

# PRACTICA 26 INTRODUCCION A FRAGMENTS

Los fragments son una herramienta especial de Android studio para desplegar objetos reutilizables con funciones de una actividad pero sin serlo, ya que los fragments deben de tener un actividad que los soporte.

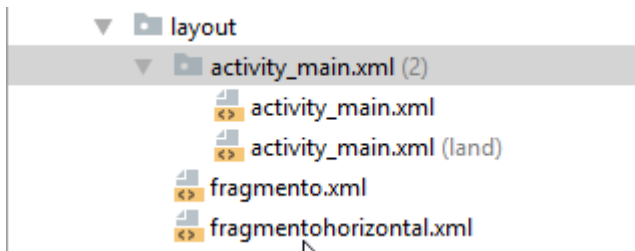
En esta practica, también se abordara el hecho de tener diseños para una orientación PORTRAIT y otra LANDSCAPE, como se ilustra:



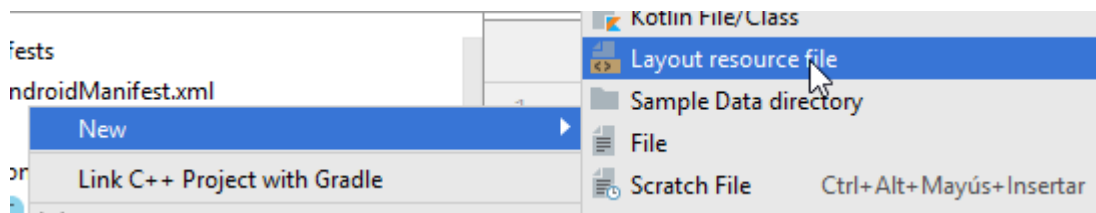
## Crear un proyecto con una empty activity.

Dentro de la carpeta layout debemos de crear dos fragments:

- fragmento
- fragmentohorizontal

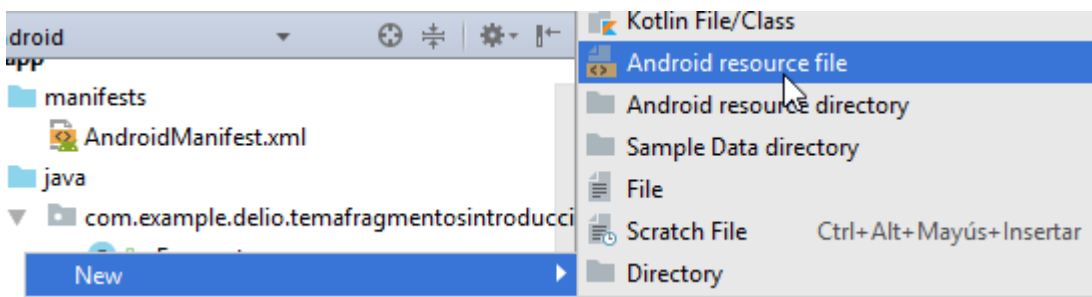


Boton derecho en la carpeta layout, new, Layout resource file.

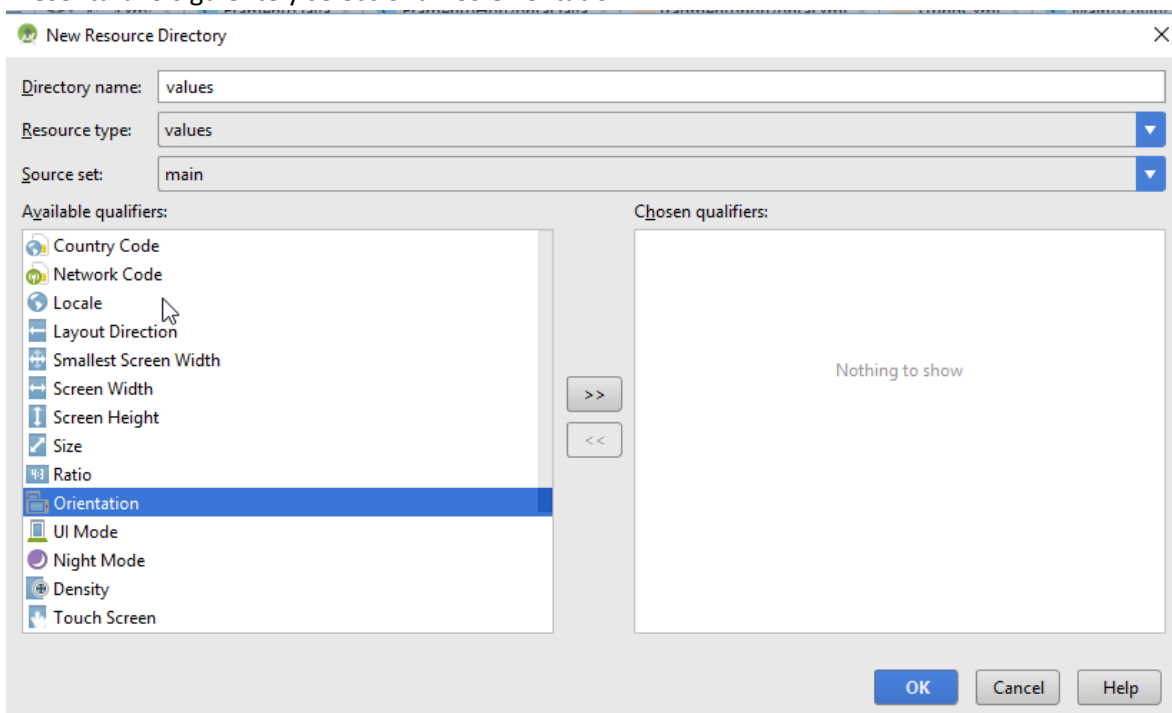


Ahora vamos a crear la carpeta para los recursos en LANDSCAPE.

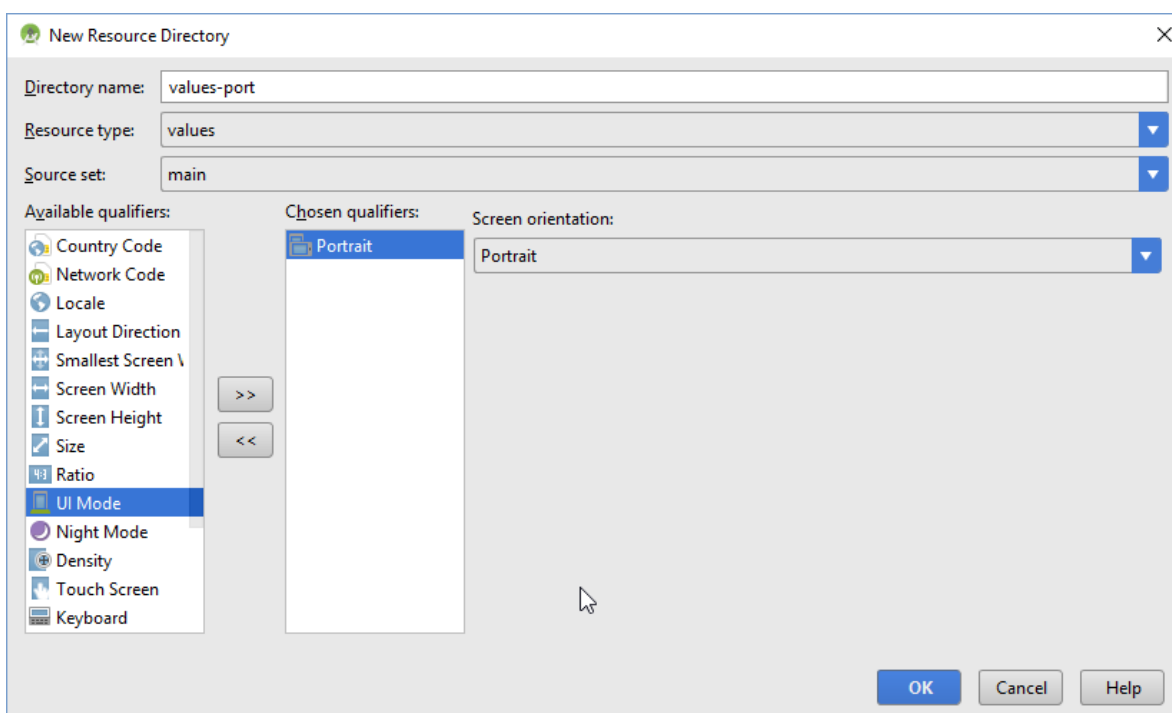
En la carpeta res botón derecho y creamos un archivo del tipo Android resource file



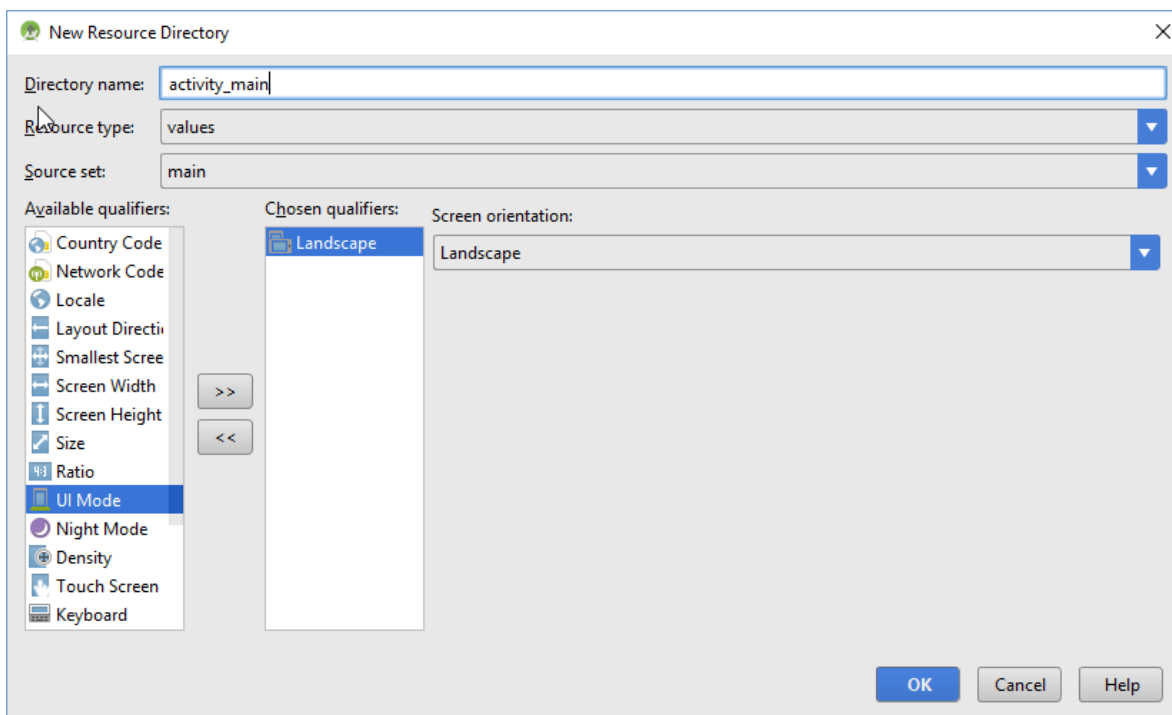
Presentara lo siguiente y seleccionamos Orientation



Lo agregamos a la ventana derecha.



## Cambiamos de PORTRAIT A LANDSCAPE



Escribimos en Directory name `activity_main`, deben ser de nombre iguales (Android internamente lo pone en otra carpeta) y aceptamos (Boton OK).

Debe de quedar algo por el estilo.

```
▼ activity_main.xml (2)
  <> activity_main.xml
  <> activity_main.xml (land)
```

## PRIMER FRAGMENT

Nuestro primer fragment de nombre fragmento tiene el siguiente código XML:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ImageView
        android:id="@+id/imgImagen"
        android:layout_margin="20dp"
        android:src="@drawable/srbern"
        android:scaleType="fitCenter"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Sr Bern (gira para ver su Biografia)"
        android:textAlignment="center"
        android:textSize="35sp" />
</LinearLayout>
```

Y el código de java:

```
public class Framento extends Fragment {
    private View view;
    @Nullable
    @Override
    public View onCreateView(LayoutInflater inflater, @Nullable ViewGroup
        container, Bundle savedInstanceState) {

        view = inflater.inflate(R.layout.fragmento, container);
        return view;
    }
}
```

## SEGUNDO FRAGMENT

El segundo fragment tiene el siguiente código XML:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="horizontal"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ImageView
        android:id="@+id/imgImagen"
        android:layout_margin="20dp"
        android:src="@drawable/srbern"
        android:scaleType="fitCenter"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
    <TextView
        android:id="@+id/txtSrbern"
        android:layout_marginTop="20dp"
        android:layout_marginRight="20dp"
        android:textSize="17sp"
        android:inputType="textMultiLine"
        android:text="@string/srbern"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
</LinearLayout>
```

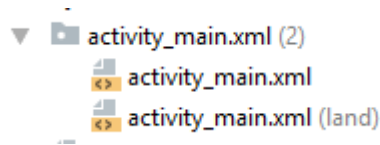
Y el código java:

```
public class FramentoHorizontal extends Fragment {
    private View view;
    @Nullable
    @Override
    public View onCreateView(LayoutInflater inflater, @Nullable ViewGroup
        container, Bundle savedInstanceState) {

        view = inflater.inflate(R.layout.fragmentohorizontal,container);
        return view;
    }
}
```

NOTA: Cada fragmento hereda de la clase Fragement y en el constructor se le indica que layout creara.

Finalmente, vamos a copiar nuestra actividad principal a la carpeta para nuestra View LANDSCAPE



Modificaremos a cada actividad según se indica:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

    tools:context="com.example.delio.temafragmentosintroduccion.MainActivity"
>
    <TextView
        android:text="PORTRAIT"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />

    <fragment
        android:id="@+id/fragment1"

        android:name="com.example.delio.temafragmentosintroduccion.Framento"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_weight="1" />
</LinearLayout>
```

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:text="LANDSCAPE"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
    <fragment
        android:id="@+id/fragment2"

        android:name="com.example.delio.temafragmentosintroduccion.FramentoHorizontal"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_weight="1" />
</LinearLayout>
```

Si analiza el código, solo cambia el llamado al fragment, de ahí todo es igual.

El archivo de lógica de la actividad principal, no se modifica.

Así terminamos la aplicación e introducimos el concepto de fragments, en este caso son del tipo estático.

La intención de utilizar fragments es para el reuso de código.