Android RecyclerView in Kotlin

Android recyclerView in Kotlin

Step by Step Implementation

Step 1: Create a New Project

On the Welcome screen of Android Studio, click on **Create New Project**. If you have a project already opened, **Go to File > New > New Project**. Then select a Project Template window, select **Empty Activity** and click **Next**. Enter your App Name in the Name field. Select **Kotlin** from the Language drop-down menu.

Step 2: Add the Dependencies

Go to app **< Gradle Scripts < gradle.build(Module: app)** and add the following dependencies.

```
dependencies{
  // for adding recyclerview
  implementation 'androidx.recyclerview:recyclerview:1.2.0'

  // for adding cardview
  implementation 'androidx.cardview:cardview:1.0.0'
}
```

Step 3: Go to activity_main.xml and add the following code

Add RecyclerView to activity_main.xml you can add it from the drag and drop from the design section or you can add it manually by writing some initial characters of RecyclerView then the IDE will give you suggestions for RecyclerView then select RecyclerView it will automatically add it to your layout file.

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

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerview"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:itemCount="5"
        tools:listitem="@layout/card_view_design" />

</LinearLayout>
```

Step 4: Create a New Layout Resource File

Now create a new Layout Resource File which will be used to design our **CardView** layout. Go to **app > res > layout > right-click on layout > New > Layout Resource File** and name that file as **card_view_design** and add the code provided below. In this file, you can design the layout to show it into the RecyclerView.

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_margin="10dp"
    app:cardElevation="6dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:padding="5dp">

        <ImageView
            android:id="@+id/imageview"
            android:layout_width="40dp"
            android:layout_height="40dp" />

        <TextView
            android:id="@+id/textView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="10dp"
            android:layout_marginLeft="15dp"
            android:text="Item"
            android:textSize="20sp"
            android:textStyle="bold" />

    </LinearLayout>

</androidx.cardview.widget.CardView>
```

Step 5: Create a new Kotlin class

Go to **app > java > package name > right-click > New > Kotlin class/file** and choose Data class from the list. Name that file as **ItemsViewModel** and then click on OK. This file will hold the information of every item which you want to show in your RecyclerView.

```
data class ItemsViewModel(val image: Int, val text: String) {
}
```

Step 6: Create Adapter Class

Go to **app > java > package name > right-click > New > Kotlin class/file** and name that file as **CustomAdapter** and then click on **OK**. After this add the code provided below. Comments are added inside the code to understand the code in more detail.

This class contains some important functions to work with the RecyclerView these are as follows:

- **onCreateViewHolder():** This function sets the views to display the items.
- **onBindViewHolder():** This function is used to bind the list items to our widgets such as TextView, ImageView, etc.
- **getItemCount():** It returns the count of items present in the list.

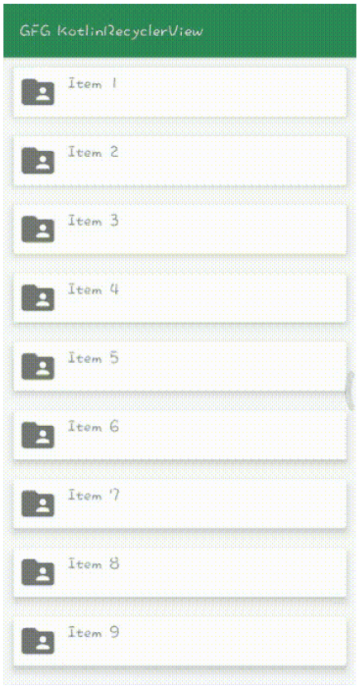
```
data class ItemsViewModel(val image: Int, val text: String) {
}
```

```
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.ImageView
import android.widget.TextView
import androidx.recyclerview.widget.RecyclerView

class CustomAdapter(private val mList: List<ItemsViewModel>) : RecyclerView.Adapter<CustomAdapter.ViewHolder>() {

    // create new views
    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): ViewHolder {
        // inflates the card_view_design view
        // that is used to hold list item
        val view = LayoutInflater.from(parent.context)
            .inflate(R.layout.card_view_design, parent, false)

        return ViewHolder(view)
    }

    // binds the list items to a view
    override fun onBindViewHolder(holder: ViewHolder, position: Int) {

        val ItemsViewModel = mList[position]

        // sets the image to the imageview from our itemHolder class
        holder.imageView.setImageResource(ItemsViewModel.image)

        // sets the text to the textview from our itemHolder class
        holder.textView.text = ItemsViewModel.text

    }

    // return the number of the items in the list
    override fun getItemCount(): Int {
        return mList.size
    }

    // Holds the views for adding it to image and text
    class ViewHolder(ItemView: View) : RecyclerView.ViewHolder(ItemView) {
        val imageView: ImageView = itemView.findViewById(R.id.imageview)
        val textView: TextView = itemView.findViewById(R.id.textView)
    }
}
```

Step 7: Working with the MainActivity.kt

Go to the MainActivity.kt file and refer to the following code. Below is the code for the MainActivity.kt file. Comments are added inside the code to understand the code in more detail.

```
import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity
import androidx.recyclerview.widget.LinearLayoutManager
import androidx.recyclerview.widget.RecyclerView

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        // getting the recyclerview by its id
        val recyclerview = findViewById<RecyclerView>(R.id.recyclerview)

        // this creates a vertical layout Manager
        recyclerview.layoutManager = LinearLayoutManager(this)

        // ArrayList of class ItemsViewModel
        val data = ArrayList<ItemsViewModel>()

        // This loop will create 20 Views containing
        // the image with the count of view
        for (i in 1..20) {
            data.add(ItemsViewModel(R.drawable.ic_baseline_folder_24, "Item " + i))
        }

        // This will pass the ArrayList to our Adapter
        val adapter = CustomAdapter(data)

        // Setting the Adapter with the recyclerview
        recyclerview.adapter = adapter

    }
}
```


GFG KotlinRecyclerView
Item 1
Item 2
Item 3
Item 4
Item 5
Item 6
Item 7
Item 8
Item 9