



VIT[®]

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

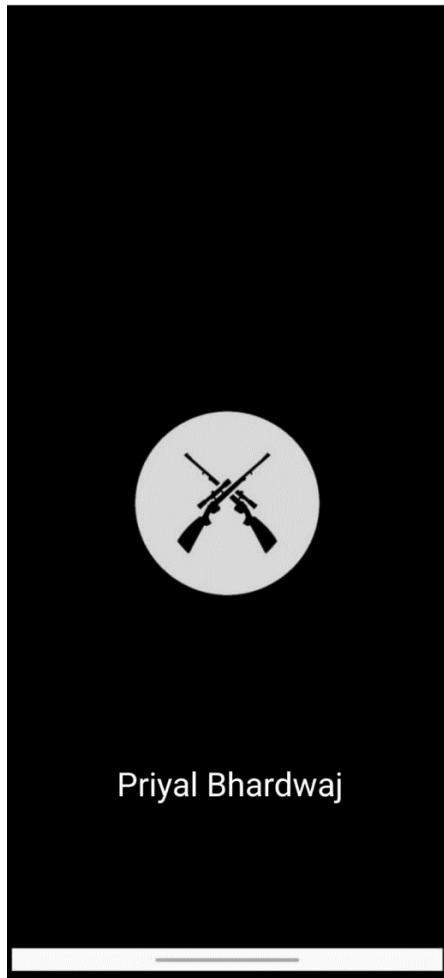
School of Information Technology and Engineering
Digital Assignment - II, DECEMBER 2021
B.Tech., Fall-2021-2022

NAME	PRIYAL BHARDWAJ
REG. NO.	18BIT0272
COURSE CODE	ITE1016
COURSE NAME	MOBILE APPLICATION DEVELOPMENT
SLOT	A1+TA1
FACULTY	Prof. SRINIVASAN P

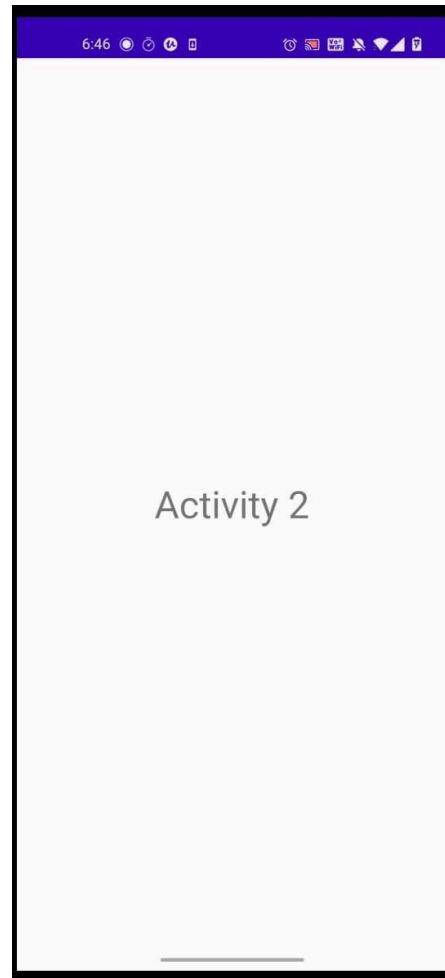
Implement the following features/ functionalities in your program and upload your own exercises with output. Output should be Unique.

1. Splash Screen

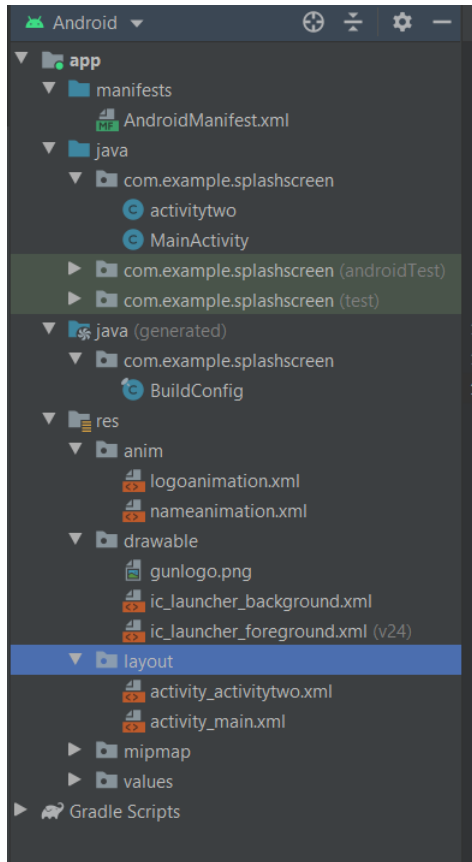
Output:



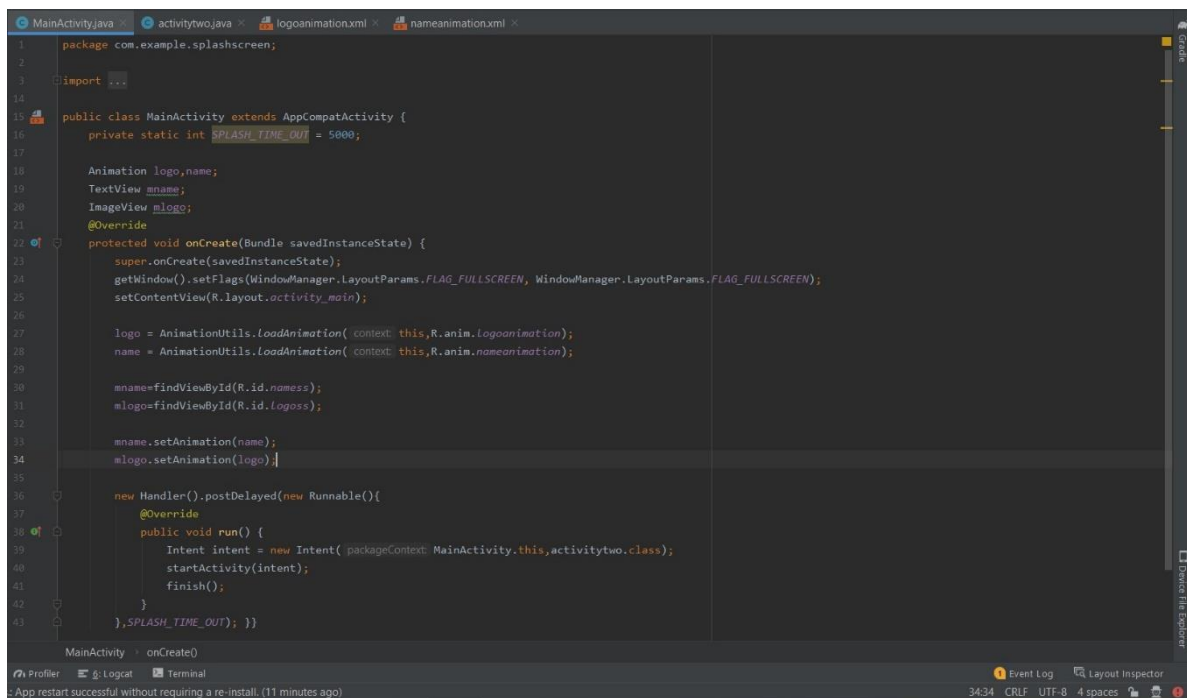
After 5 seconds moves to activitytwo.java:



Project Directory:



MainActivity.java



Animation XML programs:

logoanimation.xml

```
MainActivity.java × activitytwo.java × logoanimation.xml × nameanimation.xml ×
1  <?xml version="1.0" encoding="utf-8"?>
2  <set xmlns:android="http://schemas.android.com/apk/res/android">
3
4      <translate
5          android:duration="1500"
6          android:fromYDelta="-500%"
7          android:fromXDelta="0%"
8      />
9      <alpha
10         android:fromAlpha="0.1"
11         android:toAlpha="1.0"
12         android:duration="1500" />
13  </set>
```

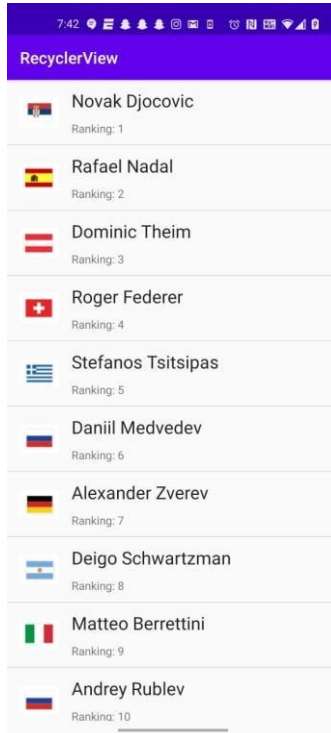
nameanimation.xml

```
MainActivity.java × activitytwo.java × logoanimation.xml × nameanimation.xml ×
1  <?xml version="1.0" encoding="utf-8"?>
2  <set xmlns:android="http://schemas.android.com/apk/res/android">
3
4      <translate
5          android:duration="1500"
6          android:fromYDelta="500%"
7          android:fromXDelta="0%"
8      />
9      <alpha
10         android:fromAlpha="0.1"
11         android:toAlpha="1.0"
12         android:duration="1500" />
13  </set>
```

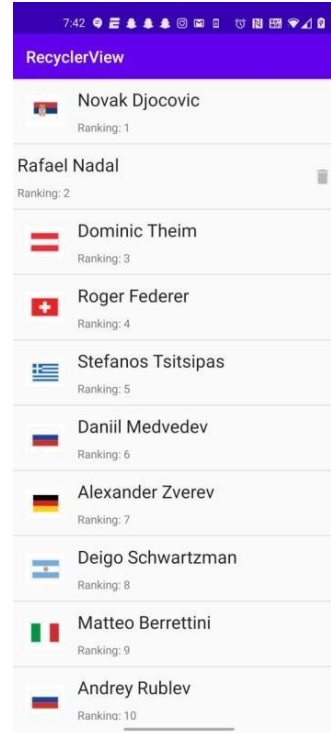
2. Swipe Functionality

Output:

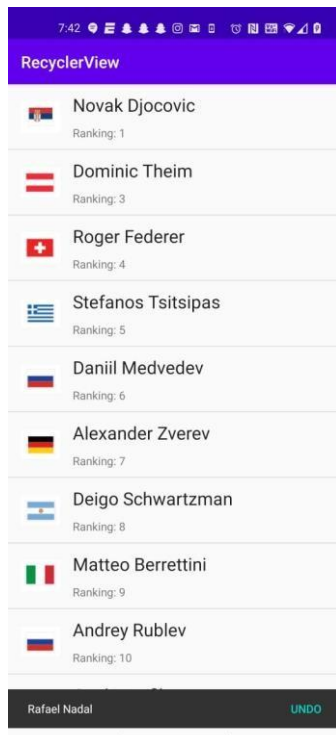
Initial screen using Recycler View:



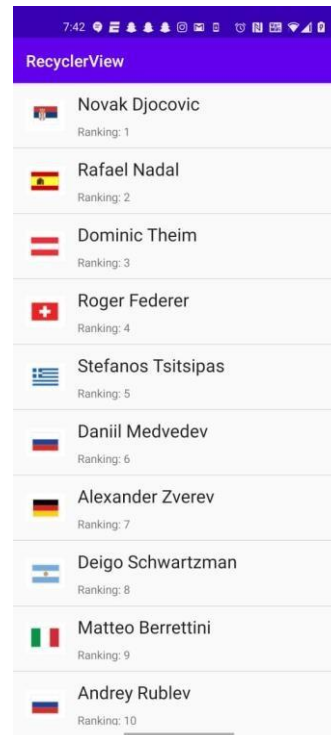
Swipe left to delete Rafael Nadal:



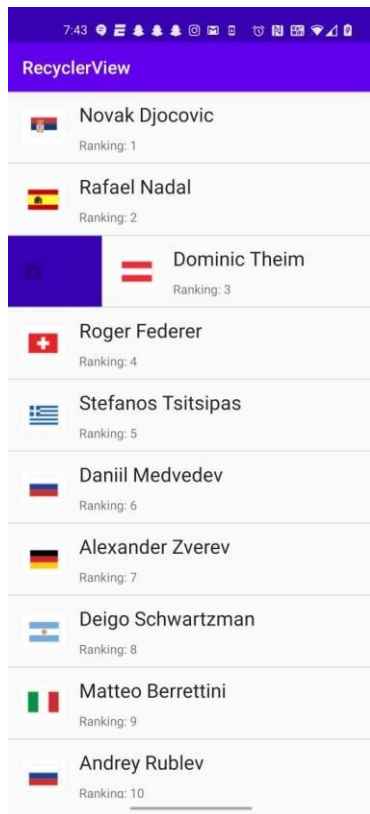
Rafael Nadal deleted:



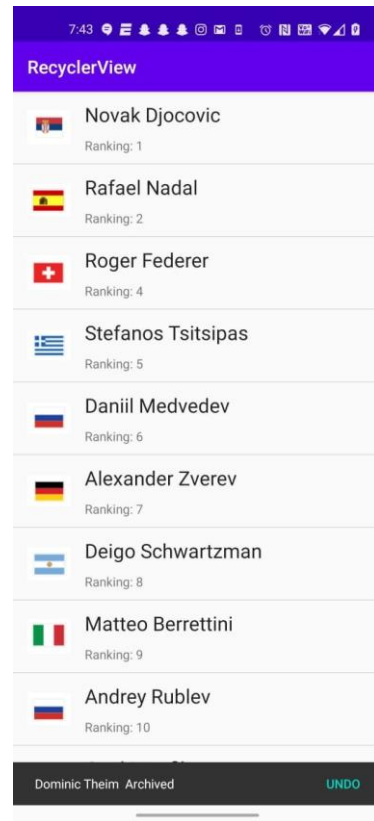
Undo Delete:



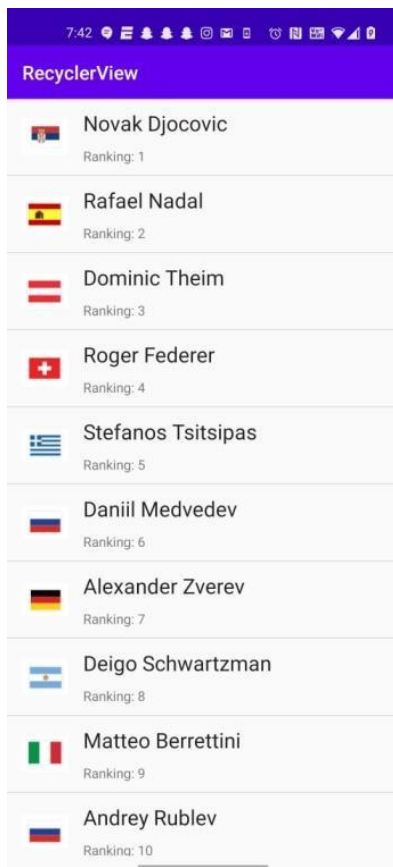
Swipe right to archive Dominic Them:



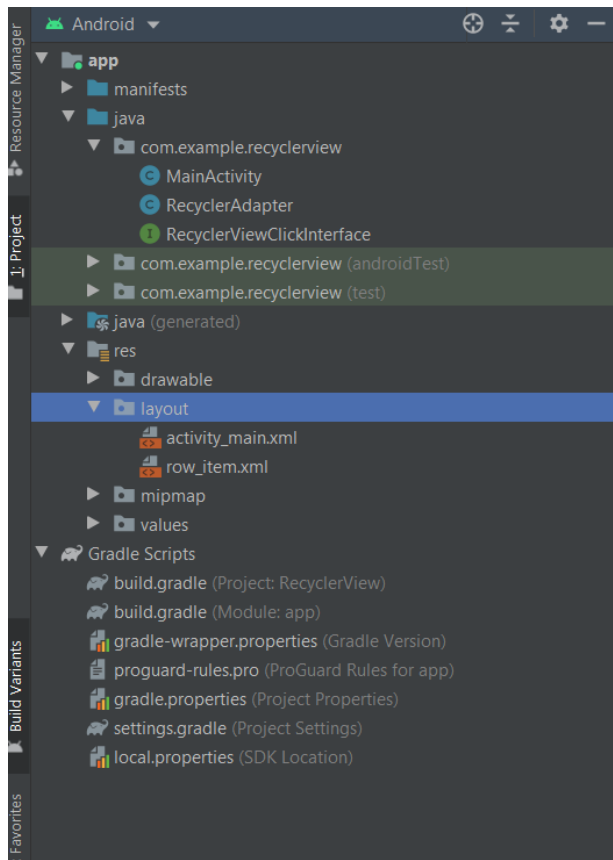
Dominic Them archived:



Undo archive:



Project Directory:



Swipe functionality code:

```
activity_main.xml x MainActivity.java x ItemTouchHelper.java x build.gradle (app) x RecyclerViewAdapter.java x RecyclerViewClickListener.java x row_item.xml x
91
92 swipeRefreshLayout = findViewById(R.id.swipeRefreshLayout);
93 swipeRefreshLayout.setOnRefreshListener(new SwipeRefreshLayout.OnRefreshListener() {
94     @Override
95     public void onRefresh() {
96         playerList.add("Miloš Raonic");
97         playerList.add("Felix Auger-Aliassime");
98         playerList.add("John Isner");
99         playerList.add("Dusan Lajovic");
100         playerList.add("Casper Ruud");
101
102         recyclerViewAdapter.notifyDataSetChanged();
103         swipeRefreshLayout.setRefreshing(false);
104     }
105 });
106
107
108 ItemTouchHelper itemTouchHelper = new ItemTouchHelper(simpleCallback);
109 itemTouchHelper.attachToRecyclerView(recyclerView);
110
111
112 String deletedPlayer = null;
113 List<String> archivedPlayers = new ArrayList<>();
114
115 ItemTouchHelper.SimpleCallback simpleCallback = new ItemTouchHelper.SimpleCallback( dragDirs: 0, swipeDirs: ItemTouchHelper.LEFT|ItemTouchHelper.RIGHT) {
116     @Override
117     public boolean onMove(@NonNull RecyclerView recyclerView, @NonNull RecyclerView.ViewHolder viewHolder, @NonNull RecyclerView.ViewHolder target) {
118
119         return false;
120     }
121 }
```

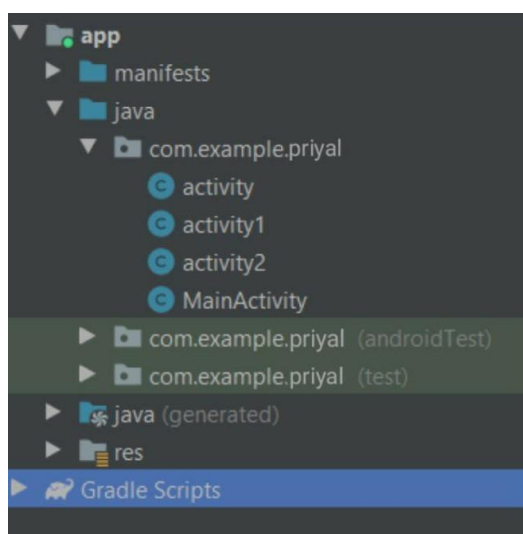
```
111
112 String deletedPlayer = null;
113 List<String> archivedPlayers = new ArrayList<>();
114
115 ItemTouchHelper.SimpleCallback simpleCallback = new ItemTouchHelper.SimpleCallback( dragDirs: 0, swipeDirs: ItemTouchHelper.LEFT|ItemTouchHelper.RIGHT) {
116     @Override
117     public boolean onMove(@NonNull RecyclerView recyclerView, @NonNull RecyclerView.ViewHolder viewHolder, @NonNull RecyclerView.ViewHolder target) {
118
119         return false;
120     }
121
122     @Override
123     public void onSwiped(@NonNull RecyclerView.ViewHolder viewHolder, int direction) {
124         final int position = viewHolder.getAdapterPosition();
125         switch(direction){
126             case ItemTouchHelper.LEFT:
127                 deletedPlayer = playerList.get(position);
128                 playerList.remove(position);
129                 recyclerViewAdapter.notifyItemRemoved(position);
130                 Snackbar.make(recyclerView,deletedPlayer,Snackbar.LENGTH_LONG).setAction( text: "Undo", new View.OnClickListener() {
131                     @Override
132                     public void onClick(View view) {
133                         playerList.add(position, deletedPlayer);
134                         recyclerViewAdapter.notifyItemInserted(position);
135                     }
136                 }).show();
137                 break;
138             case ItemTouchHelper.RIGHT:
139                 final String playerName = playerList.get(position);
140                 archivedPlayers.add(playerName);
141                 playerList.remove(position);
142                 recyclerViewAdapter.notifyItemRemoved(position);
143
144                 Snackbar.make(recyclerView, text: playerName + " Archived",Snackbar.LENGTH_LONG).setAction( text: "Undo", new View.OnClickListener() {
145                     @Override
146                     public void onClick(View view) {
147                         archivedPlayers.remove(archivedPlayers.lastIndexOf(playerName));
148                         playerList.add(position, playerName);
149                         recyclerViewAdapter.notifyItemInserted(position);
150                     }
151                 }).show();
152                 break;
153         }
154     }
155
156     @Override
157     public void onChildDraw(@NonNull Canvas c, @NonNull RecyclerView recyclerView, @NonNull RecyclerView.ViewHolder viewHolder, float dx, float dy, int actionState, boolean isCurrentlyActive)
158         new RecyclerViewSwipeDecorator.Builder( context: MainActivity.this, c, recyclerView, viewHolder, dx, dy, actionState, isCurrentlyActive)
159             .addSwipeLeftActionIcon(ContextCompat.getColor( context: MainActivity.this, R.color.colorPrimary))
160             .addSwipeLeftActionIcon(R.drawable.ic_baseline_delete_24)
161             .addSwipeRightBackgroundColor(ContextCompat.getColor( context: MainActivity.this, R.color.colorPrimaryDark))
162             .addSwipeRightActionIcon(R.drawable.ic_baseline_archive_24)
163             .create()
164             .decorate();
165         super.onChildDraw(c, recyclerView, viewHolder, dx, dy, actionState, isCurrentlyActive);
166     }
167
168     @Override
169     public void onItemClick(int position) {
170         Toast.makeText( context: this,playerList.get(position), Toast.LENGTH_SHORT).show();
171     }
172 }
```

```
145
146
147     @Override
148     public void onClick(View view) {
149         archivedPlayers.remove(archivedPlayers.lastIndexOf(playerName));
150         playerList.add(position, playerName);
151         recyclerViewAdapter.notifyItemInserted(position);
152     }
153     }).show();
154     break;
155 }
156
157 @Override
158 public void onChildDraw(@NonNull Canvas c, @NonNull RecyclerView recyclerView, @NonNull RecyclerView.ViewHolder viewHolder, float dx, float dy, int actionState, boolean isCurrentlyActive)
159     new RecyclerViewSwipeDecorator.Builder( context: MainActivity.this, c, recyclerView, viewHolder, dx, dy, actionState, isCurrentlyActive)
160         .addSwipeLeftActionIcon(ContextCompat.getColor( context: MainActivity.this, R.color.colorPrimary))
161         .addSwipeLeftActionIcon(R.drawable.ic_baseline_delete_24)
162         .addSwipeRightBackgroundColor(ContextCompat.getColor( context: MainActivity.this, R.color.colorPrimaryDark))
163         .addSwipeRightActionIcon(R.drawable.ic_baseline_archive_24)
164         .create()
165         .decorate();
166     super.onChildDraw(c, recyclerView, viewHolder, dx, dy, actionState, isCurrentlyActive);
167 }
168
169 @Override
170 public void onItemClick(int position) {
171     Toast.makeText( context: this,playerList.get(position), Toast.LENGTH_SHORT).show();
172 }
```

3. Card View



Project Directory:



Activity_main.xml

Top card (Novak Djokovic's card):

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView 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:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#000000"
    tools:context=".MainActivity"
    android:id="@+id/scrollView">

    <LinearLayout
        android:id="@+id/linearlayout"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <androidx.cardview.widget.CardView
            android:id="@+id/c"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="10sp"
            android:layout_marginLeft="20sp"
            android:layout_marginRight="20sp"
            app:cardCornerRadius="10sp">

            <LinearLayout
                android:id="@+id/linearlayout1"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="horizontal">
```

```
                <ImageView
                    android:id="@+id/image1"
                    android:layout_width="100sp"
                    android:layout_height="100sp"
                    android:src="@drawable/image_1"
                    android:padding="5sp"
                    />

                <TextView
                    android:id="@+id/t1"
                    android:layout_width="match_parent"
                    android:layout_height="match_parent"
                    android:gravity="center_horizontal"
                    android:paddingTop="25sp"
                    android:text="Novak Djokovic"
                    android:textAlignment="center"
                    android:textColor="#000000"
                    android:textSize="30sp" />

            </LinearLayout>

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

Card grid using Grid layout and Card View (1,2,3,4....cards):

```
<GridLayout
    android:id="@+id/gridlayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10sp"
    android:layout_marginLeft="10sp"
    android:layout_marginRight="10sp"
    android:rowCount="2"
    android:columnCount="2"
>
<androidx.cardview.widget.CardView
    android:id="@+id/c1"
    android:layout_gravity="fill"
    android:layout_rowWeight="1"
    android:layout_columnWeight="1"
    app:cardCornerRadius="10sp"
    android:layout_margin="10sp"

    >
    <TextView
        android:id="@+id/t2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="1"
        android:textAlignment="center"
        android:layout_gravity="center"
        android:textSize="20sp"
        android:textColor="#000000"
        android:paddingTop="30sp"
        android:paddingBottom="30sp"
```

```
<TextView
    android:id="@+id/t2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="1"
    android:textAlignment="center"
    android:layout_gravity="center"
    android:textSize="20sp"
    android:textColor="#000000"
    android:paddingTop="30sp"
    android:paddingBottom="30sp"
    >

</TextView>

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

mainActivity.java

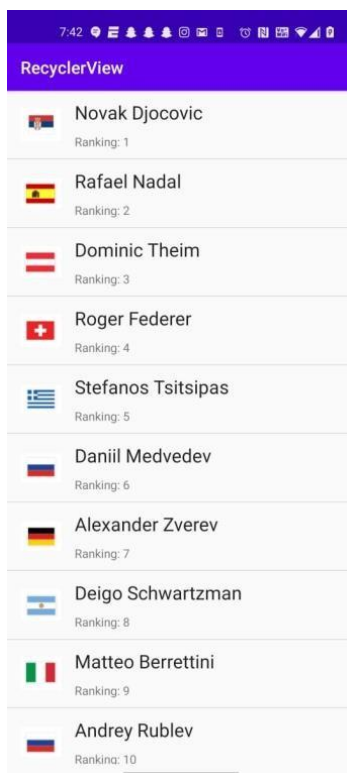
```
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ImageView;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

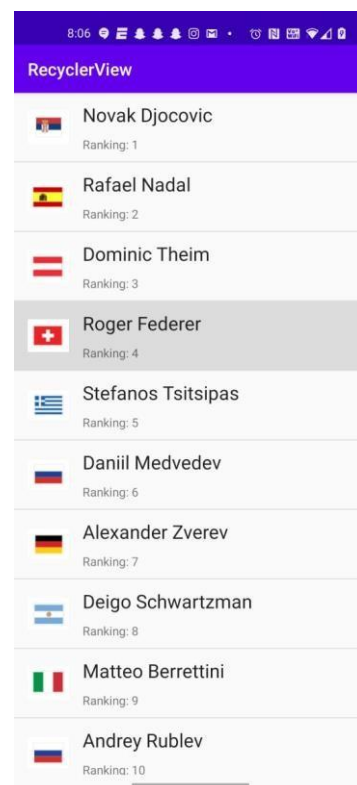
        CardView card, card1, card2;
        card=(CardView)findViewById(R.id.c);
        card1=(CardView)findViewById(R.id.c1);
        card2=(CardView)findViewById(R.id.c2);
        card.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i= new Intent(getApplicationContext(),activity.class);
                startActivity(i);
            }
        });
        card1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i= new Intent(getApplicationContext(),activity1.class);
                startActivity(i);
            }
        });
        card2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i= new Intent(getApplicationContext(),activity2.class);
                startActivity(i);
            }
        });
    }
}
```

4. RecyclerView

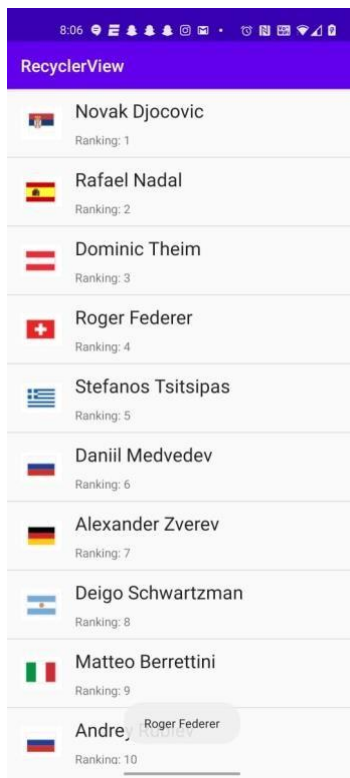
Output:



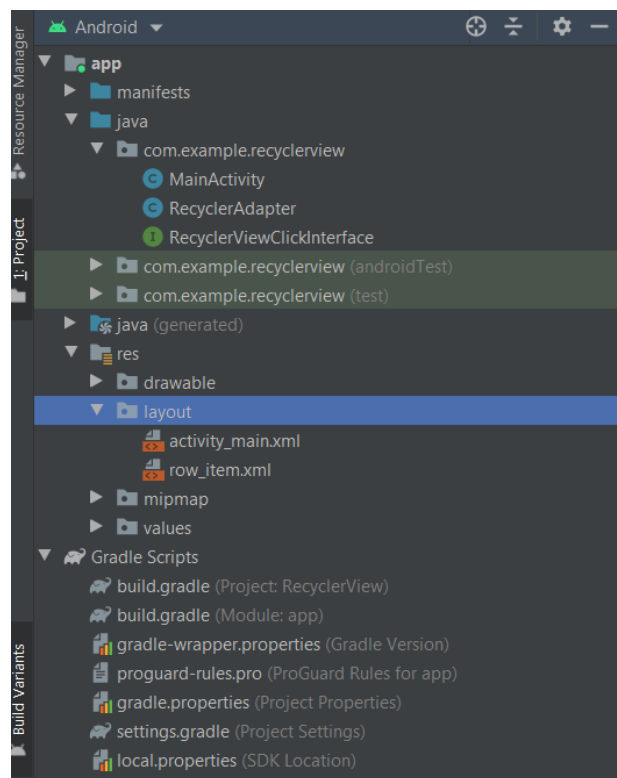
Onclick Listener implemented using interface:



Toast of the item clicked:



Project Directory:



RecyclerViewAdapter.java

```
22 public class RecyclerViewAdapter extends RecyclerView.Adapter<RecyclerView.ViewHolder>{
23     private static final String TAG = "RecyclerViewAdapter";
24     List<String> playerList;
25     private RecyclerViewClickInterface recyclerViewClickInterface;
26
27
28
29     public RecyclerViewAdapter(List<String> playerList, RecyclerViewClickInterface recyclerViewClickInterface) {
30         this.playerList = playerList;
31         this.recyclerViewClickInterface = recyclerViewClickInterface;
32     }
33
34
35     @NonNull
36     @Override
37     public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
38
39         LayoutInflater inflater = LayoutInflater.from(parent.getContext());
40         View view= inflater.inflate(R.layout.row_item, parent, attachToRoot: false);
41         ViewHolder viewHolder = new ViewHolder(view);
42         return viewHolder;
43     }
44
45
46     @Override
47     public void onBindViewHolder(@NonNull ViewHolder holder, int position) {
48         holder.rowCountTextView.setText("Ranking: " + String.valueOf(position+1));
49         holder.textView.setText(playerList.get(position));
50         String imageId = MainActivity.imageViewList.get(position);
51         imageId = imageId.replace( target: ".PNG", replacement: "");
52         //String imageId = "image_1";
53         Context context = holder.imageView.getContext();
54         int id = context.getResources().getIdentifier(imageId, "drawable", context.getPackageName());
```

```

54         int id = context.getResources().getIdentifier(imageId, "drawable", context.getPackageName());
55         Drawable drawable = ResourcesCompat.getDrawable(context.getResources(), id, theme: null);
56         holder.imageView.setImageDrawable(drawable);
57     }
58 }
59
60 @Override
61 public int getItemCount() {
62     return playerList.size();
63 }
64
65 class ViewHolder extends RecyclerView.ViewHolder{
66     ImageView imageView;
67     TextView textView, rowCountTextView;
68     public ViewHolder(@NonNull View itemView) {
69         super(itemView);
70         imageView = itemView.findViewById(R.id.imageView);
71         textView = itemView.findViewById(R.id.textView);
72         rowCountTextView = itemView.findViewById(R.id.rowCountTextView);
73
74         itemView.setOnClickListener(new View.OnClickListener() {
75             @Override
76             public void onClick(View view) {
77                 recyclerViewClickInterface.onItemClick(getAdapterPosition());
78             }
79         });
80     }
81 }
82
83 }
84 }

```

MainActivity.java

```

1 package com.example.recyclerview;
2
3 import ...
4
24
25 public class MainActivity extends AppCompatActivity implements RecyclerViewClickInterface{
26     RecyclerView recyclerView;
27     RecyclerViewAdapter recyclerViewAdapter;
28
29     List<String> playerList;
30     public static List<String> imageViewList;
31     SwipeRefreshLayout swipeRefreshLayout;
32     @Override
33     protected void onCreate(Bundle savedInstanceState) {
34         super.onCreate(savedInstanceState);
35         setContentView(R.layout.activity_main);
36         playerList = new ArrayList<>();
37         imageViewList = new ArrayList<>();
38
39
40         recyclerView = findViewById(R.id.recyclerview);
41         recyclerViewAdapter = new RecyclerViewAdapter(playerList, recyclerViewClickInterface: this);
42         recyclerView.setLayoutManager(new LinearLayoutManager(context: this));
43         recyclerView.setAdapter(recyclerViewAdapter);
44         DividerItemDecoration dividerItemDecoration = new DividerItemDecoration(context: this, DividerItemDecoration.VERTICAL);
45         recyclerView.addItemDecoration(dividerItemDecoration);
46
47         playerList.add("Novak Djokovic");
48         playerList.add("Rafael Nadal");
49         playerList.add("Dominic Thiem");
50         playerList.add("Roger Federer");
51         playerList.add("Stefanos Tsitsipas");
52         playerList.add("Daniil Medvedev");
53         playerList.add("Alexander Zverev");

```

```

54     playerList.add("Deigo Schwartzman");
55     playerList.add("Matteo Berrettini");
56     playerList.add("Andrey Rublev");
57     playerList.add("Gael Monfils");
58     playerList.add("Denis Shapovalov");
59     playerList.add("Roberto Bautista Agut");
60     playerList.add("David Goffin");
61     playerList.add("Pablo Carreno Busta");
62     playerList.add("Fabio Fognini");
63     playerList.add("Karen Khachanov");
64     playerList.add("Stanislas Wawrinka");
65     playerList.add("Grigor Dimitrov");
66     playerList.add("Cristian Garin");
67
68     imageViewList.add("image_1.PNG");
69     imageViewList.add("image_2.PNG");
70     imageViewList.add("image_3.PNG");
71     imageViewList.add("image_4.PNG");
72     imageViewList.add("image_5.PNG");
73     imageViewList.add("image_6.PNG");
74     imageViewList.add("image_7.PNG");
75     imageViewList.add("image_8.PNG");
76     imageViewList.add("image_9.PNG");
77     imageViewList.add("image_10.PNG");
78     imageViewList.add("image_11.PNG");
79     imageViewList.add("image_12.PNG");
80     imageViewList.add("image_13.PNG");
81     imageViewList.add("image_14.PNG");
82     imageViewList.add("image_15.PNG");
83     imageViewList.add("image_16.PNG");
84     imageViewList.add("image_17.PNG");
85     imageViewList.add("image_18.PNG");

```

```

85     imageViewList.add("image_18.PNG");
86     imageViewList.add("image_19.PNG");
87     imageViewList.add("image_20.PNG");
88
89
90
91
92     swipeRefreshLayout = findViewById(R.id.swipeRefreshLayout);
93     swipeRefreshLayout.setOnRefreshListener(new SwipeRefreshLayout.OnRefreshListener() {
94         @Override
95         public void onRefresh() {
96             playerList.add("Milos Raonic");
97             playerList.add("Felix Auger-Aliassime");
98             playerList.add("John Isner");
99             playerList.add("Dusan Lajovic");
100             playerList.add("Casper Ruud");
101
102             recyclerView.notifyDataSetChanged();
103             swipeRefreshLayout.setRefreshing(false);
104         }
105     });
106
107
108     ItemTouchHelper itemTouchHelper = new ItemTouchHelper(simpleCallback);
109     itemTouchHelper.attachToRecyclerView(recyclerView);
110 }

```

row_item.xml

