 1 ScrollView (Containers)
 - ImageView
 - 4 TextView (labels)
 - 1Button(por cada película)
- ✓ **MainActivity2** :
 - ImageView
 - 4 TextView (labels)
 - 1 Switch(commons)
 - 1 Button
- ✓ **MainActivity3 : (generar el layout landscape)**
 - VideoView
- ✓ **Implementar un aplicación que permite interactuar con contenido multimedia (imagen, texto, audio, video)**
- ✓ **Hacer énfasis en el manejo de eventos (botón ATRAS).**
- ✓ **Usar avisos o mensajes de alerta para controlar errores.**
- ✓ Agregar un AVD (Virtual Device) para correr la app móvil (optimo)
- ✓ Compilar y Revisar Errores.

Pantallas



Procedimiento

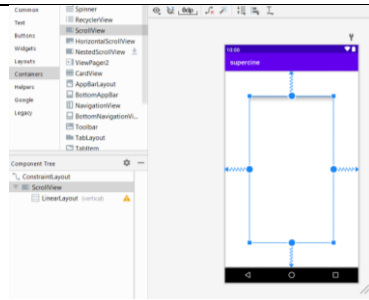
```
//cambiar del titulo del action bar  
getSupportActionBar().setTitle("Acceso");
```

Crear MainActivity 2, MainActivity 3

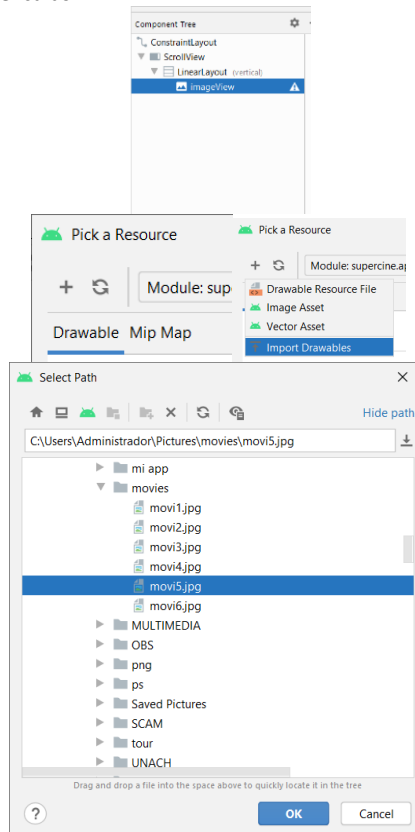
Files-New-Activity-Empty Activity

Sobre el MainActivity1

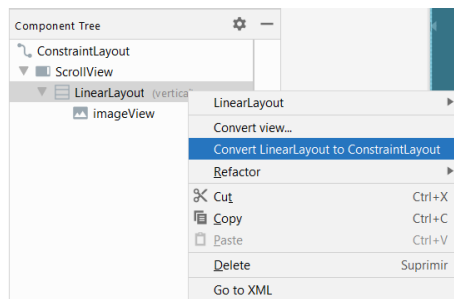
Agregar el componente **ScrollView** (ocupar toda la pantalla)

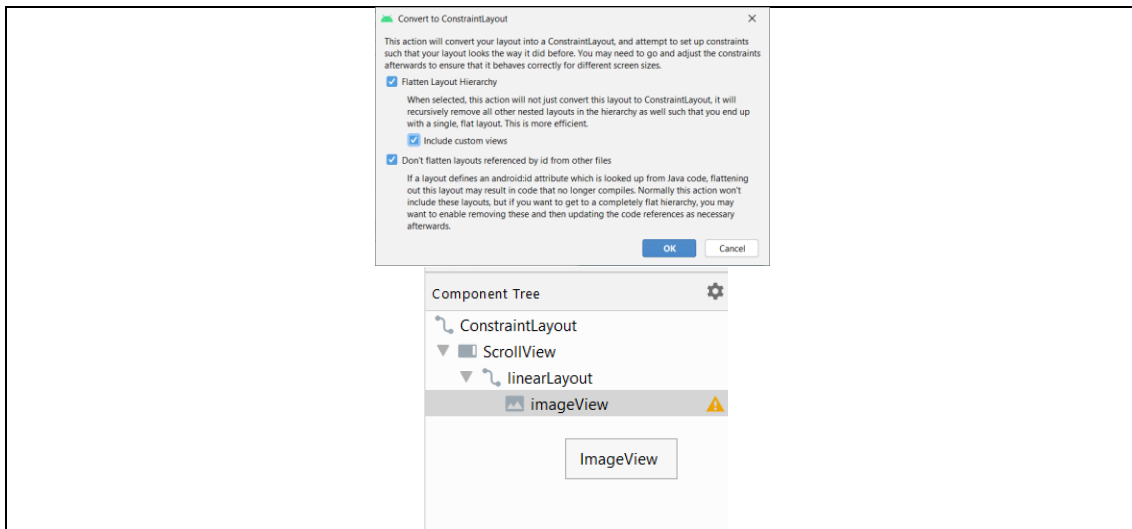


Agregar ImageView dentro del ScrollView/LinearLayout en el “Árbol de Componentes” y agregar las imágenes personalizadas de 3 portadas de películas:

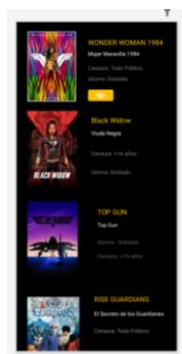
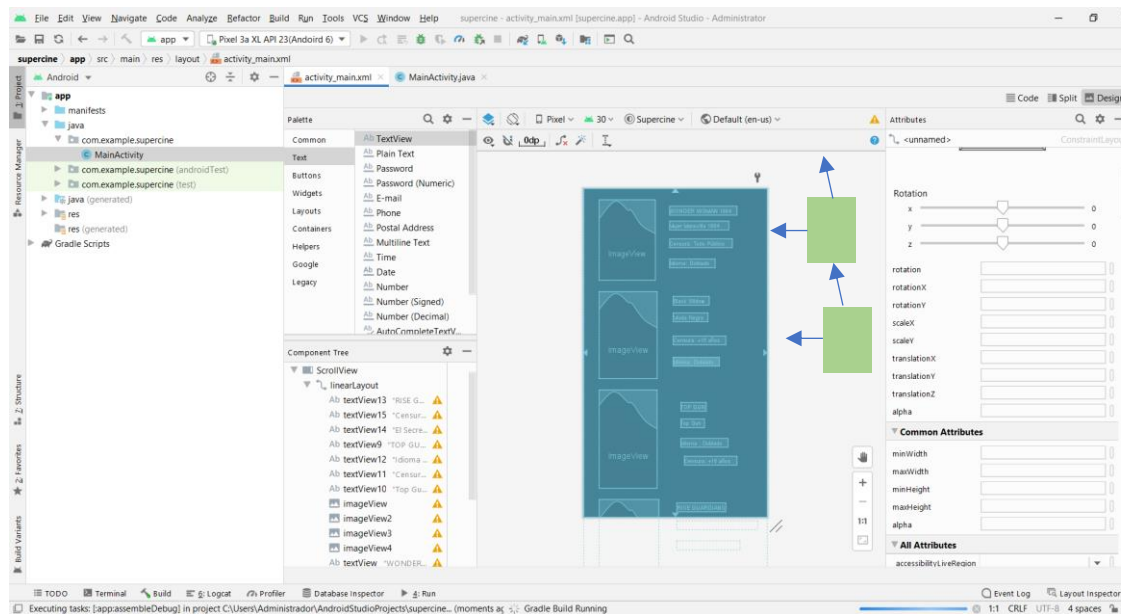


Para poder Ordenar lo componentes dentro del ScrollView/LinearLayout convertirlo a ConstrainLayout; click derecho en linearLayout:





Agregar el resto de Componentes necesarios hasta obtener el diseño propuesto (se recomienda trabajar en BLUEPRINT para acomodar los componentes)



Programamos el Boton (Button1) correspondiente a la primera película (Recuerde Crear el Activity2)

//Declaro las variables propias usar:

```
Button btn1;
```

//Asigno variables propias a los componentes

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    btn1= (Button) findViewById(R.id.button);

}
}
```

//Creamos el evento OnClickListener y enlazamos al MainActivity2

```
btn1= (Button) findViewById(R.id.button);

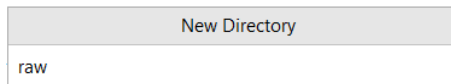
btn1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        Intent peli1 = new Intent(MainActivity.this, MainActivity2.class);
        startActivity(peli1);
        finish();

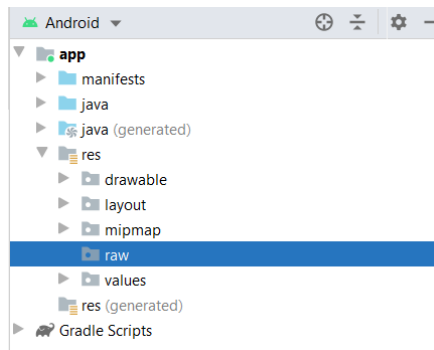
    }
});
```

//Agregamos Sonido al click del botón usando la librería SoundPool

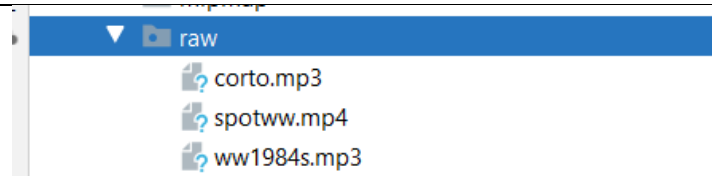
Creamos la carpeta raw (manualmente), en el arbol de proyecto, click derecho sobre el directorio **res** ; new-Directory y ponemos de nombre **raw**



El árbol se verá así:



Sobre esta nueva carpeta(directorio) copiamos (ctr+c, ctr+v) todos los elementos multimedia (sonidos, audios, videos- mp3,mp4); los nombres de archivo deben estar en minúsculas sin espacios.



//Sonido en el clic del botón, librería SoundPool (para sonidos cortos)

```
//Creamos las variables propias
SoundPool sp1;
int id_sonido;
```

//en el evento créate MainActivity creamos el objeto

```
sp1 = new SoundPool(1, AudioManager.STREAM_MUSIC,1);
id_sonido=sp1.load(this, R.raw.corto, 1);
```

//en el evento OnClikListener del button1 reproducimos el sonido

```
sp1.play(id_sonido,1,1, 1,0,1);
```

Sobre el MainActivity2

Agregar los componentes necesarios

✓ **MainActivity2 :**

- ImageView
- 4 TextView (labels)
- 1 Switch(commons)
- 1 Button



//Poner soundtrack en el botón tipo switch ()

//declaro variable propia para el switch y para el componente MediaPlayer

```
Switch sw1;
MediaPlayer mp;
```

//asigno variables propias a los compOnjnetes del sistema

```
sw1= (Switch)findViewById(R.id.switch1);
```

```
mp = MediaPlayer.create(this,R.raw.guardians);
getSupportActionBar().setTitle(" PELÍCULA ► WONDER WOMAN 1984");
```

//en el evento click del switch valido y activo el sonido (no olvidar copiar el sonido en la carpeta raw)

```
sw1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (sw1.isChecked())
        {
            sw1.setText("Soundtrack ON");
            mp.start();
        }
    }
});
```

```

    }
    else
    {
        sw1.setText("Soundtrack OFF");
        mp.pause();
    }
}

});

```

//creo la función del onBackPressed para el botón de retroceder al layout anterior y pongo detener sonido(stop)

Click derecho, en el código, GENERATE-METHODS OVERRIDE – onBackPressed

```

@Override
public void onBackPressed() {

    //super.onBackPressed();
    mp.stop();

    Intent pantallamenu = new Intent(MainActivity2.this, MainActivity.class);
    startActivity(pantallamenu);
    finish();
}

```

//programar el boton Trailer (llama al Activity3, no olvidar crearla) para mostrar video Trailer

// declaro variable propia

```
Button btn2;
```

//Asigno variable al componente

```
Btn2 = (Button) findViewById(R.id.button2);
```

//Programo el evento onClickListener

```

btn2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        Intent trailer1 = new Intent(MainActivity2.this, MainActivity3.class);
        startActivity(trailer1);
        finish();
    }
});

```

Sobre el MainActivity3 (no olvidar copiar el video mp4 al directorio raw)

//Agrego el componente VideoView , que ocupe toda el Layout

//Crear Variable propia del Componente VideoView

```
VideoView videotrailer;
```

//En el Evento onCreate() asigno la variable al componte, asigno path de video y agrego el controller

```
getSupportActionBar().setTitle(" Trailer ► WONDER WOMAN 1984");
```

```

videotrailer = (VideoView) findViewById(R.id.videoView);

videotrailer.setVideoURI(Uri.parse("android.resource://" + getPackageName() + "/" + R.raw.spotwm));
MediaController mc = new MediaController(this);
videotrailer.setMediaController(mc);
videotrailer.start();

```

//programo el evento OnBackPressed

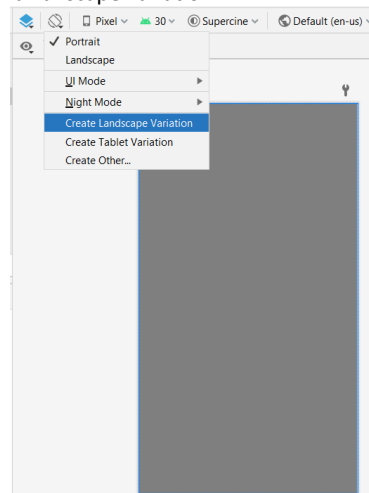
```

@Override
public void onBackPressed() {
    //super.onBackPressed();
    Intent peli1 = new Intent(MainActivity3.this, MainActivity2.class);
    startActivity(peli1);
    finish();
}

```

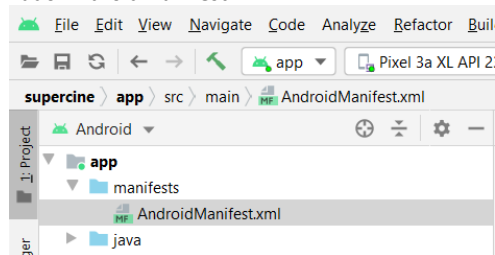
HABILITAR MODO LANDSCAPE ACTIVITY3

Estando en la Activity escogida , crear la Landscape variation



Notaremos que ahora tenemos 2 layouts idénticos pero de diferente orientación

Nos dirigimos al archivo xml llamado AndroidManifest.xml



Buscamos el **Activity** que queremos mostrar en modo Landscape (horizontal) y agregamos un parámetro a la **etiqueta** correspondiente.

```

<activity android:name="MainActivity3" android:screenOrientation="landscape" ></activity>

```

Nota: si quiere ocupar todo el ancho y alto del VideoView poner el las propiedades del componente:

```

Layout_Height match_parent
Laayout_Width match_parent

```

layout_height	match_parent	▼	0
▶ layout_margin	[?, ?, ?, ?]		
layout_width	match_parent	▼	0
scrollbarDefaultDelayB...	400		0