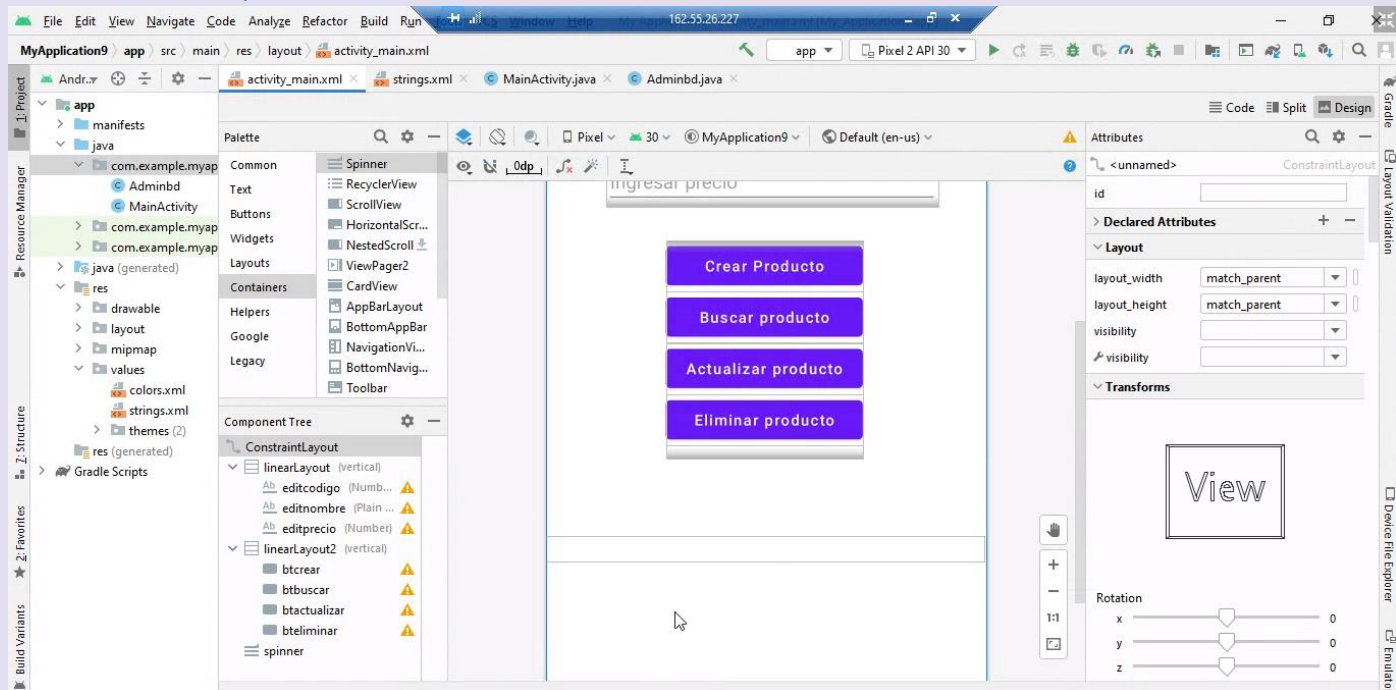


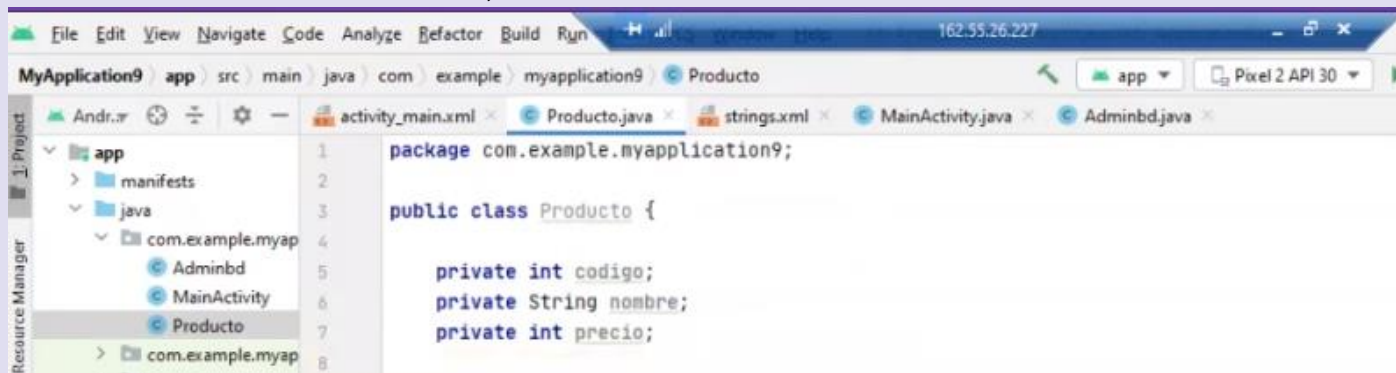
CLASE LUNES-05-JULIO-2021

INICIO DE CLASE-

Pasar valores a un spinner

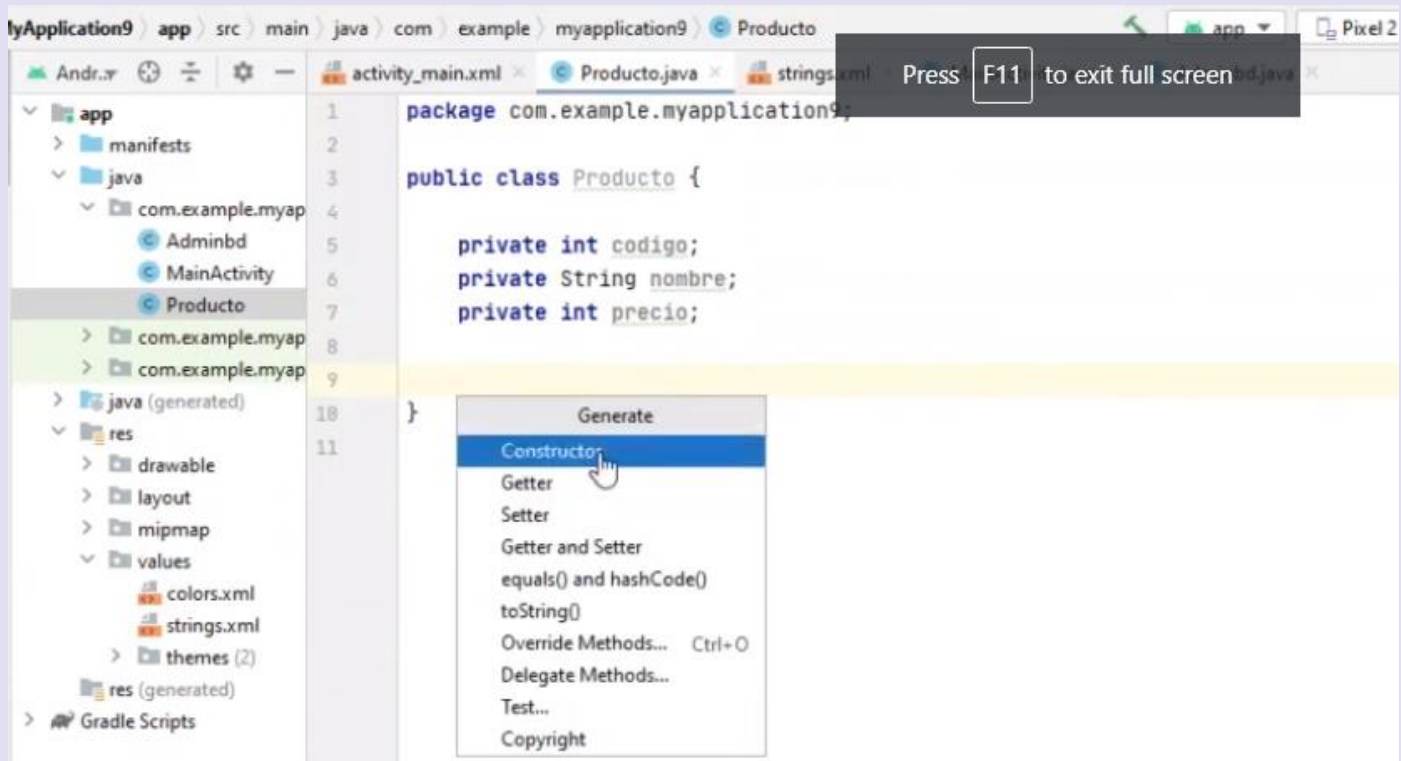


Primero crearemos una clase llamada producto, porque necesitamos trabajar con un ArrayList, entonces declaramos nuestros atributos,

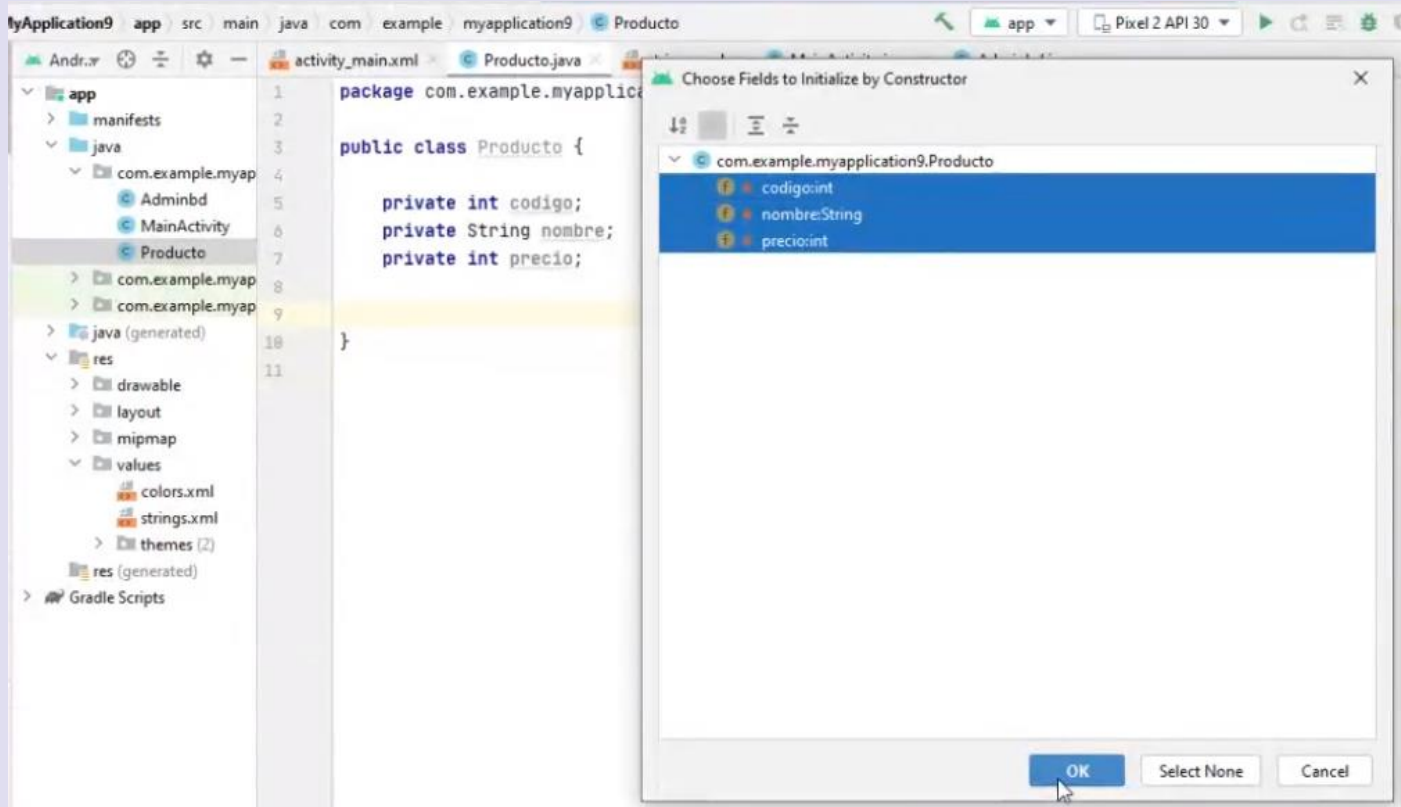


Estos son los campos que tenía nuestra tabla (código, nombre, precio).

Alt+Insert

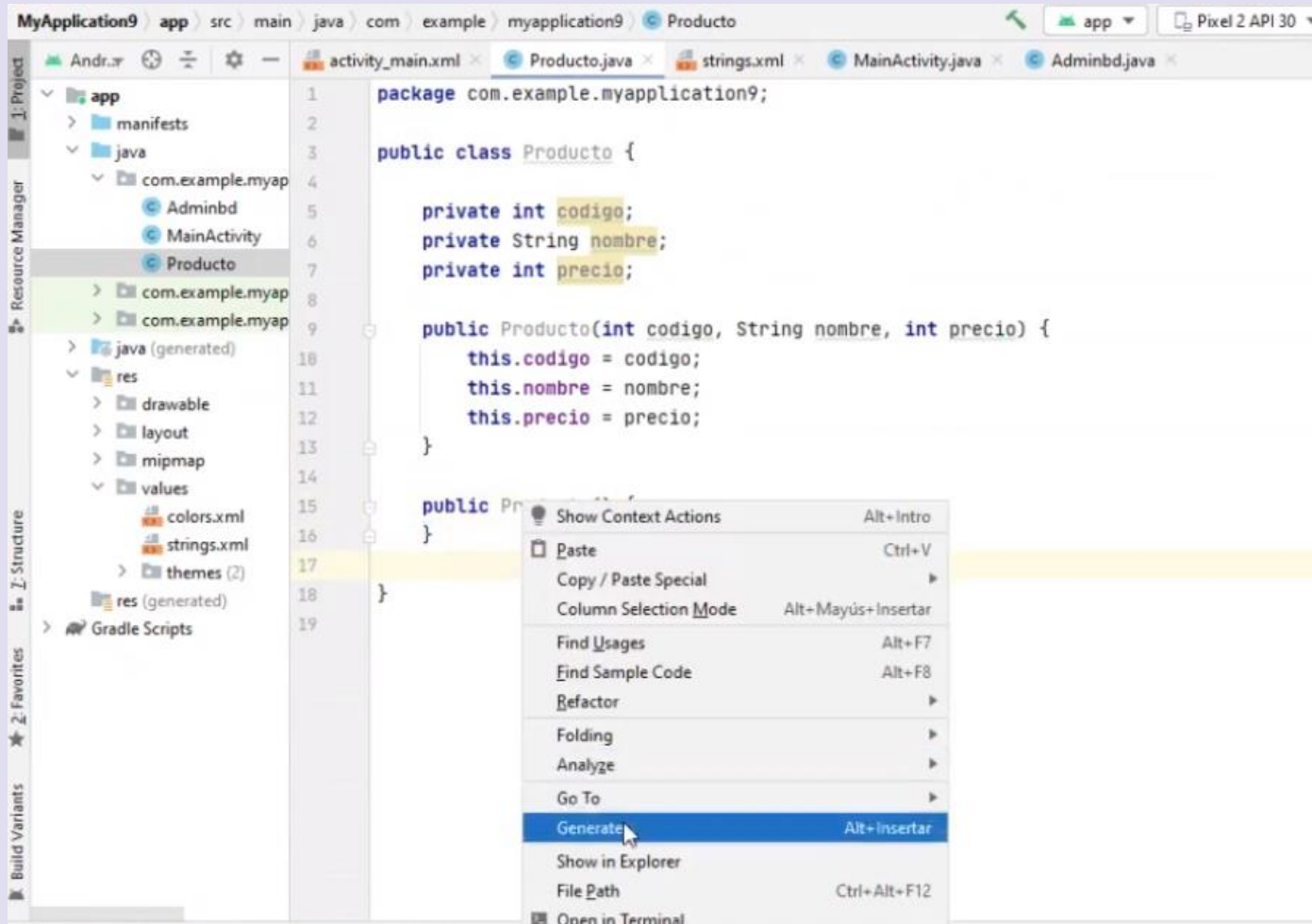


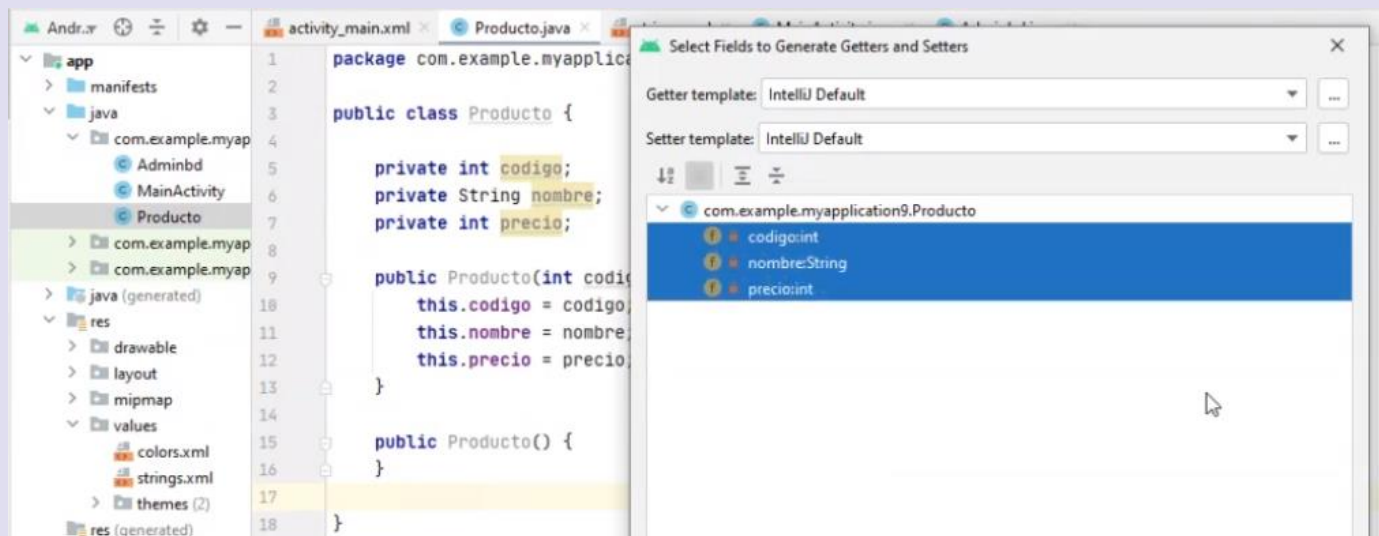
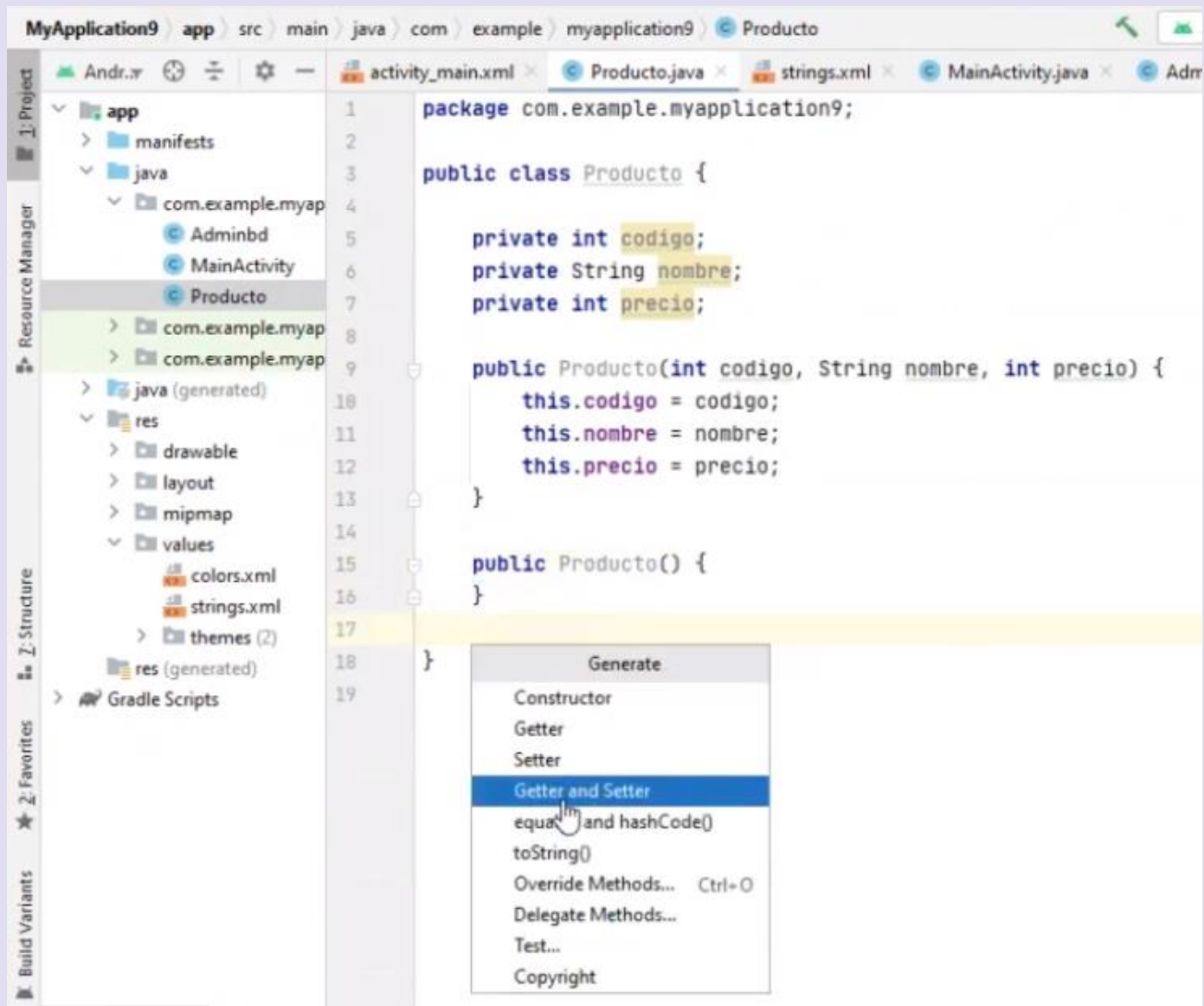
Generamos método constructor por defecto

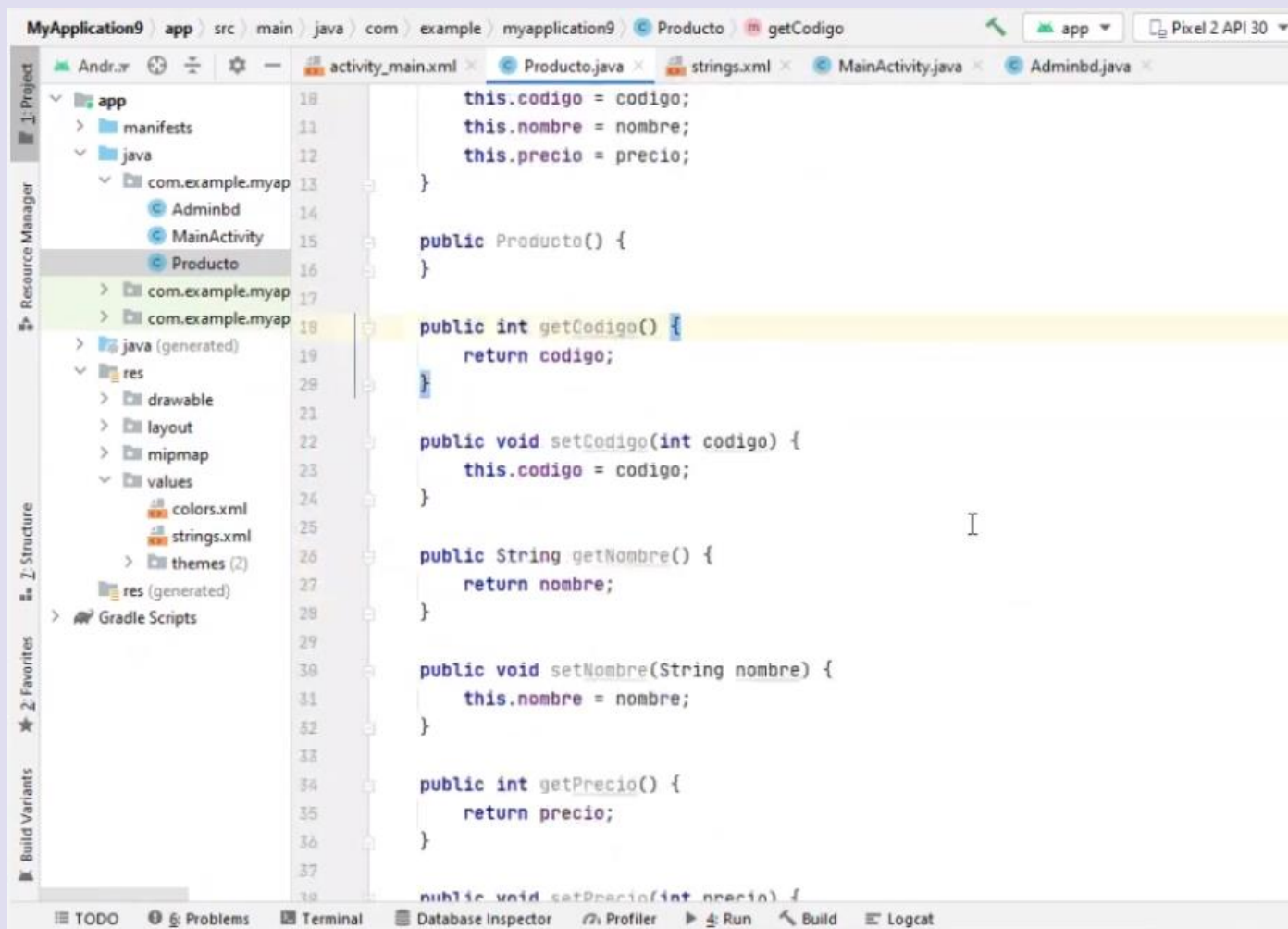




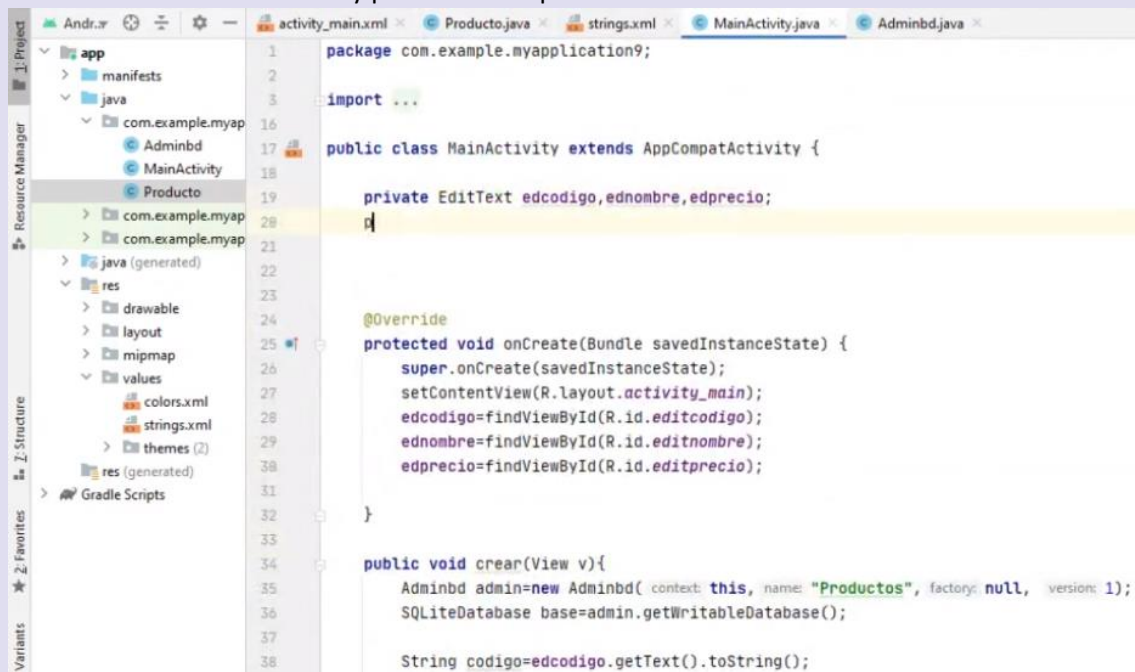
Ahora los métodos get y set



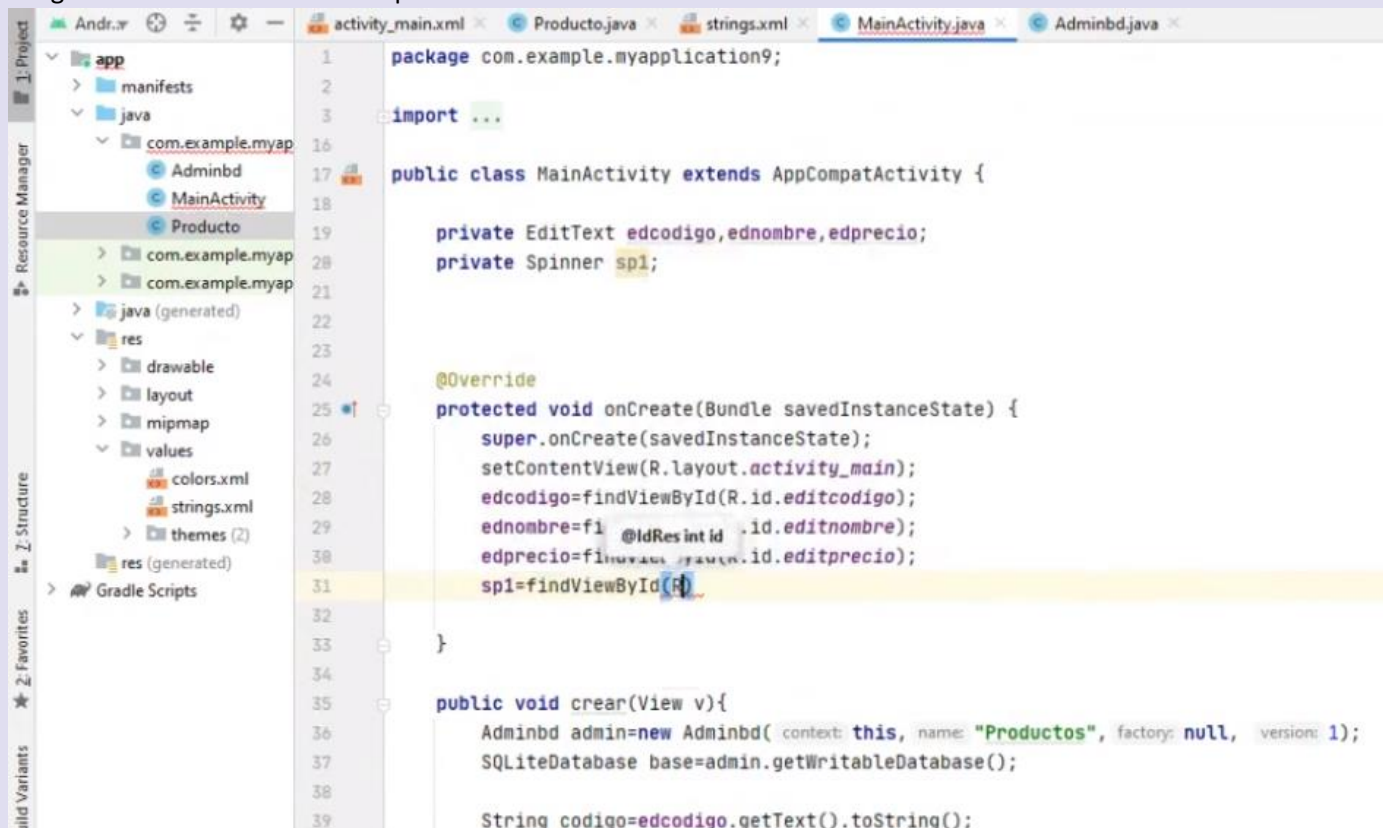




Vamos a nuestra MainActivity para crear el spinner

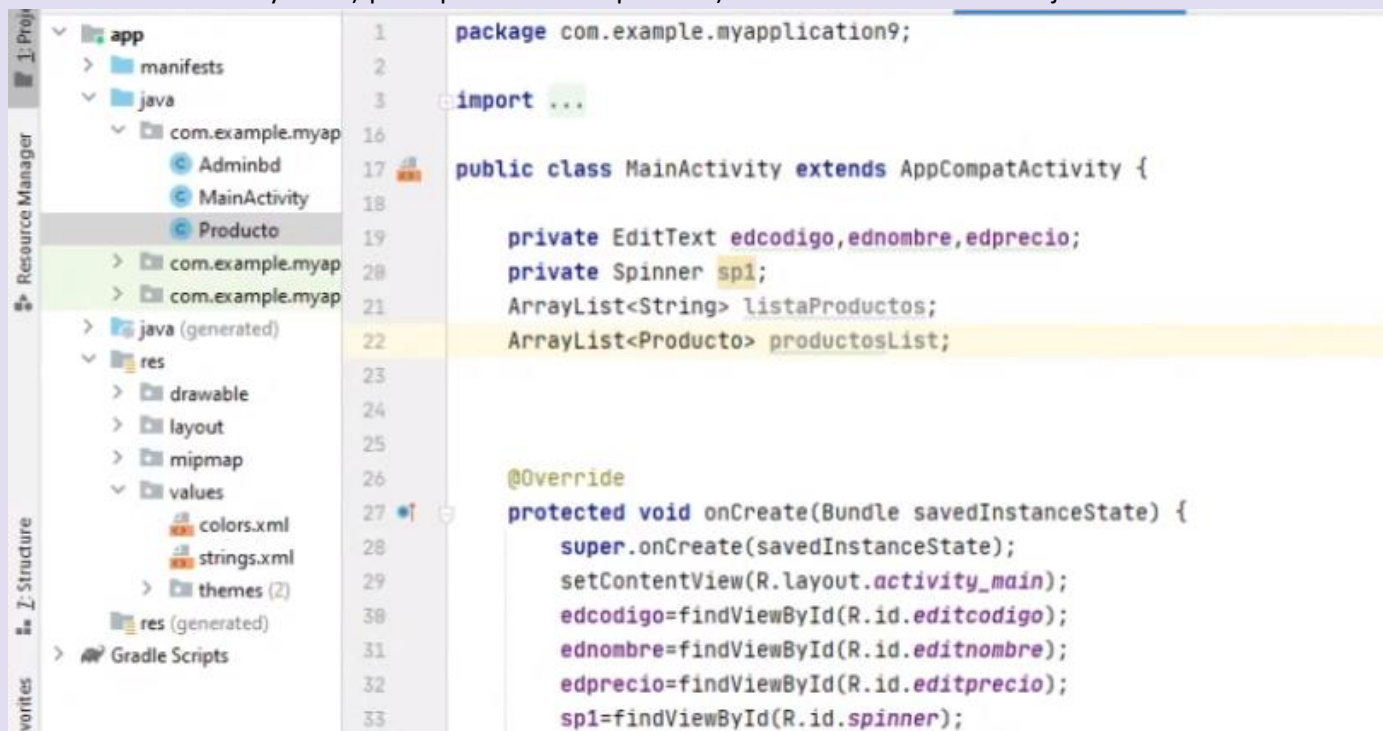


Luego hacemos la referencia al spinner



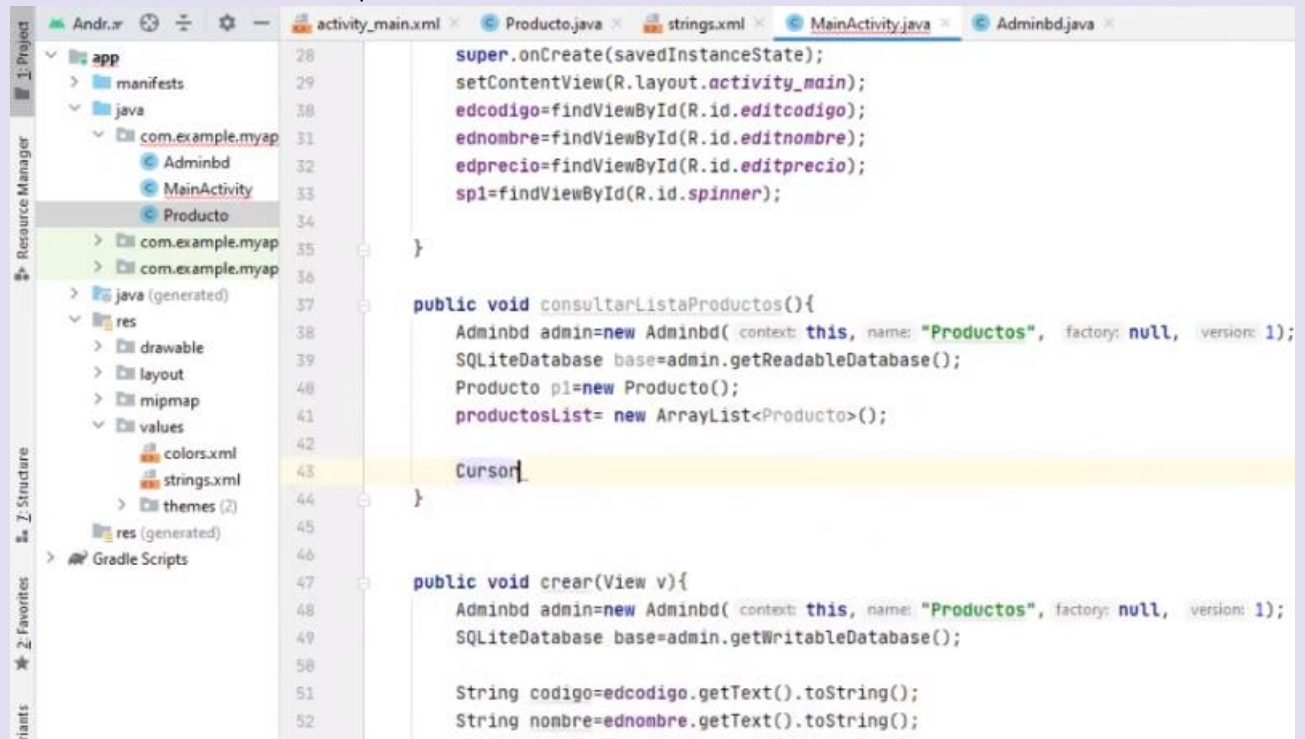
```
1 package com.example.myapplication9;
2
3 import ...
4
5 public class MainActivity extends AppCompatActivity {
6
7     private EditText edcodigo,ednombre,edprecio;
8     private Spinner sp1;
9
10    @Override
11    protected void onCreate(Bundle savedInstanceState) {
12        super.onCreate(savedInstanceState);
13        setContentView(R.layout.activity_main);
14        edcodigo=findViewById(R.id.editcodigo);
15        ednombre=findViewById(R.id.editnombre);
16        edprecio=findViewById(R.id.editprecio);
17        sp1=findViewById(R.id.spinner);
18    }
19
20    public void crear(View v){
21        Adminbd admin=new Adminbd( context: this, name: "Productos", factory: null, version: 1);
22        SQLiteDatabase base=admin.getWritableDatabase();
23
24        String codigo=edcodigo.getText().toString();
```

Ahora creamos 2 ArrayList. 1)que le pasaremos al spinner 2)en donde tendremos los objetos.



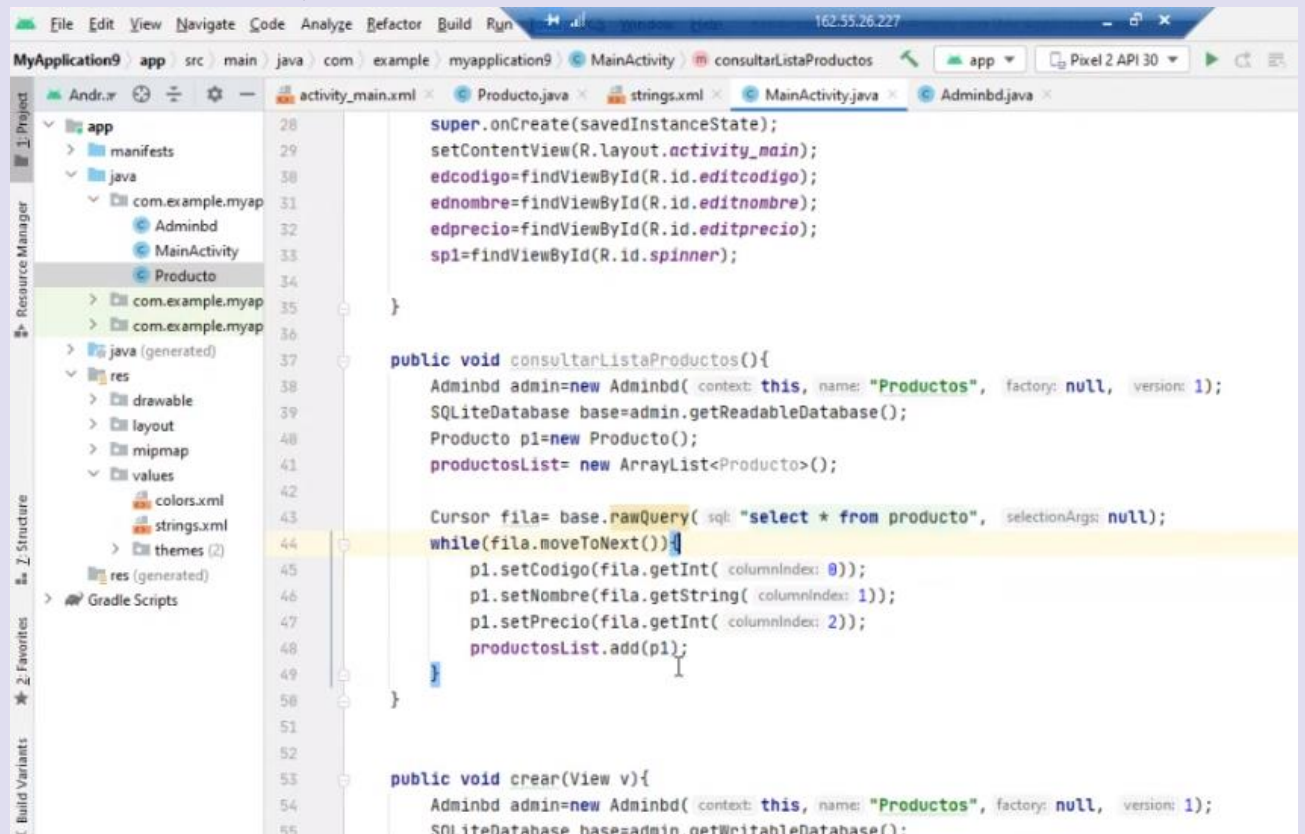
```
1 package com.example.myapplication9;
2
3 import ...
4
5 public class MainActivity extends AppCompatActivity {
6
7     private EditText edcodigo,ednombre,edprecio;
8     private Spinner sp1;
9     ArrayList<String> listaProductos;
10    ArrayList<Producto> productosList;
11
12    @Override
13    protected void onCreate(Bundle savedInstanceState) {
14        super.onCreate(savedInstanceState);
15        setContentView(R.layout.activity_main);
16        edcodigo=findViewById(R.id.editcodigo);
17        ednombre=findViewById(R.id.editnombre);
18        edprecio=findViewById(R.id.editprecio);
19        sp1=findViewById(R.id.spinner);
```

Ahora creamos los métodos que vamos a utilizar



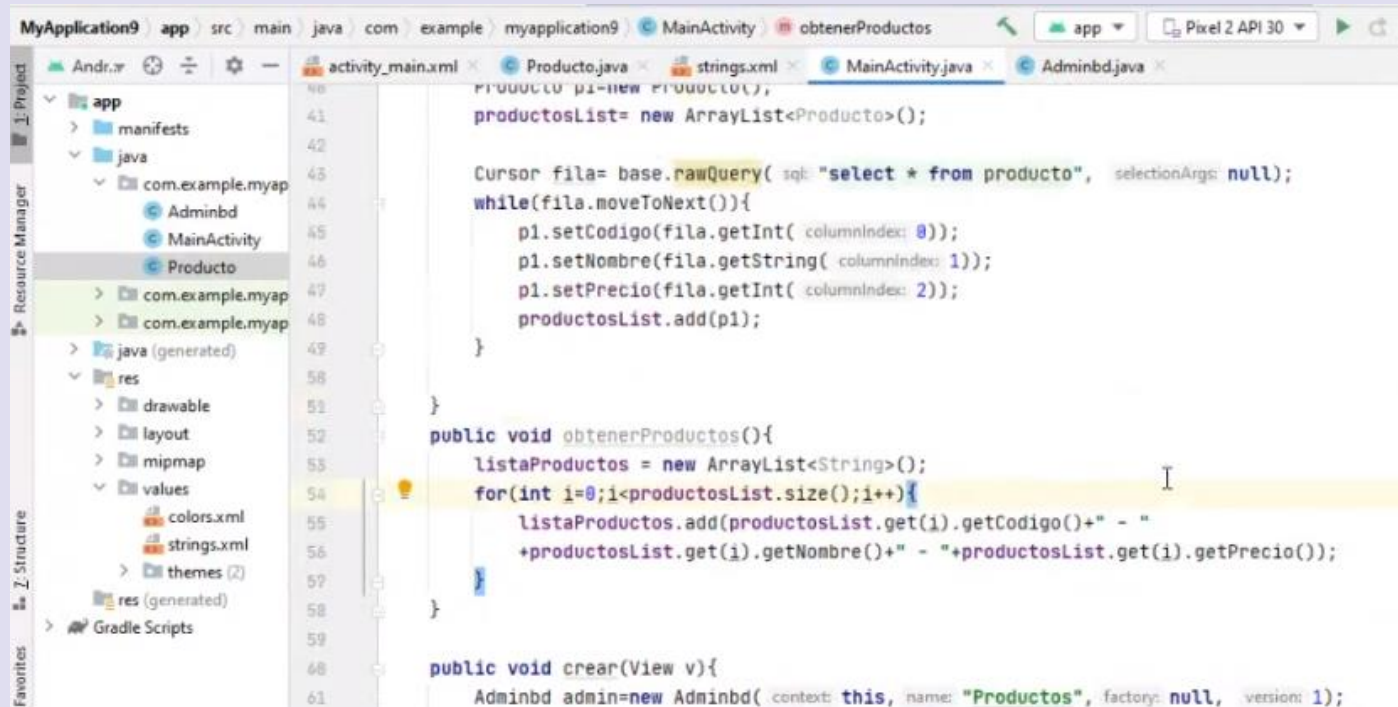
```
28 super.onCreate(savedInstanceState);
29 setContentView(R.layout.activity_main);
30 edcodigo=findViewById(R.id.editcodigo);
31 ednombre=findViewById(R.id.editnombre);
32 edprecio=findViewById(R.id.editprecio);
33 sp1=findViewById(R.id.spinner);
34
35 }
36
37 public void consultarListaProductos(){
38     Adminbd admin=new Adminbd( context: this, name: "Productos", factory: null, version: 1);
39     SQLiteDatabase base=admin.getReadableDatabase();
40     Producto p1=new Producto();
41     productosList= new ArrayList<Producto>();
42
43     Cursor
44 }
45
46
47 public void crear(View v){
48     Adminbd admin=new Adminbd( context: this, name: "Productos", factory: null, version: 1);
49     SQLiteDatabase base=admin.getWritableDatabase();
50
51     String codigo=edcodigo.getText().toString();
52     String nombre=ednombre.getText().toString();
```

Como vamos a consultar, necesitaremos un cursor

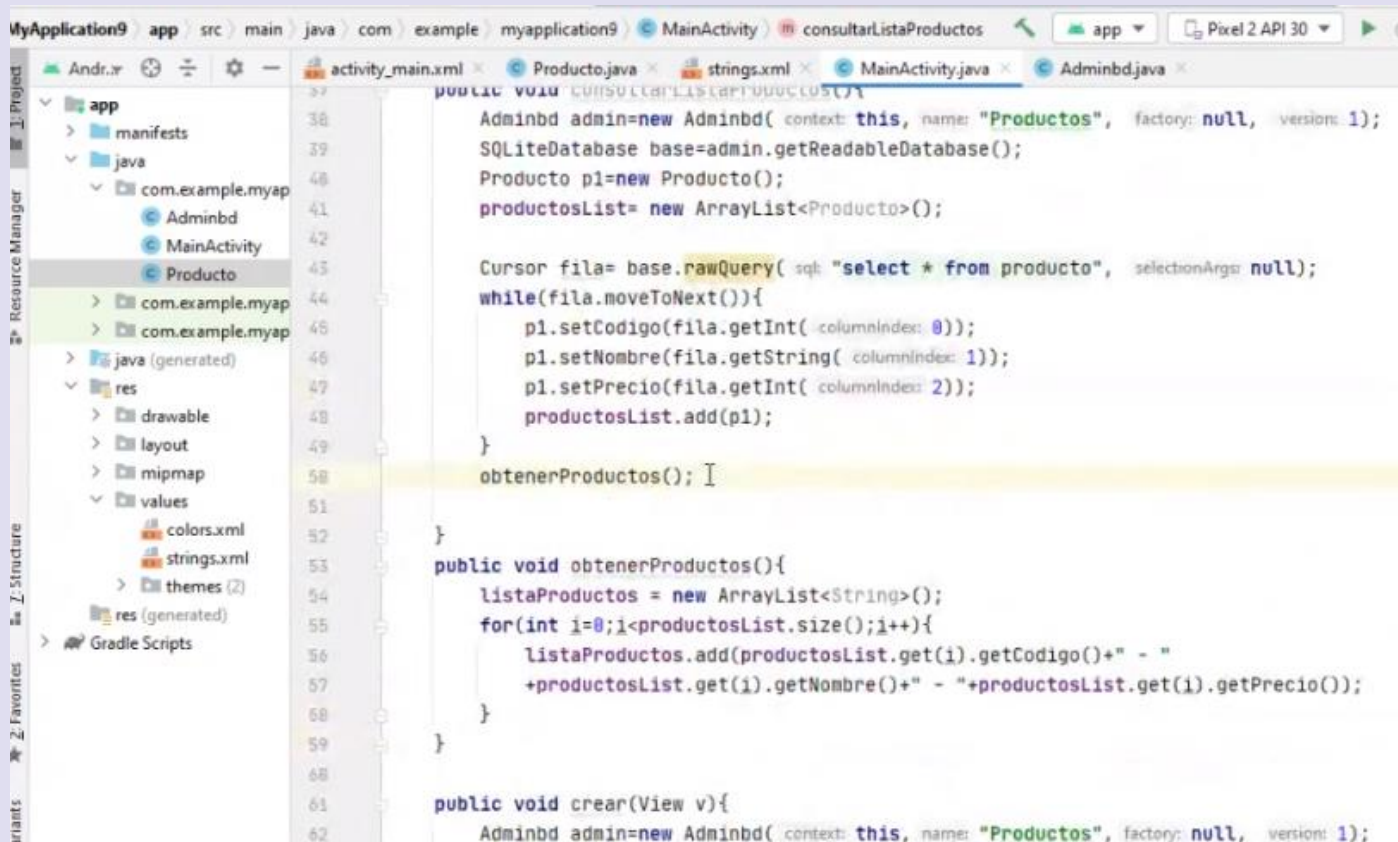


```
28 super.onCreate(savedInstanceState);
29 setContentView(R.layout.activity_main);
30 edcodigo=findViewById(R.id.editcodigo);
31 ednombre=findViewById(R.id.editnombre);
32 edprecio=findViewById(R.id.editprecio);
33 sp1=findViewById(R.id.spinner);
34
35 }
36
37 public void consultarListaProductos(){
38     Adminbd admin=new Adminbd( context: this, name: "Productos", factory: null, version: 1);
39     SQLiteDatabase base=admin.getReadableDatabase();
40     Producto p1=new Producto();
41     productosList= new ArrayList<Producto>();
42
43     Cursor fila= base.rawQuery( sql: "select * from producto", selectionArgs: null);
44     while(fila.moveToNext()){
45         p1.setCodigo(fila.getInt( columnIndex: 0));
46         p1.setNombre(fila.getString( columnIndex: 1));
47         p1.setPrecio(fila.getInt( columnIndex: 2));
48         productosList.add(p1);
49     }
50 }
51
52
53 public void crear(View v){
54     Adminbd admin=new Adminbd( context: this, name: "Productos", factory: null, version: 1);
55     SQLiteDatabase base=admin.getWritableDatabase();
```


Ahora necesitamos otro método que nos permita obtener los productos



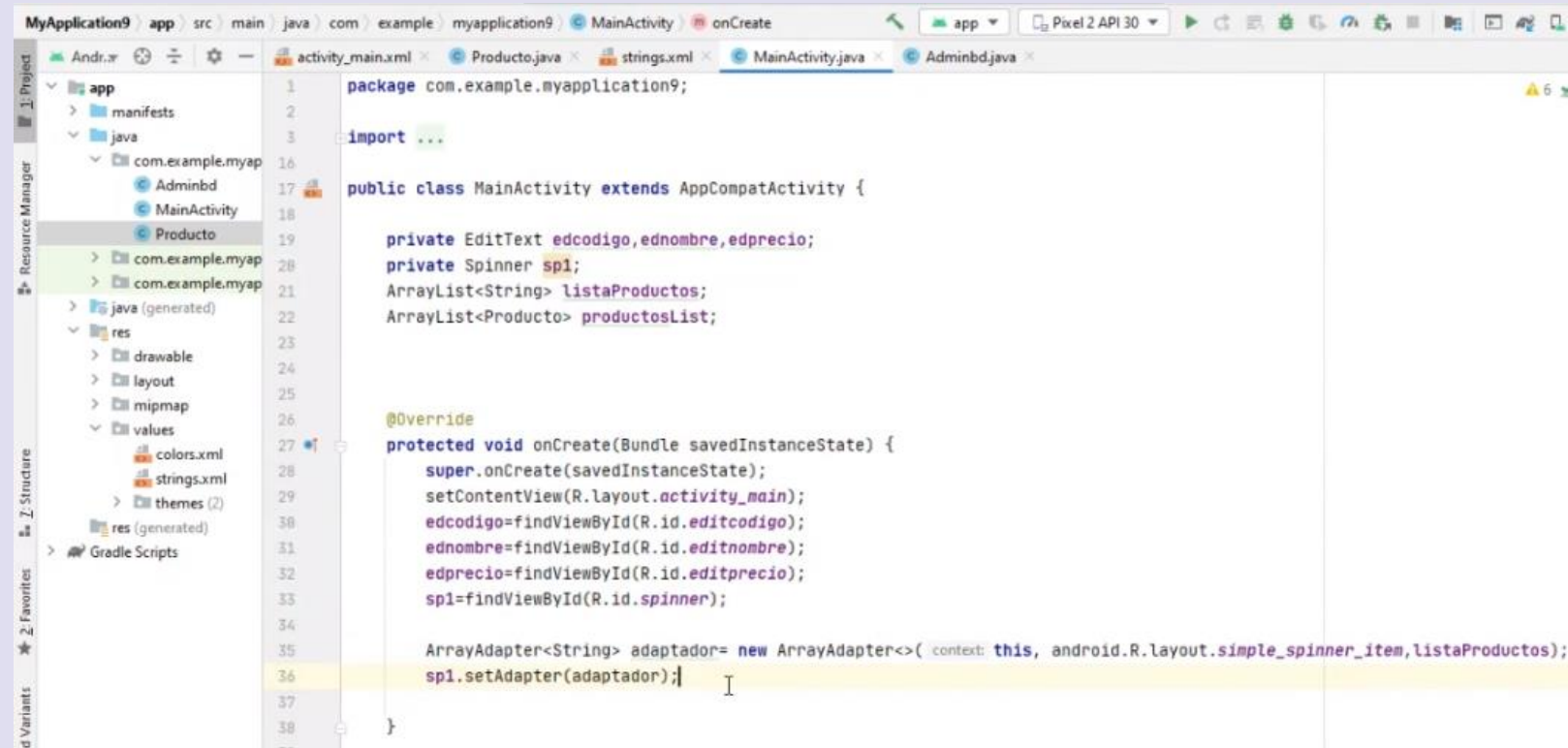
```
MyApplication9 app src main java com example.myapplication9 MainActivity obtenerProductos app Pixel 2 API 30
activity_main.xml Producto.java strings.xml MainActivity.java Adminbd.java
40 Producto p1=new Producto();
41 productosList= new ArrayList<Producto>();
42
43 Cursor fila= base.rawQuery( sql: "select * from producto", selectionArgs: null);
44 while(fila.moveToNext()){
45     p1.setCodigo(fila.getInt( columnIndex: 0));
46     p1.setNombre(fila.getString( columnIndex: 1));
47     p1.setPrecio(fila.getInt( columnIndex: 2));
48     productosList.add(p1);
49 }
50
51 public void obtenerProductos(){
52     listaProductos = new ArrayList<String>();
53     for(int i=0;i<productosList.size();i++){
54         listaProductos.add(productosList.get(i).getCodigo()+" - "
55         +productosList.get(i).getNombre()+" - "+productosList.get(i).getPrecio());
56     }
57 }
58
59 public void crear(View v){
60     Adminbd admin=new Adminbd( context: this, name: "Productos", factory: null, version: 1);
61 }
```



```
MyApplication9 app src main java com example.myapplication9 MainActivity consultarListaProductos app Pixel 2 API 30
activity_main.xml Producto.java strings.xml MainActivity.java Adminbd.java
37 public void consultarListaProductos(){
38     Adminbd admin=new Adminbd( context: this, name: "Productos", factory: null, version: 1);
39     SQLiteDatabase base=admin.getReadableDatabase();
40     Producto p1=new Producto();
41     productosList= new ArrayList<Producto>();
42
43     Cursor fila= base.rawQuery( sql: "select * from producto", selectionArgs: null);
44     while(fila.moveToNext()){
45         p1.setCodigo(fila.getInt( columnIndex: 0));
46         p1.setNombre(fila.getString( columnIndex: 1));
47         p1.setPrecio(fila.getInt( columnIndex: 2));
48         productosList.add(p1);
49     }
50     obtenerProductos();
51 }
52
53 public void obtenerProductos(){
54     listaProductos = new ArrayList<String>();
55     for(int i=0;i<productosList.size();i++){
56         listaProductos.add(productosList.get(i).getCodigo()+" - "
57         +productosList.get(i).getNombre()+" - "+productosList.get(i).getPrecio());
58     }
59 }
60
61 public void crear(View v){
62     Adminbd admin=new Adminbd( context: this, name: "Productos", factory: null, version: 1);
63 }
```

Y estarían listos nuestros métodos.

Ahora debemos pasar el arreglo a nuestro spinner



The screenshot shows the Android Studio interface with the 'MyApplication9' project. The 'Productos' item in the 'res' folder is selected in the Project Manager. The 'MainActivity.java' file is open in the editor, showing the following code:

```
package com.example.myapplication9;

import ...

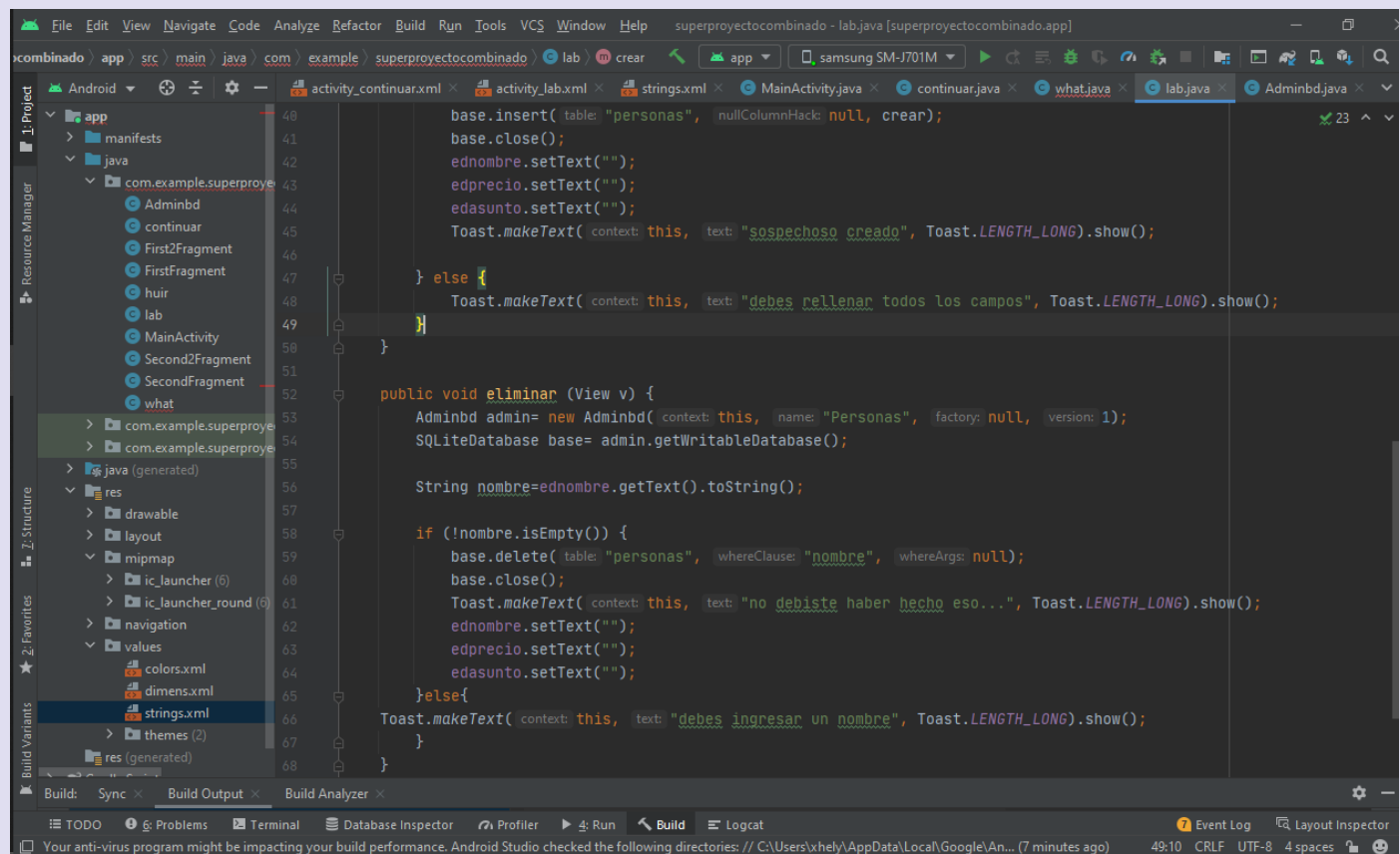
public class MainActivity extends AppCompatActivity {

    private EditText edcodigo, ednombre, edprecio;
    private Spinner spl;
    ArrayList<String> listaProductos;
    ArrayList<Producto> productosList;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        edcodigo=findViewById(R.id.editcodigo);
        ednombre=findViewById(R.id.editnombre);
        edprecio=findViewById(R.id.editprecio);
        spl=findViewById(R.id.spinner);

        ArrayAdapter<String> adaptador= new ArrayAdapter<>( context: this, android.R.layout.simple_spinner_item, listaProductos);
        spl.setAdapter(adaptador);}

}
```



The screenshot shows the Android Studio interface with the 'superproyectocombinado' project. The 'lab' item in the 'res' folder is selected in the Project Manager. The 'MainActivity.java' file is open in the editor, showing the following code:

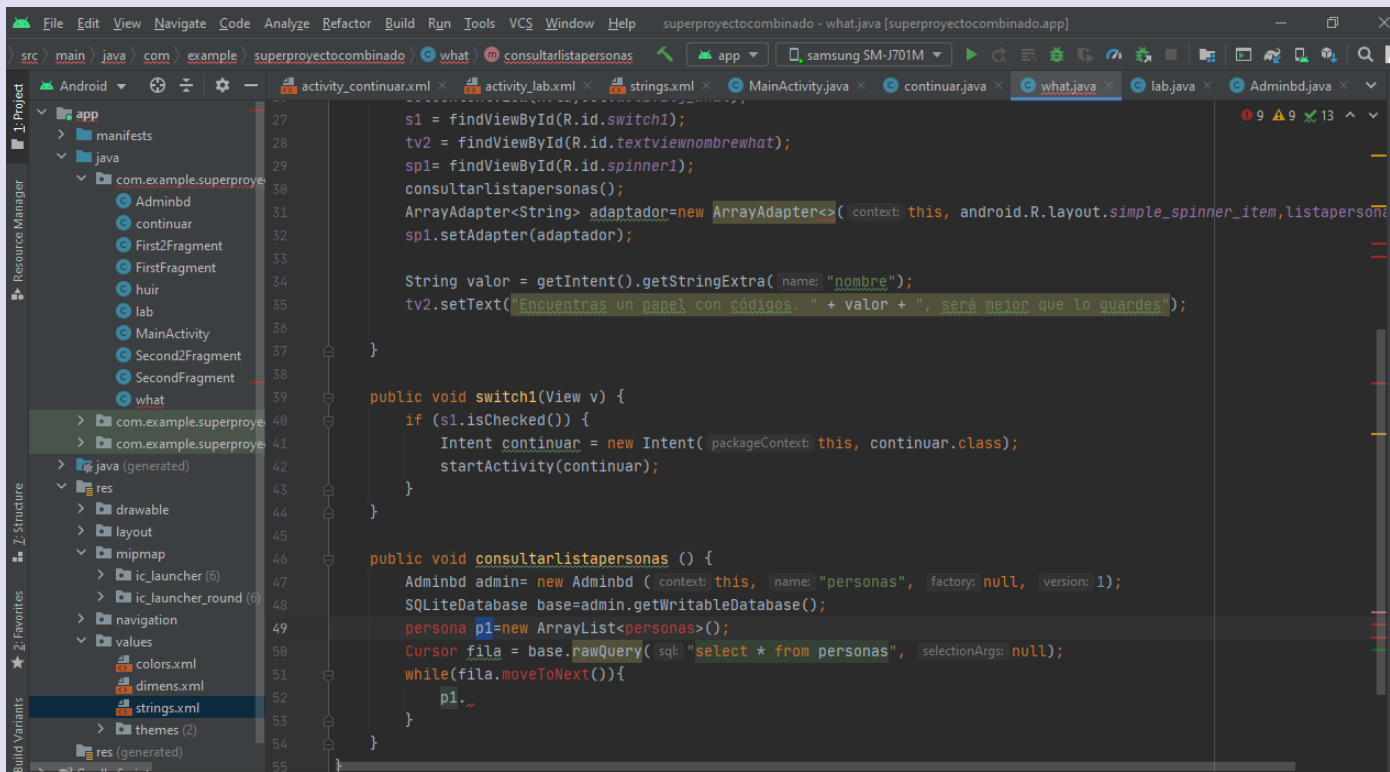
```
base.insert( table: "personas", nullColumnHack: null, crear);
base.close();
ednombre.setText("");
edprecio.setText("");
edasunto.setText("");
Toast.makeText( context: this, text: "sospechoso creado", Toast.LENGTH_LONG).show();
} else {
    Toast.makeText( context: this, text: "debes rellenar todos los campos", Toast.LENGTH_LONG).show();
}

}

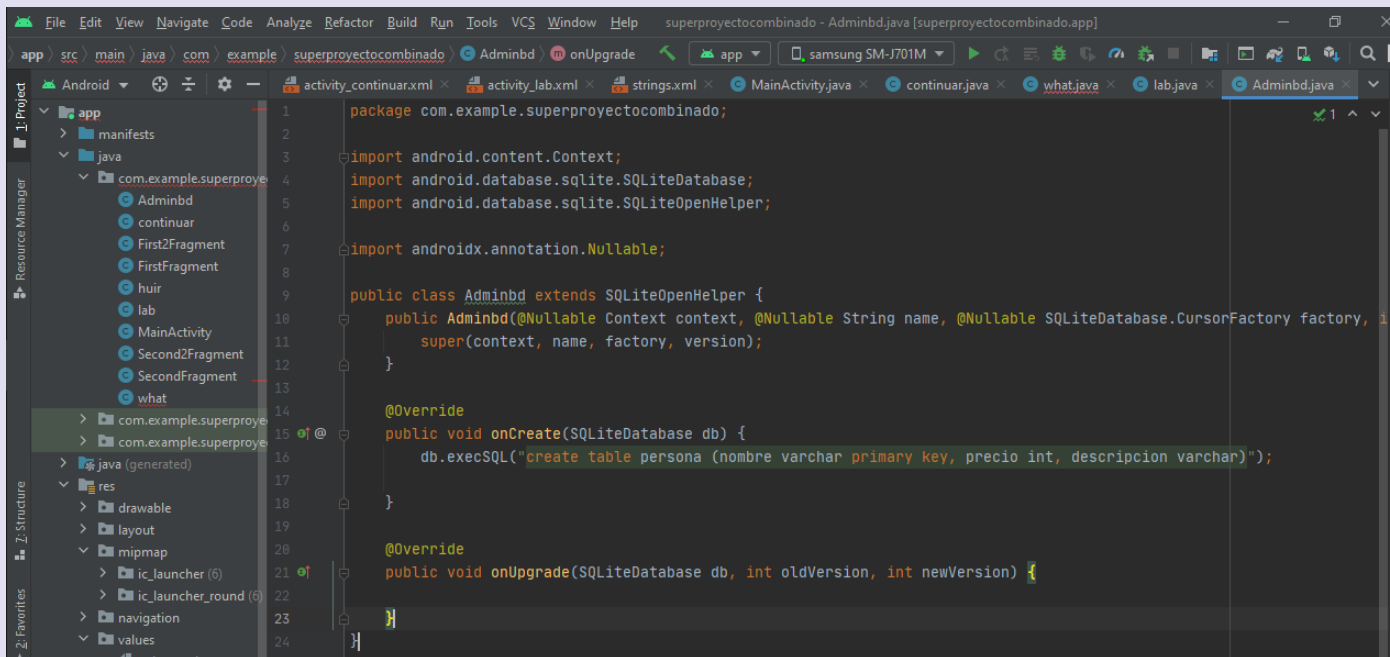
public void eliminar (View v) {
    Adminbd admin= new Adminbd( context: this, name: "Personas", factory: null, version: 1);
    SQLiteDatabase base= admin.getWritableDatabase();

    String nombre=ednombre.getText().toString();

    if (!nombre.isEmpty()) {
        base.delete( table: "personas", whereClause: "nombre", whereArgs: null);
        base.close();
        Toast.makeText( context: this, text: "no debiste haber hecho eso...", Toast.LENGTH_LONG).show();
        ednombre.setText("");
        edprecio.setText("");
        edasunto.setText("");
    }else{
        Toast.makeText( context: this, text: "debes ingresar un nombre", Toast.LENGTH_LONG).show();
    }
}
```



```
27 s1 = findViewById(R.id.switch1);
28 tv2 = findViewById(R.id.textviewnombrewhat);
29 sp1 = findViewById(R.id.spinner1);
30 consultarlistapersonas();
31 ArrayAdapter<String> adaptador = new ArrayAdapter<>(context: this, android.R.layout.simple_spinner_item, listapersonas);
32 sp1.setAdapter(adaptador);
33
34 String valor = getIntent().getStringExtra( name: "nombre");
35 tv2.setText("Encuentras un papel con codigos. " + valor + ", será mejor que lo guardes");
36
37
38
39 public void switch1(View v) {
40     if (s1.isChecked()) {
41         Intent continuar = new Intent( packageContext: this, continuar.class);
42         startActivity(continuar);
43     }
44 }
45
46 public void consultarlistapersonas () {
47     Adminbd admin = new Adminbd ( context: this, name: "personas", factory: null, version: 1);
48     SQLiteDatabase base = admin.getWritableDatabase();
49     persona p1 = new ArrayList<personas>();
50     Cursor fila = base.rawQuery( sql: "select * from personas", selectionArgs: null);
51     while(fila.moveToNext()){
52         p1..
53     }
54 }
55 }
```



```
1 package com.example.superproyectocombinado;
2
3 import android.content.Context;
4 import android.database.sqlite.SQLiteDatabase;
5 import android.database.sqlite.SQLiteOpenHelper;
6
7 import androidx.annotation.Nullable;
8
9 public class Adminbd extends SQLiteOpenHelper {
10     public Adminbd(@Nullable Context context, @Nullable String name, @Nullable SQLiteDatabase.CursorFactory factory, int version) {
11         super(context, name, factory, version);
12     }
13
14     @Override
15     public void onCreate(SQLiteDatabase db) {
16         db.execSQL("create table persona (nombre varchar primary key, precio int, descripcion varchar)");
17     }
18
19     @Override
20     public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
21     }
22 }
23
24 }
```

Trabajando, trabajando... aún no lo termino pero creo que va bien encaminado ¿?. PD. Logré arreglar el proyecto de las Activities que no me funcionaba (∴. Siento que voy leyendo en general, necesito mucho apoyo para poder completar las task pedidas, pero he aprendido un montón y me ha gustado mucho todo esto de programar (: quiero seguir entrenando para poder crear cosas geniales (porque además me entretiene un montón). Y también necesito adquirir fluidez.