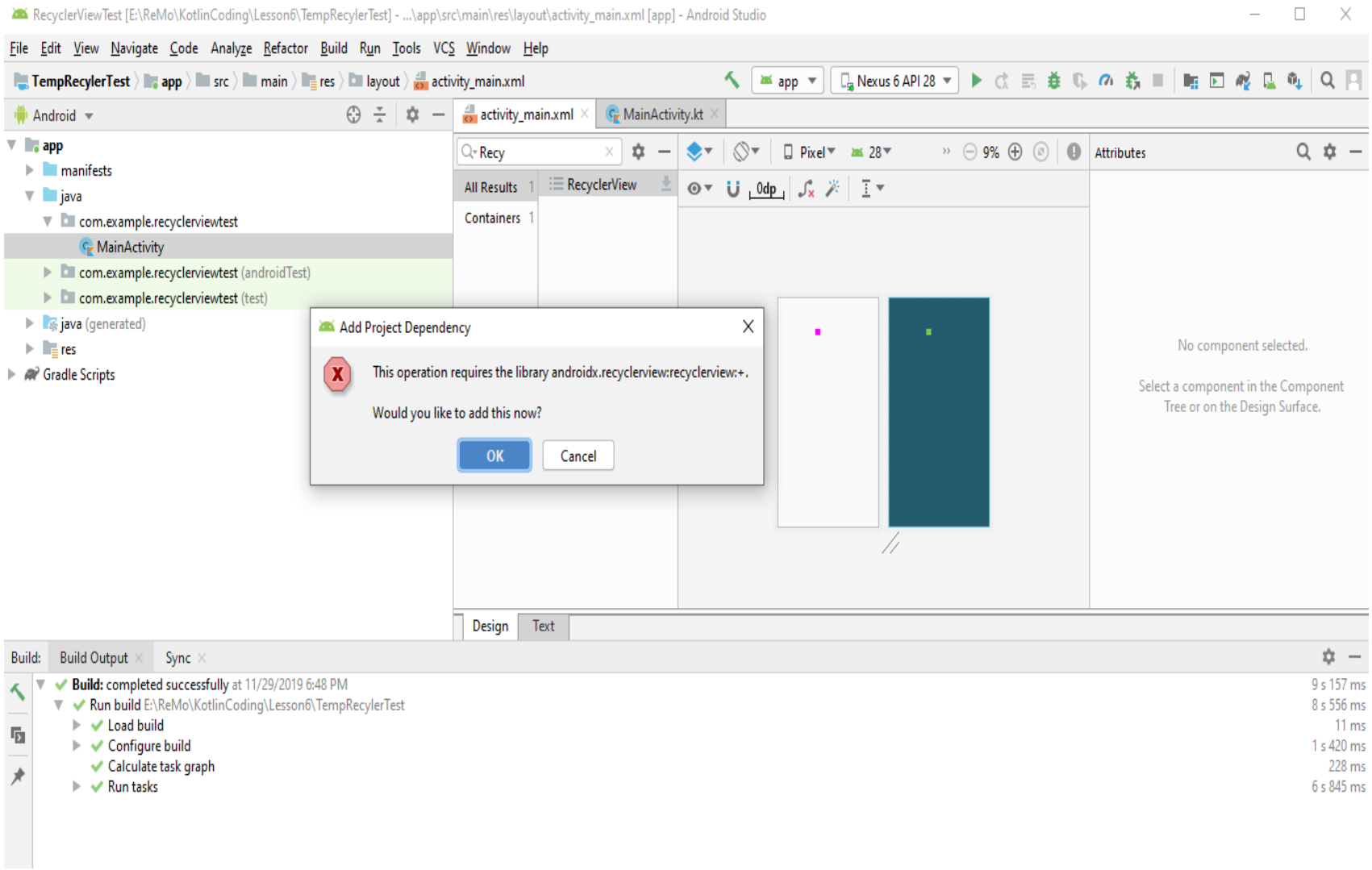
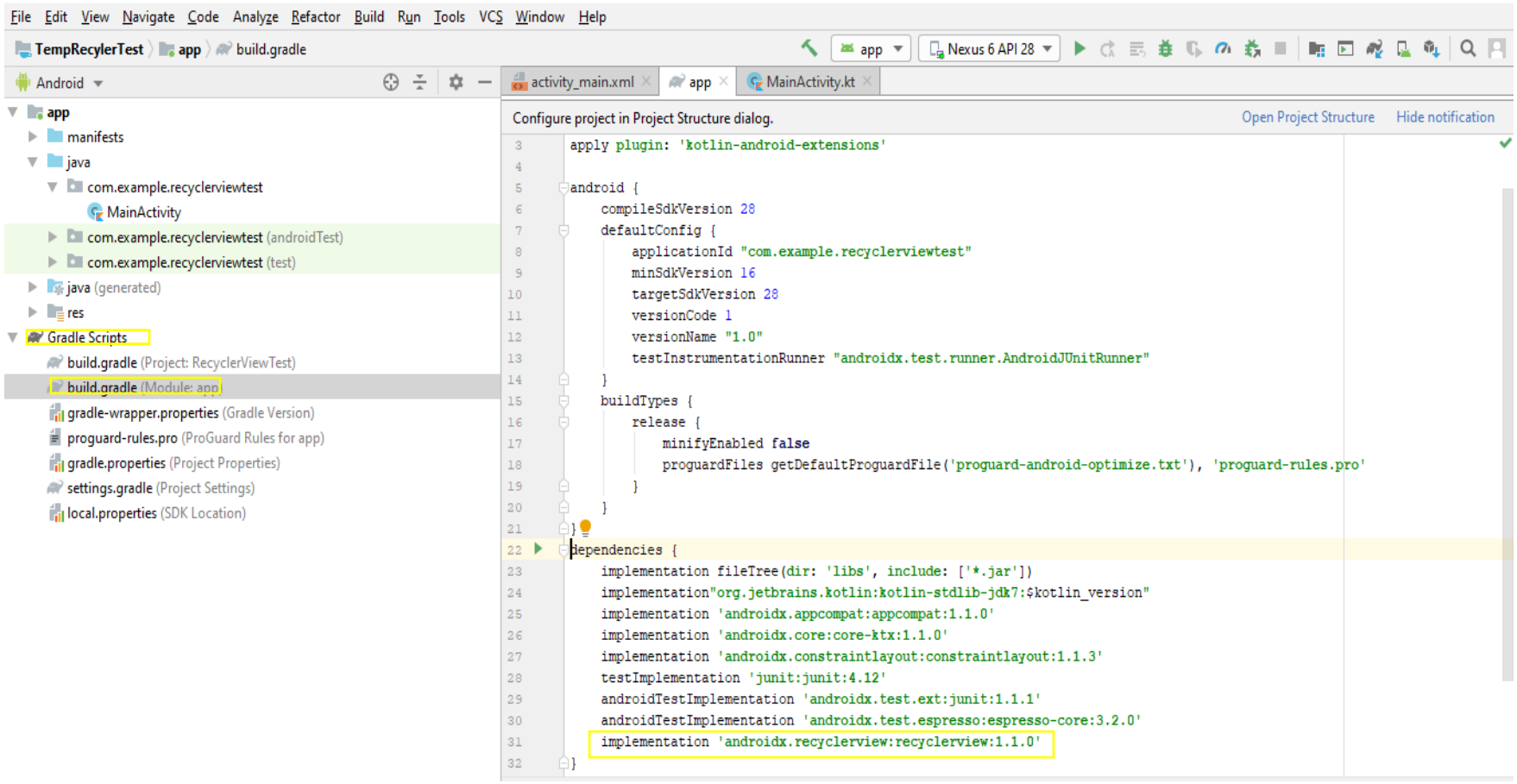
Step 1: Create a new Project and open activity_main.xml, remove the TextView. Go to your XML Design editor and drag RecyclerView, automatically you will get the warning to include RecyclerView Dependency on the gradle. Click OK.

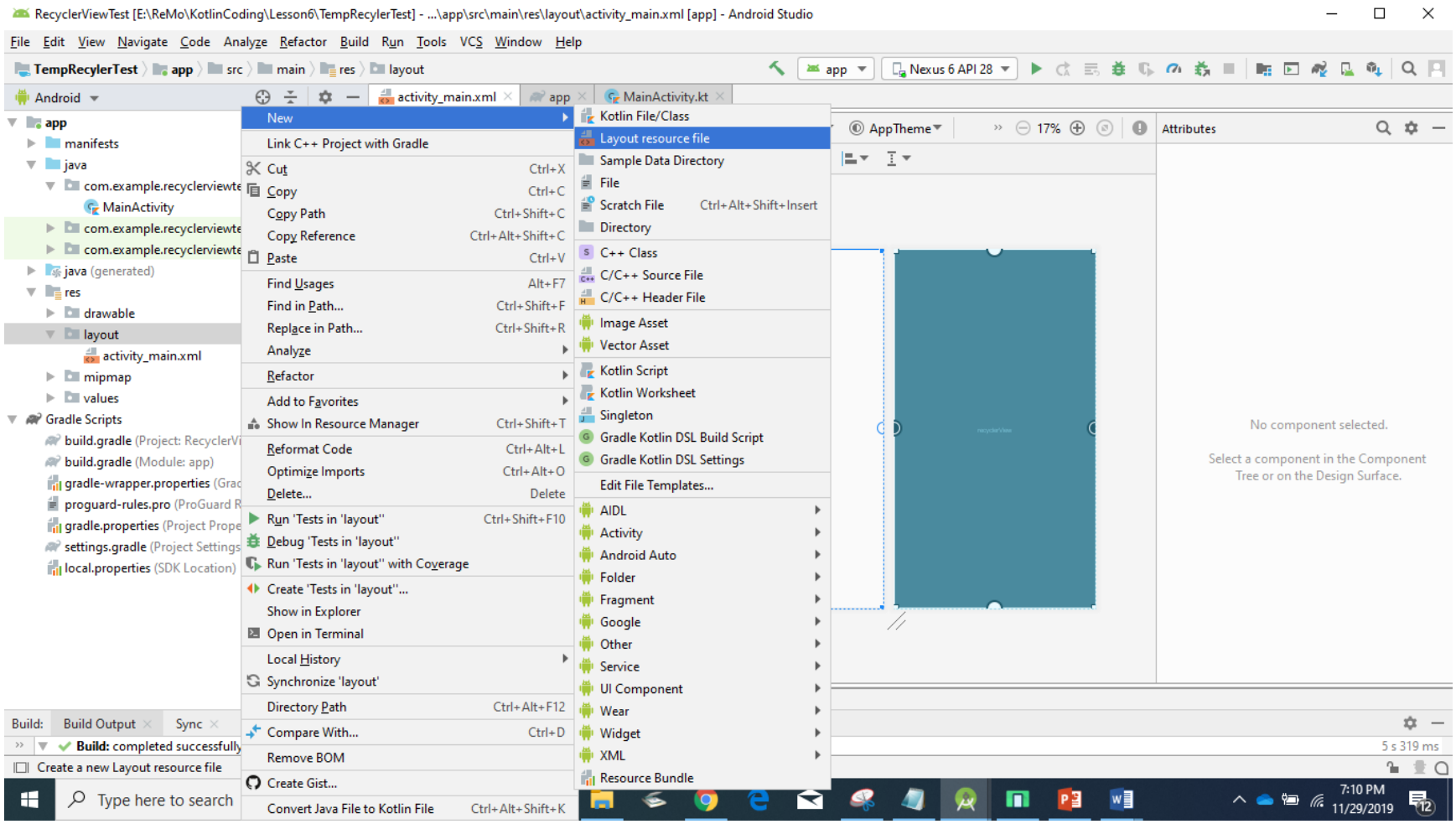


Step 2: To check the added dependency, selecting the highlighted part on your Project Explorer. You can able to see the below dependency on the build.gradle(Module:app)

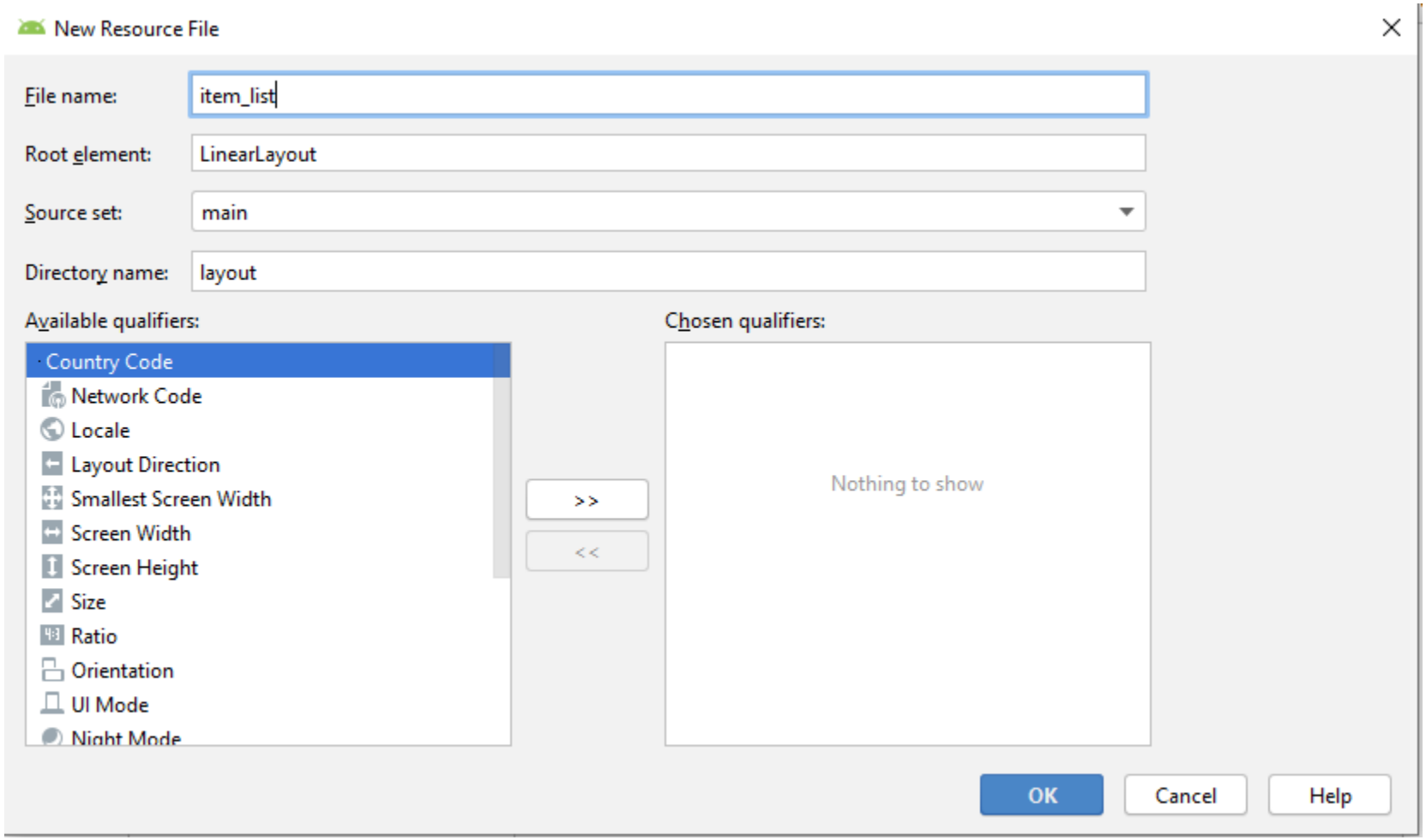
implementation **'androidx.recyclerview:recyclerview:1.1.0'**



Step 3: Create Layout according to your requirement to list on RecyclerView. So create a new Layout resource file as specified below



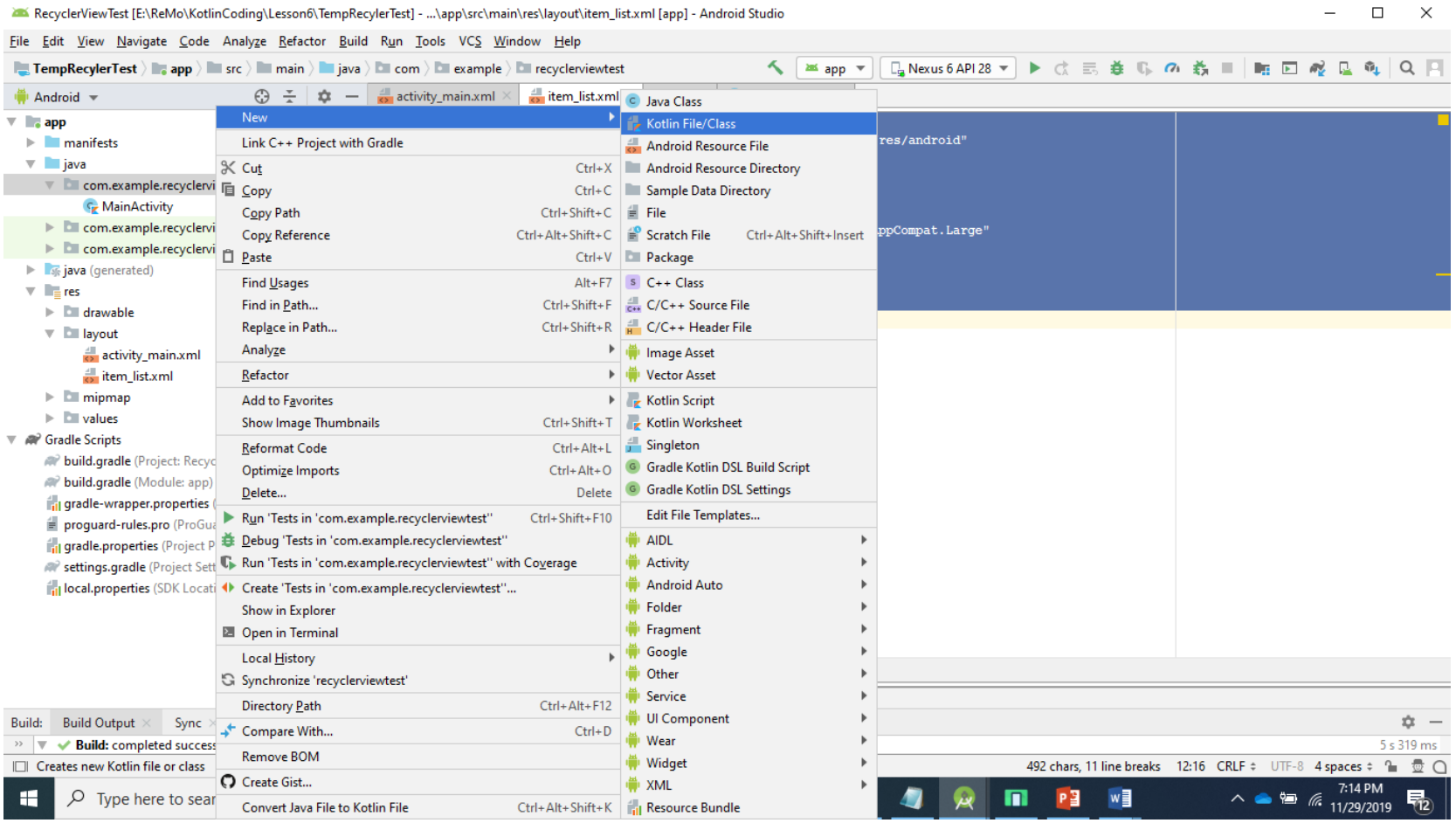
Step4 : Provide the name of your Layout



Step 5: Type the following code in your item_list.xml. Only one TextView is used for this example. You can customize according to your needs.

```
<?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="wrap_content">
   <TextView
      android:textAppearance="@style/Base.TextAppearance.AppCompat.Large"
      android:id="@+id/name"
      android:text="Course Name"
      android:layout_width="match_parent"
      android:layout_height="wrap_content" />
</LinearLayout>
```

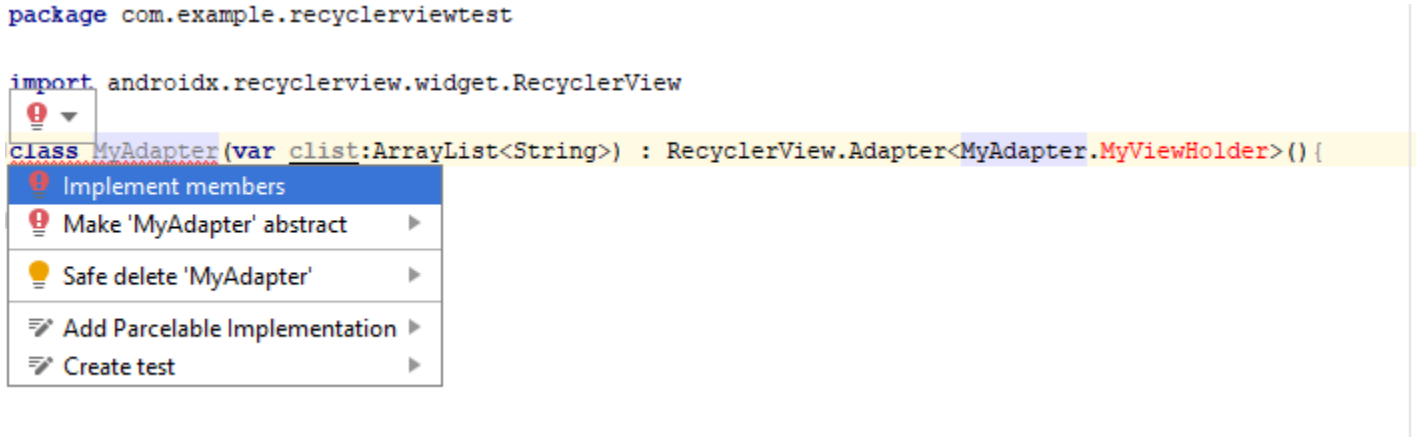
Step 6: Create MyAdapter.kt as per the screen shot and name is as MyAdapter.kt.



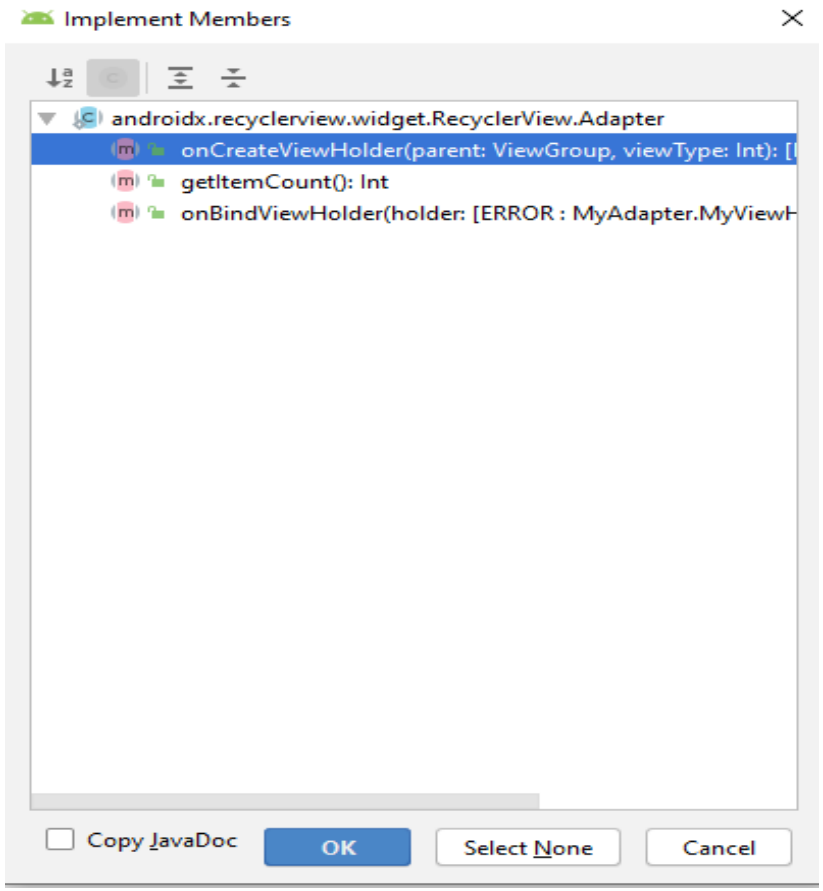
Step 7: Create a Class as Mentioned below. Your class should have a constructor to accept the input list and should inherit from RecyclerView.Adapter<Type>. The type be your own inner class Which is here MyViewHolder from MyAdapter class. MyViewHolder inherit from RecyclerView.ViewHolder(itemView)

```
class MyAdapter(var clist:ArrayList<String>) : RecyclerView.Adapter<MyAdapter.MyViewHolder>(){
}
```

Step 9: You will get errors as below



Step 10: Click the Red Error, you will get the methods needs to override from RecyclerView.Adapter, select all the methods as per the screen shot and click OK.



Step 11: Your code will look like this

```
class MyAdapter(var clist:ArrayList<String>) : RecyclerView.Adapter<MyAdapter.MyViewHolder>(){
    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): MyAdapter.MyViewHolder {
        TODO("not implemented") //To change body of created functions use File | Settings | File Templates.
    }

    override fun getItemCount(): Int {
        TODO("not implemented") //To change body of created functions use File | Settings | File Templates.
    }

    override fun onBindViewHolder(holder: MyAdapter.MyViewHolder, position: Int) {
        TODO("not implemented") //To change body of created functions use File | Settings | File Templates.
    }

}
```

Step 12: class MyAdapter accepts the list of Strings to show it on the RecyclerViewList from the constructor and also inherit from RecyclerView.Adapter<MyAdapter.MyViewHolder>() class as a type of MyViewHolder need to implement as a Inner class to show the view. Here is the complete code to get rid of errors. Remove the TODO part from the above code and use given below code

```
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import androidx.recyclerview.widget.RecyclerView
import kotlinx.android.synthetic.main.item_list.view.*

class MyAdapter(var clist:ArrayList<String>) : RecyclerView.Adapter<MyAdapter.MyViewHolder>(){
    // Inflate the Layout to set in the RecyclerView and return the ViewHolder object
    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): MyAdapter.MyViewHolder {
        val v = LayoutInflater.from(parent?.context).inflate(R.layout.item_list, parent, false)
        return MyViewHolder(v);
    }
```

```
    // @return Size of the list of data.

    override fun getItemCount(): Int {
        return clist.size
    }
    // Binds the Array<String> values in to the ViewHolder name TextView designed in the item_list.xml
    override fun onBindViewHolder(holder: MyAdapter.MyViewHolder, position: Int) {
            holder.cname.text = clist[position]
    }

    /*RecyclerView.Adapter accepts the generic type of your Adapter inner class ViewHolder type.
 In this example Adapter class name is MyAdapter and the MyViewHolder  is the inner class */
    class MyViewHolder(itemView: View): RecyclerView.ViewHolder(itemView) {
        var cname = itemView.name

    }
    }
```

Step 13: Do the implementation in the MainActivity.kt

MainActivity.kt

```
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import androidx.recyclerview.widget.LinearLayoutManager
import kotlinx.android.synthetic.main.activity_main.*

class MainActivity : AppCompatActivity() {

     // Declare ArrayList to store the course list
    lateinit var list:ArrayList<String>
    override fun onCreate(savedInstanceState: Bundle?) {
```

```
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        recyclerView?.layoutManager = LinearLayoutManager(this)

        // Intialize the values
        list = ArrayList(arrayListOf("CS471 Parallel Programming","CS473 Mobile Device Programming","CS472
                                     WAP","CS390 FPP","CS401 MPP","CS435Algorithms"))
        // Create an object for the MyAdapter and pass the values to its constructor
        var adpt = MyAdapter(list)
        // Set the MyAdapter object to your RecyclerView component
        recyclerView.adapter = adpt
    }
}
```

Step 14: If you do everything as per the screen the instructions, after running will get the output as below