Greetings App

package com.mastercoding.greetingsapp;  
  
import androidx.appcompat.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
import android.widget.Toast;  
  
public class MainActivity extends AppCompatActivity {  
 EditText editText;  
 Button myBtn;  
 TextView title;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 editText = findViewById(R.id.*edittext*);  
 myBtn = findViewById(R.id.*btn*);  
 title = findViewById(R.id.*title*);  
  
  
 myBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
  
 String inputName = editText.getText().toString();  
  
 Toast.*makeText*(  
 MainActivity.this,  
 "Welcome "+inputName +" to our App",  
 Toast.*LENGTH\_LONG*).show();  
 }  
 });  
 }  
}

Counter App

package com.example.counterapp;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
  
public class MainActivity extends AppCompatActivity {  
  
 TextView title, count;  
 Button btn;  
 int counter = 0;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 title = findViewById(R.id.*Title*);  
 count = findViewById(R.id.*Count*);  
 btn = findViewById(R.id.*Btn*);  
  
 btn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 count.setText(""+increase());  
 }  
 });  
 }  
  
 public int increase(){  
 return ++counter;  
 }  
  
}

Copy paste img to drawable folder, the name musnt start with capital letter or number, and no space.

Converter App

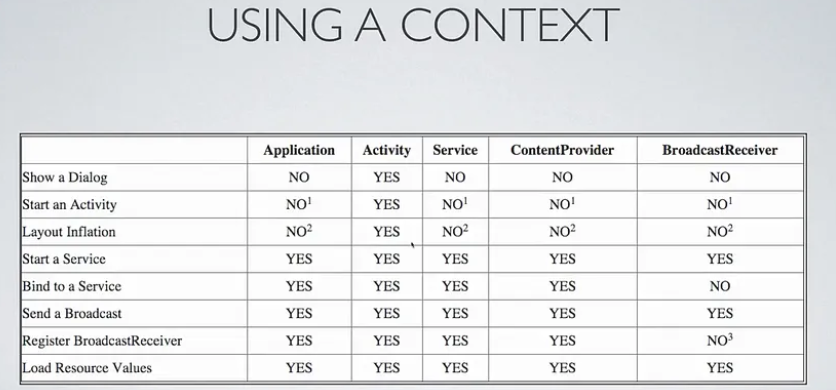
public class MainActivity extends AppCompatActivity {  
  
 TextView degreeK;  
 EditText degreeC;  
 Button btn;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 degreeK = findViewById(R.id.*Kelvin*);  
 degreeC = findViewById(R.id.*Celcius*);  
 btn = findViewById(R.id.*Btn*);  
  
 btn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String inputText = degreeC.getText().toString();  
 float inputCelcius = Float.*parseFloat*(inputText);  
 degreeK.setText(""+convert(inputCelcius));  
 }  
 });  
  
 }  
  
 public float convert(float celcius){  
 float kelvin = (float) (celcius + 273);  
 return kelvin;  
 }  
  
}

Context

<https://www.codingninjas.com/studio/library/context-in-android>

A Context gives you access to information about the current state of your Application.

 allows us to access application specific classes and resources.



Explicit & Implicit Intent

package com.mastercoding.layoutapp;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.net.Uri;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.Toast;  
  
public class MainActivity extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 // Intents: facilitates communication bet. different components of an app,  
 // as well as bet. different applications.  
  
 // types of intents:  
 // 1- Explicit Intents  
 Button btn = findViewById(R.id.*btn*);  
 btn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 goToSecondActivity();  
 }  
 });  
  
  
 // 2- Implicit Intents  
 Button btn2 = findViewById(R.id.*openBrowser*);  
 btn2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 openWebPage();  
 }  
 });  
 }  
  
  
 public void goToSecondActivity(){  
 Intent intent = new Intent(this, SecondActivity.class );  
 startActivity(intent);  
 }  
  
  
 public void openWebPage(){  
 Uri webpage = Uri.*parse*("https://www.google.com");  
  
 Intent intent = new Intent(Intent.*ACTION\_VIEW*, webpage);  
  
 startActivity(intent);  
  
 }  
  
}

Lucky Number App + share

Main Activity

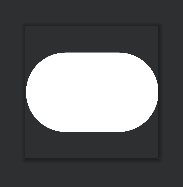
package com.example.luckynumber;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
  
import …  
  
public class MainActivity extends AppCompatActivity {  
  
 EditText editText;  
 Button btn;  
 TextView txt;  
  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 getMenuInflater().inflate(R.menu.*menu*, menu);  
 return true;  
 }  
  
 @Override  
 public boolean onOptionsItemSelected(@NonNull MenuItem item) {  
 int itemId = item.getItemId();  
  
 if(itemId == R.id.*Search*){  
 Toast.*makeText*(this, "search", Toast.*LENGTH\_SHORT*).show();  
 } else if (itemId == R.id.*Home*) {  
 Toast.*makeText*(this, "Home", Toast.*LENGTH\_SHORT*).show();  
 }  
  
 return super.onOptionsItemSelected(item);  
 }  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 txt = findViewById(R.id.*textView*);  
 editText = findViewById(R.id.*name*);  
 btn = findViewById(R.id.*btn*);  
  
 btn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 String name = editText.getText().toString();  
  
 Intent i = new Intent(getApplicationContext(), SecondActivity.class );  
  
 i.putExtra("name", name);  
  
 startActivity(i);  
 }  
 });  
 }  
   
}

Second Activity

package com.example.luckynumber;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import …  
  
import java.util.Random;  
  
public class SecondActivity extends AppCompatActivity {  
  
 TextView welcomeText, luckyNumberTxt;  
 Button share\_btn;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_second*);  
  
 welcomeText = findViewById(R.id.*TextView2*);  
 luckyNumberTxt = findViewById(R.id.*luckynumber*);  
 share\_btn = findViewById(R.id.*shareBtn*);  
  
 Intent a = getIntent();  
 String userName = a.getStringExtra("name");  
  
 int randomNum = generateRand();  
 luckyNumberTxt.setText(""+ randomNum);  
  
 share\_btn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 shareData(userName, randomNum);  
 }  
 });  
  
 }  
  
 public int generateRand(){  
 Random random = new Random();  
 int upperLimit = 1000;  
  
 int randGenerated = random.nextInt(upperLimit);  
 return randGenerated;  
  
 }  
  
 public void shareData(String userName, int randNum){  
 Intent i = new Intent(Intent.*ACTION\_SEND*);  
 i.setType("text/plain");  
  
 i.putExtra(Intent.*EXTRA\_SUBJECT*, userName + "lucky today!");  
 i.putExtra(Intent.*EXTRA\_TEXT*, "Your lucky number is " +randNum);  
  
 startActivity(Intent.*createChooser*(i, "Choose a platform!"));  
 }  
  
}

Drawable

<?xml version="1.0" encoding="utf-8"?>  
<shape  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:shape="rectangle">  
  
 <solid  
 android:color="@color/white"/>  
 <corners  
 android:radius="200dp"/>  
 <size  
 android:height="60dp"  
 android:width="100dp"/>  
  
</shape>



Use in layout as background : @drawable/…

Mipmap

New -> Image Asset -> Choose the img for icon, don’t change the name

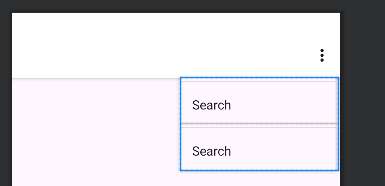
Theme

<resources xmlns:tools="http://schemas.android.com/tools">  
 <!-- Base application theme. -->  
 <style name="Base.Theme.LuckyNumber" parent="Theme.Material3.DayNight">  
 <!-- Customize your light theme here. -->  
 <!-- <item name="colorPrimary">@color/my\_light\_primary</item> -->  
  
 <item name="colorPrimary">@color/white</item>  
 </style>  
  
 <style name="Theme.LuckyNumber" parent="Base.Theme.LuckyNumber" />  
</resources>

Add an item, choose by name

Menu

<?xml version="1.0" encoding="utf-8"?>  
<menu xmlns:android="http://schemas.android.com/apk/res/android">  
 <item android:id="@+id/Search"  
 android:title="Search"  
 android:icon="@drawable/background1" />  
 <item android:id="@+id/Home"  
 android:title="Search"  
 android:icon="@drawable/background1" />  
</menu>



@Override  
public boolean onCreateOptionsMenu(Menu menu) {  
 getMenuInflater().inflate(R.menu.*menu*, menu);  
 return true;  
}  
  
@Override  
public boolean onOptionsItemSelected(@NonNull MenuItem item) {  
 int itemId = item.getItemId();  
  
 if(itemId == R.id.*Search*){  
 Toast.*makeText*(this, "search", Toast.*LENGTH\_SHORT*).show();  
 } else if (itemId == R.id.*Home*) {  
 Toast.*makeText*(this, "Home", Toast.*LENGTH\_SHORT*).show();  
 }  
  
 return super.onOptionsItemSelected(item);  
}

In theme, remove style no action bar to inflate the menus

In this line, **super.onOptionsItemSelected(item)** calls the **onOptionsItemSelected** method of the superclass, allowing the parent class to handle the menu item selection logic.

Using **super** in this way is common when you want to extend the behavior of a method provided by the superclass while still benefiting from the functionality implemented in the parent class.

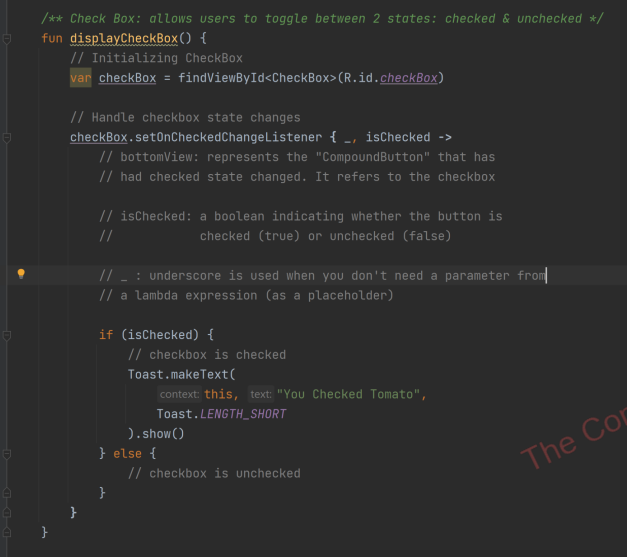
Fonts

Res -> New -> Android Resource Directory -> Choose font -> add the file

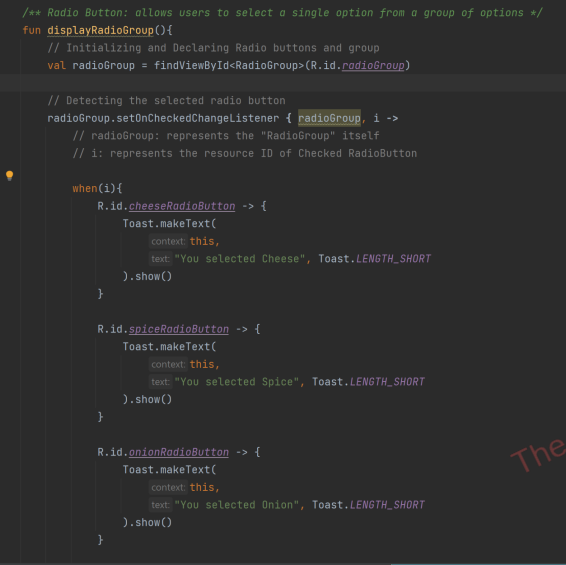
Use as font family in layout

**Widgets**

Checkbox



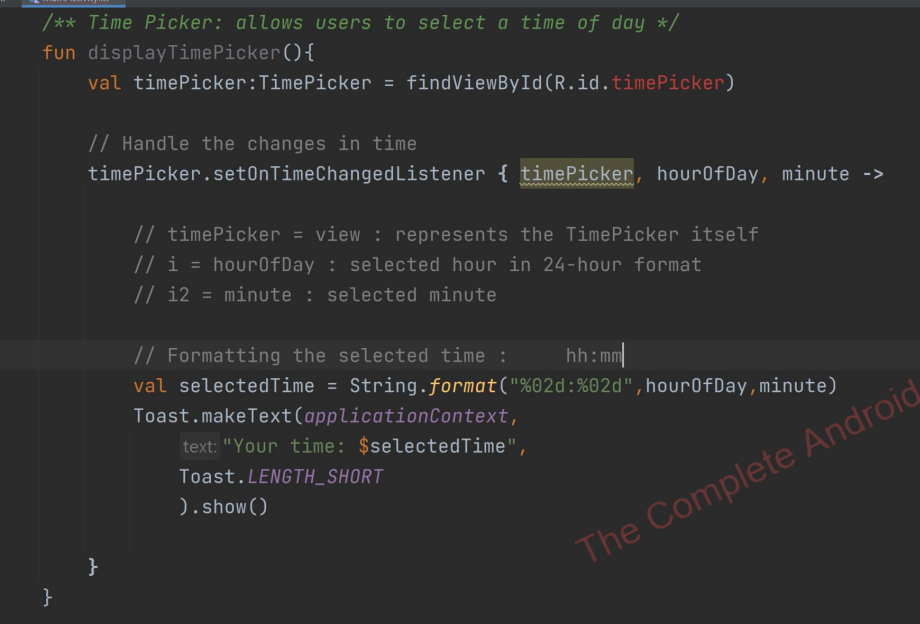
Radiogroup & radiobutton



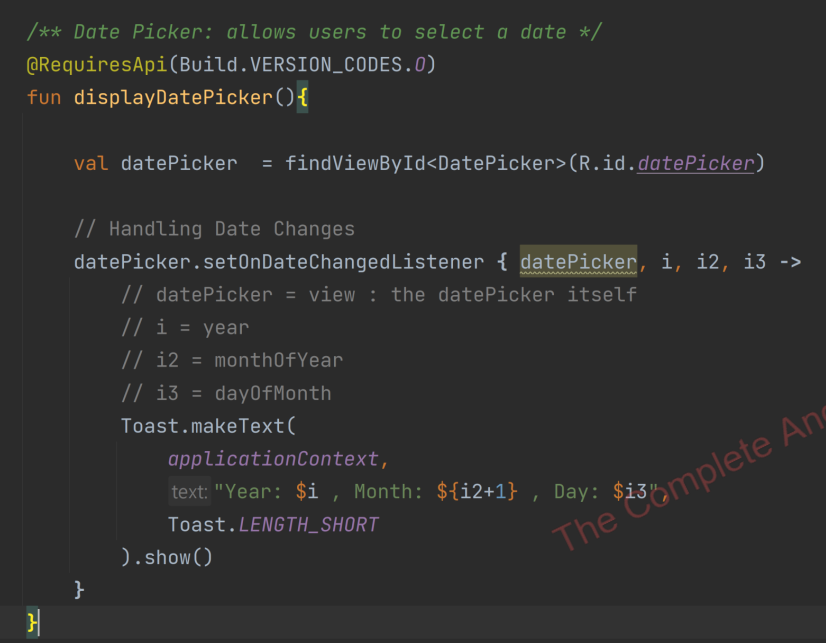
Spinner



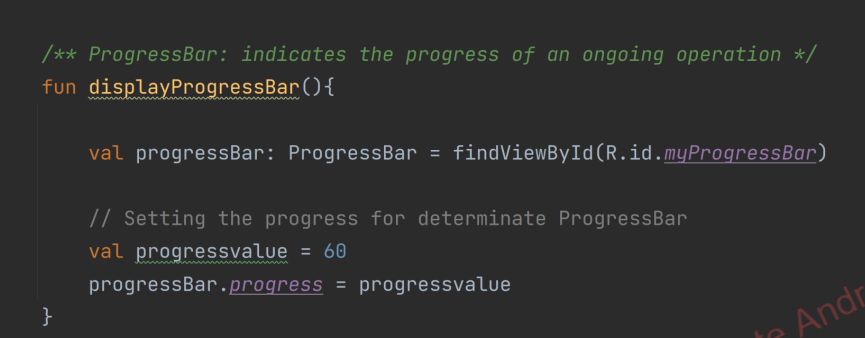
Time Picker



Date Picker



Progress bar

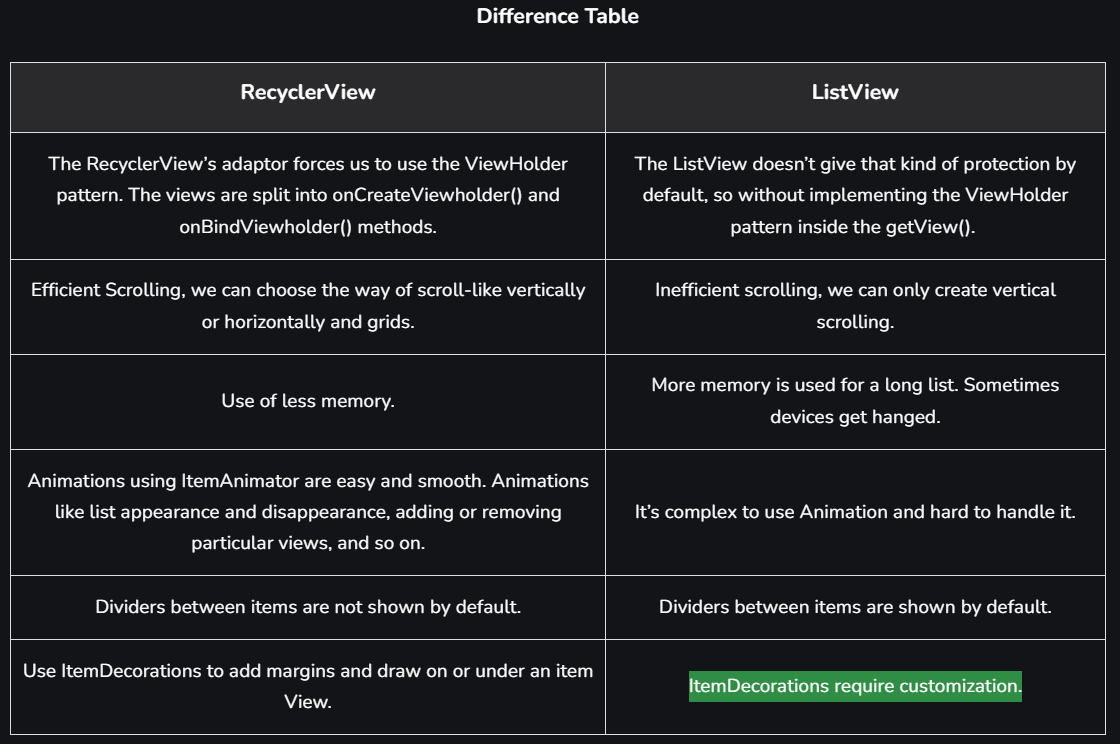


French Teacher App Media Player

package com.example.frenchteacher;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.media.MediaPlayer;  
import android.os.Bundle;  
import android.provider.MediaStore;  
import android.view.View;  
import android.widget.Button;  
  
public class MainActivity extends AppCompatActivity implements View.OnClickListener{  
  
 Button song1btn, song2btn, song3btn;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 song1btn = findViewById(R.id.*button*);  
 song2btn = findViewById(R.id.*button2*);  
 song3btn = findViewById(R.id.*button3*);  
  
 song1btn.setOnClickListener(this);  
 song2btn.setOnClickListener(this);  
 song3btn.setOnClickListener(this);  
 }  
  
 @Override  
 public void onClick(View view) {  
 int clickedBtn = view.getId();  
  
 if (clickedBtn == R.id.*button*){  
 PlaySound(R.raw.*song1*);  
 } else if (clickedBtn == R.id.*button2*) {  
 PlaySound(R.raw.*song2*);  
 }else if (clickedBtn == R.id.*button3*) {  
 PlaySound(R.raw.*song3*);  
 }  
 }  
  
 public void PlaySound(int id){  
 MediaPlayer mediaPlayer = MediaPlayer.*create*(getApplicationContext(), id);  
 mediaPlayer.start();  
 }  
  
}

put the mp3 in raw folder inside res

Implements View.OnClickListener interface



CustomAdapter

package com.mastercoding.adaptersapp;  
  
import …  
  
public class MyCustomAdapter extends BaseAdapter {  
 private Context context;  
 private String[] items; // Data Source  
  
 public MyCustomAdapter(Context context, String[] items) {  
 this.context = context;  
 this.items = items;  
 }  
  
 @Override  
 public int getCount() {  
 return items.length; // Returns the number of items in your data source  
 }  
  
 @Override  
 public Object getItem(int position) {  
 return items[position]; // Returns the data item at the given position  
 }  
  
 @Override  
 public long getItemId(int position) {  
 return position; // Returns a unique Identifier for the item at the given position  
 }  
  
 @Override  
 public View getView(int position, View convertView, ViewGroup parent) {  
 ViewHolder holder;  
  
 if (convertView == null){  
 // convertView: is a recycled View that you can reuse to  
 // improve the performance of your list.  
 convertView= LayoutInflater.*from*(context)  
 .inflate(R.layout.*my\_list\_item*, parent, false);  
  
 holder = new ViewHolder();  
 holder.textView = convertView.findViewById(R.id.*text1*);  
 convertView.setTag(holder);  
  
  
 }else{  
 // Reusing the View (that's recycled)  
 holder = (ViewHolder) convertView.getTag();  
 }  
  
 // Set the data to the view  
 holder.textView.setText(items[position]);  
  
  
  
 // Binding data to views within the convertView  
 return convertView; // Displays the data at a position in the data set  
 }  
  
 static class ViewHolder{  
 // Holds references to the views within an item layout  
 TextView textView;  
 }  
}

Main Activity

package com.mastercoding.adaptersapp;  
import …  
  
public class MainActivity extends AppCompatActivity {  
 ListView listview;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 // 1- AdapterView: ListView  
 listview = findViewById(R.id.*listview*);  
  
 // 2- Data Source: String Array  
 String[] countries = {"USA", "Germany", "Saudi Arabia","France"};  
  
  
 // 3- Adapter: acts as a bridge between the  
 // 'data source' and the 'AdapterView'  
 MyCustomAdapter adapter = new MyCustomAdapter(this, countries);  
  
 // Link Listview with the Adapter  
 listview.setAdapter(adapter);  
  
 }  
}

Planet ListView with Custom Adapter

CustomAdapter

package com.mastercoding.planetsapp;  
  
import ….  
  
public class MyCustomAdapter extends ArrayAdapter<Planet> {  
  
 // Using Custom Layouts --> MyCustomAdapter  
 // Using Custom Objects --> extends ArrayAdapter<Planet>  
  
 private ArrayList<Planet> planetsArrayList;  
 Context context;  
  
 public MyCustomAdapter( ArrayList<Planet> planetsArrayList, Context context) {  
 super(context, R.layout.item\_list\_layout, planetsArrayList);  
 this.planetsArrayList = planetsArrayList;  
 this.context = context;  
 }  
  
 // View Holder Class: used to cache references to the views within  
 // an item layout, so that they don't need to be  
 // repeatedly looked up during scrolling  
  
 private static class MyViewHolder{  
 TextView planetName;  
 TextView moonCount;  
 ImageView planetImg;  
  
 }  
  
 // getView(): used to create and return a view for a  
 // specific item in the list.  
  
 @NonNull  
 @Override  
 public View getView(int position, @Nullable View convertView, @NonNull ViewGroup parent) {  
 // 1- Get the planet object for the current position  
 Planet planets = getItem(position);  
  
 // 2- Inflate Layout:  
 MyViewHolder myViewHolder;  
 final View result;

if (convertView == null){  
 myViewHolder = new MyViewHolder();  
 LayoutInflater inflater = LayoutInflater.from(getContext());  
 convertView = inflater.inflate(  
 R.layout.item\_list\_layout,  
 parent,  
 false  
 );  
  
 // Finding Views:  
 myViewHolder.planetName = (TextView) convertView.findViewById(R.id.planet\_name);  
 myViewHolder.moonCount = (TextView) convertView.findViewById(R.id.moon\_count\_text);  
 myViewHolder.planetImg = (ImageView) convertView.findViewById(R.id.imageView);  
  
 result = convertView;  
  
 convertView.setTag(myViewHolder);  
 }else{  
 // the view is recycled  
 myViewHolder = (MyViewHolder) convertView.getTag();  
 result = convertView;  
 }  
  
 // Getting the data from model class (Planet)  
 myViewHolder.planetName.setText(planets.getPlanetName());  
 myViewHolder.moonCount.setText(planets.getMoonCount());  
 myViewHolder.planetImg.setImageResource(planets.getPlanetImage());  
  
 return result;  
   
 }  
}

Main Activity

package com.mastercoding.planetsapp;  
  
import …  
  
public class MainActivity extends AppCompatActivity {  
 ListView listView;  
 ArrayList<Planet> planetsArrayList;  
 private static MyCustomAdapter adapter;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 // 1- AdapterView: a ListView  
 listView = findViewById(R.id.listview);  
  
 // 2- Data Source: ArrayList<Planet>  
 planetsArrayList = new ArrayList<>();  
  
 Planet planet3 = new Planet("Earth","1 Moon",R.drawable.earth);  
 Planet planet1 = new Planet("Mercury","0 Moons",R.drawable.mercury);  
 Planet planet2 = new Planet("Venus", "0 Moons", R.drawable.venus);  
  
 Planet planet4 = new Planet("Mars", "2 Moons", R.drawable.mars);  
 Planet planet5 = new Planet("Jupiter", "79 Moons", R.drawable.jupiter);  
 Planet planet6 = new Planet("Saturn", "83 Moons", R.drawable.saturn);  
  
 Planet planet7 = new Planet("Uranus", "27 Moons", R.drawable.uranus);  
 Planet planet8 = new Planet("Neptune", "14 Moons", R.drawable.neptune);  
  
 planetsArrayList.add(planet1);  
 planetsArrayList.add(planet2);  
 planetsArrayList.add(planet3);  
 planetsArrayList.add(planet4);  
 planetsArrayList.add(planet5);  
 planetsArrayList.add(planet6);  
 planetsArrayList.add(planet7);  
 planetsArrayList.add(planet8);  
  
  
 // Adapter:  
 adapter = new MyCustomAdapter(planetsArrayList, getApplicationContext());  
  
 listView.setAdapter(adapter);  
  
  
 // Handling Click Events  
  
 listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
 @Override  
 public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
 Toast.makeText(  
 MainActivity.this,  
 "Planet Name: " +adapter.getItem(position).getPlanetName(),  
 Toast.LENGTH\_SHORT).show();  
 }  
 });  
  
  
  
 }  
}

Volume Calculator App

CustomAdapter

package com.example.calculatorshape;  
import …

public class MyCustomAdapter extends ArrayAdapter<Shape> {  
  
 private ArrayList<Shape> myShapeList;  
 Context context;  
  
 public MyCustomAdapter (ArrayList<Shape> myShapeList, Context context) {  
 super(context, R.layout.*grid\_layout*, myShapeList);  
 this.myShapeList = myShapeList;  
 this.context = context;  
 }  
  
 private static class MyViewHolder{  
 TextView shapeName;  
 ImageView shapeImg;  
 }  
  
 @NonNull  
 @Override  
 public View getView(int position, @Nullable View convertView, @NonNull ViewGroup parent) {  
  
 Shape shape = getItem(position);  
  
 MyViewHolder myViewHolder;  
  
  
 if (convertView == null){  
  
 myViewHolder = new MyViewHolder();  
 LayoutInflater inflater = LayoutInflater.*from*(getContext());  
  
 convertView = inflater.inflate(  
 R.layout.*grid\_layout*,  
 parent,  
 false  
 );  
  
 myViewHolder.shapeName = (TextView) convertView.findViewById(R.id.*shapeName*);  
 myViewHolder.shapeImg = (ImageView) convertView.findViewById(R.id.*shapeImg*);  
  
 convertView.setTag(myViewHolder);  
  
 }else{  
 myViewHolder = (MyViewHolder) convertView.getTag();  
 }  
  
 myViewHolder.shapeName.setText(shape.getShapeName());  
 myViewHolder.shapeImg.setImageResource(shape.getShapeImg());  
 return convertView;  
 }  
}

Main Activity

package com.example.calculatorshape;  
import …  
  
public class MainActivity extends AppCompatActivity {  
  
 GridView gridView;  
 ArrayList<Shape> shapeArrayList;  
 MyCustomAdapter adapter;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 gridView = findViewById(R.id.*gridView*);  
 shapeArrayList = new ArrayList<>();  
  
 Shape s1 = new Shape(R.drawable.*shape*, "Bola");  
 Shape s2 = new Shape(R.drawable.*shape*, "Balok");  
 Shape s3 = new Shape(R.drawable.*shape*, "Kerucut");  
 Shape s4 = new Shape(R.drawable.*shape*, "Silinder");  
 Shape s5 = new Shape(R.drawable.*shape*, "Kontol");  
 Shape s6 = new Shape(R.drawable.*shape*, "Memek");  
  
 shapeArrayList.add(s1);  
 shapeArrayList.add(s2);  
 shapeArrayList.add(s3);  
 shapeArrayList.add(s4);  
 shapeArrayList.add(s5);  
 shapeArrayList.add(s6);  
  
 adapter = new MyCustomAdapter(shapeArrayList, getApplicationContext());  
  
 gridView.setAdapter(adapter);  
 gridView.setNumColumns(2);  
  
 gridView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
 @Override  
 public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
 Shape selectedShape = shapeArrayList.get(position);  
  
 // Create an Intent to start the calculation activity  
 Intent intent = new Intent(getApplicationContext(), CalculatorActivity.class);  
  
 // Pass information about the selected shape to the next activity  
 intent.putExtra("SHAPE\_NAME", selectedShape.getShapeName());  
  
 // Start the activity  
 startActivity(intent);  
 }  
 });  
  
 }  
}

Calculator Activity

package com.example.calculatorshape;  
  
import ...  
  
public class CalculatorActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_calculator*);  
  
 // Retrieve information about the selected shape  
 Intent intent = getIntent();  
 String shapeName = intent.getStringExtra("SHAPE\_NAME");  
  
  
 // Use the information to perform calculations or display relevant content  
 // You can switch on shapeName and perform specific calculations or show UI accordingly  
 // For simplicity, let's assume you have a TextView to display the selected shape name  
 TextView shapeNameTextView = findViewById(R.id.*shapeGanti*);  
 shapeNameTextView.setText(shapeName);  
  
 }  
}

gridView.setOnItemClickListener(new AdapterView.OnItemClickListener() {

@Override

public void onItemClick(AdapterView<?> parent, View view, int position, long id) {

Intent intent;

switch (position) {

case 0: // Bola

intent = new Intent(getApplicationContext(), Bola.class);

break;

case 1: // Balok

intent = new Intent(getApplicationContext(), Balok.class);

break;

case 2: // Kerucut

intent = new Intent(getApplicationContext(), Kerucut.class);

break;

case 3: // Silinder

intent = new Intent(getApplicationContext(), Silinder.class);

break;

case 4: // Kontol

intent = new Intent(getApplicationContext(), Kontol.class);

break;

case 5: // Memek

intent = new Intent(getApplicationContext(), Memek.class);

break;

default:

// Handle unexpected position

return;

}

startActivity(intent);

}

});

Recycler Views

package com.example.recyclerview;  
  
import …  
  
public class CustomActivity extends RecyclerView.Adapter<CustomActivity.MyViewHolder> {  
  
 private List<item> itemList;  
  
 public ItemClickListener clickListener;  
  
 public void setClickListener(ItemClickListener myListener){  
 this.clickListener = myListener;  
 }  
  
 public CustomActivity(List<item> itemList) {  
 this.itemList = itemList;  
 }  
  
 @NonNull  
 @Override  
 public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  
  
 View itemView = LayoutInflater.*from*(parent.getContext()).inflate(  
 R.layout.*item\_layout*,  
 parent,  
 false  
 );  
  
 return new MyViewHolder(itemView) ;  
  
 }  
  
 @Override  
 public void onBindViewHolder(@NonNull MyViewHolder holder, int position) {  
  
 item item = itemList.get(position);  
 holder.title.setText(item.getItemTitle());  
 holder.imageView.setImageResource(item.getItemImg());  
 holder.description.setText(item.getItemDesc());  
  
 }  
  
 @Override  
 public int getItemCount() {  
 return itemList.size();  
 }  
  
 public class MyViewHolder extends RecyclerView.ViewHolder implements View.OnClickListener{  
  
 ImageView imageView;  
 TextView title;  
 TextView description;  
  
 public MyViewHolder(@NonNull View itemView) {  
 super(itemView);  
  
 imageView = itemView.findViewById(R.id.*imageView*);  
 title = itemView.findViewById(R.id.*textView*);  
 description = itemView.findViewById(R.id.*desc*);  
  
 itemView.setOnClickListener(this);  
  
 }  
  
 @Override  
 public void onClick(View v) {  
 if (clickListener != null ){  
 clickListener.onClick(v,getAdapterPosition());  
 }  
 }  
 }  
}

MyViewHolder need to be made first, so that it can be passed to the adapter, before creating any constructor or method.

Main Activity

package com.example.recyclerview;  
  
import …  
  
public class MainActivity extends AppCompatActivity implements ItemClickListener{  
 //Adapter View  
 RecyclerView recyclerView;  
  
 //Data source  
 List<item> itemList;  
  
 //Adapter  
 CustomActivity adapter;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 recyclerView = findViewById(R.id.*recyclerView*);  
  
 itemList = new ArrayList<>();  
  
 item item1 = new item(R.drawable.*icon*, "Title1", "Description Lorem Ipsum");  
 item item2 = new item(R.drawable.*icon*, "Title2", "Description Lorem Ipsum");  
 item item3 = new item(R.drawable.*icon*, "Title3", "Description Lorem Ipsum");  
 item item4 = new item(R.drawable.*icon*, "Title4", "Description Lorem Ipsum");  
 item item5 = new item(R.drawable.*icon*, "Title5", "Description Lorem Ipsum");  
 item item6 = new item(R.drawable.*icon*, "Title6", "Description Lorem Ipsum");  
 item item7 = new item(R.drawable.*icon*, "Title7", "Description Lorem Ipsum");  
  
 itemList.add(item1);  
 itemList.add(item2);  
 itemList.add(item3);  
 itemList.add(item4);  
 itemList.add(item5);  
 itemList.add(item6);  
 itemList.add(item7);  
  
  
 LinearLayoutManager layoutManager = new LinearLayoutManager(this);  
 recyclerView.setLayoutManager(layoutManager);  
  
  
 adapter = new CustomActivity(itemList);  
 recyclerView.setAdapter(adapter);  
  
 adapter.setClickListener(this);  
 }  
  
 @Override  
 public void onClick(View v, int pos) {  
 Toast.*makeText*(this, ""+itemList.get(pos).getItemTitle(), Toast.*LENGTH\_SHORT*).show();  
 }  
}

ItemClickListener Interface

package com.example.recyclerview;  
  
import android.view.View;  
  
public interface ItemClickListener {  
 void onClick(View v,int pos);  
}

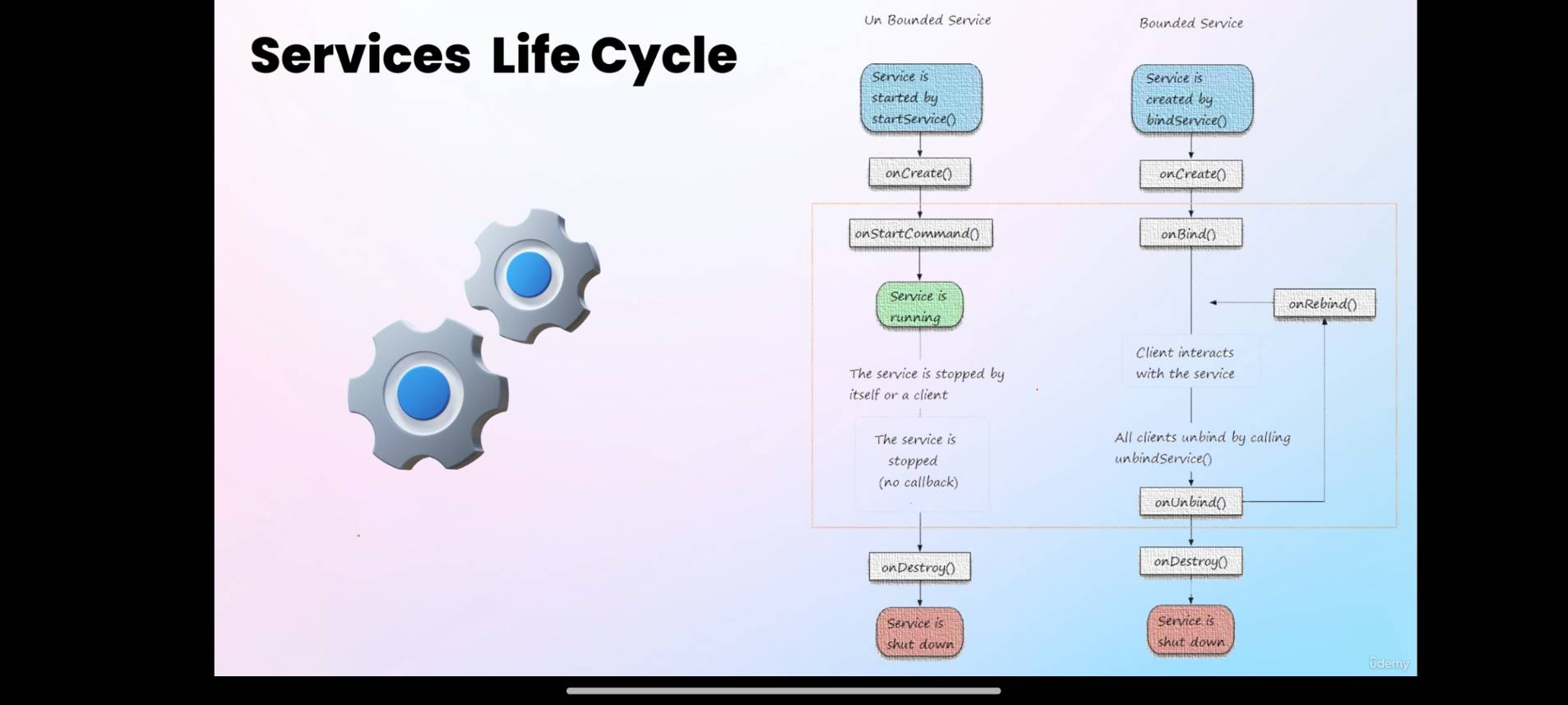
Card View Recycler View

package com.example.cardview;  
  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.recyclerview.widget.LinearLayoutManager;  
import androidx.recyclerview.widget.RecyclerView;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Toast;  
  
import java.util.ArrayList;  
import java.util.List;  
  
 public class MainActivity extends AppCompatActivity implements ClickListener{  
 //adapter view  
 private RecyclerView recyclerView;  
  
 //Data source  
 private List<Card> cardList;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 recyclerView = findViewById(R.id.*recyclerView*);  
 cardList = new ArrayList<>();  
  
 Card card1 = new Card(R.drawable.*icon*, "card1");  
 Card card2 = new Card(R.drawable.*icon*, "card2");  
 Card card3 = new Card(R.drawable.*icon*, "card3");  
 Card card4 = new Card(R.drawable.*icon*, "card4");  
 Card card5 = new Card(R.drawable.*icon*, "card5");  
  
 cardList.add(card1);  
 cardList.add(card2);  
 cardList.add(card3);  
 cardList.add(card4);  
 cardList.add(card5);  
  
 //adapter  
 MyAdapter adapter = new MyAdapter(cardList);  
  
 recyclerView.setAdapter(adapter);  
 recyclerView.setLayoutManager(new LinearLayoutManager(this));  
  
 adapter.setClickListener(this);  
  
 }  
  
 @Override  
 public void onCLick(View v, int position) {  
 Toast.*makeText*(this, "" + cardList.get(position).getCardText() , Toast.*LENGTH\_SHORT*).show();  
 }  
 }

package com.example.cardview;  
  
import…  
public class MyAdapter extends RecyclerView.Adapter<MyAdapter.MyViewHolder> {  
 private List<Card> cardList;  
  
 //add instance that receive listener event from main act  
 public ClickListener cardItemListener;  
  
 public void setClickListener(ClickListener myListener){  
 this.cardItemListener = myListener;  
 }  
  
  
  
 public MyAdapter(List<Card> cardList) {  
 this.cardList = cardList;  
 }  
  
 @NonNull  
 @Override  
 public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  
  
 View view = LayoutInflater.*from*(parent.getContext()).inflate(  
 R.layout.*cardview\_layout*,  
 parent,  
 false  
 );  
  
 return new MyViewHolder(view);  
  
 }  
  
 @Override  
 public void onBindViewHolder(@NonNull MyViewHolder holder, int position) {  
  
 Card card = cardList.get(position);  
 holder.textView.setText(card.getCardText());  
 holder.imageView.setImageResource(card.getCardImg());  
  
 }  
  
 @Override  
 public int getItemCount() {  
 return cardList.size();  
 }  
  
  
 public class MyViewHolder extends RecyclerView.ViewHolder implements View.OnClickListener{  
 TextView textView;  
 ImageView imageView;  
  
 public MyViewHolder(@NonNull View itemView) {  
 super(itemView);  
  
 textView = itemView.findViewById(R.id.*cardText*);  
 imageView = itemView.findViewById(R.id.*cardImg*);  
  
 itemView.setOnClickListener(this);  
 }  
  
 @Override  
 public void onClick(View v) {  
 if (cardItemListener != null){  
 cardItemListener.onCLick(v, getAdapterPosition());  
 }  
 }  
 }  
  
}

package com.example.cardview;  
  
import android.view.View;  
  
public interface ClickListener {  
 void onCLick (View v, int position);  
}

Service

CustomService

package com.example.service;  
  
import …  
public class CustomService extends Service {  
  
 private MediaPlayer player;  
  
 @Override  
 public int onStartCommand(Intent intent, int flags, int startId) {  
  
 player = MediaPlayer.*create*(this, Settings.System.*DEFAULT\_RINGTONE\_URI*);  
  
 player.setLooping(true);  
 player.start();  
  
 return *START\_STICKY*;  
 }  
  
 @Nullable  
 @Override  
 public IBinder onBind(Intent intent) {  
 return null;  
 }  
  
 @Override  
 public void onDestroy() {  
 super.onDestroy();  
  
 player.stop();  
 }  
  
}

Main Activity

package com.example.service;  
  
import …  
public class MainActivity extends AppCompatActivity {  
 Button start\_btn, stop\_btn;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 start\_btn = findViewById(R.id.*playBtn*);  
 stop\_btn = findViewById(R.id.*stopBtn*);  
  
 start\_btn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent serviceIntent = new Intent(getApplicationContext(), CustomService.class);  
 startService(serviceIntent);  
 }  
 });  
  
 stop\_btn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent serviceIntent = new Intent(getApplicationContext(), CustomService.class);  
 stopService(serviceIntent);  
 }  
 });  
  
  
 }  
}

Add this in android manifest

<activity  
 android:name=".MainActivity"  
 android:exported="true">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
  
 <service android:name=".CustomService" />  
  
 </application>  
  
</manifest>

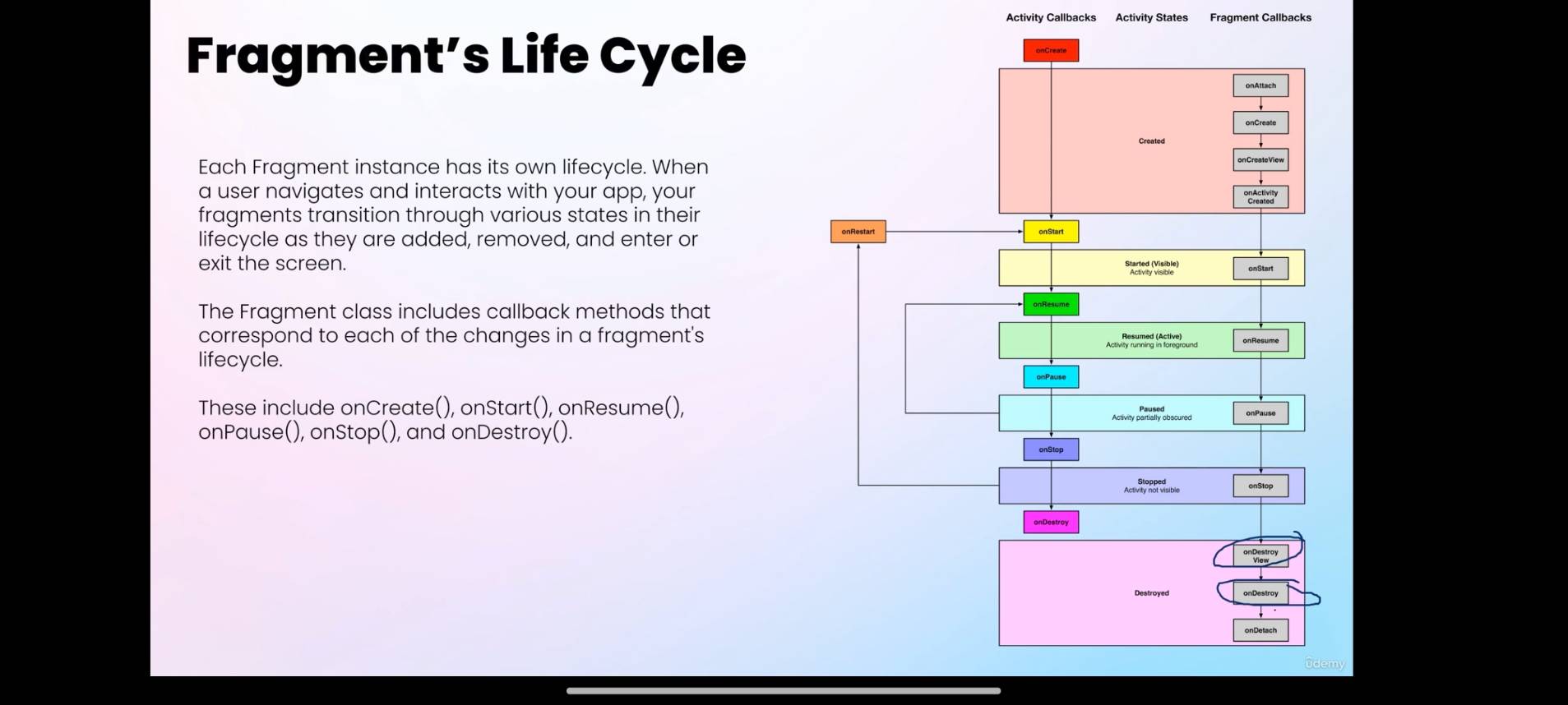
Airplane Mode

Airplane Receiver

package com.example.airplanemode;  
  
import ..  
  
public class AirplaneReceiver extends BroadcastReceiver {  
 @Override  
 public void onReceive(Context context, Intent intent) {  
  
 if(intent.getAction() != null && intent.getAction().equals(Intent.*ACTION\_AIRPLANE\_MODE\_CHANGED*)) {  
  
 boolean isAirplaneModeOn = intent.getBooleanExtra("state", false);  
  
 String msg = isAirplaneModeOn ? "Airplane Mode ON" : "Airplane Mode OFF";  
  
 Toast.*makeText*(context, ""+msg, Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
}

package com.example.airplanemode;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.IntentFilter;  
import android.os.Bundle;  
  
public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 IntentFilter intentFilter = new IntentFilter("android.intent.action.AIRPLANE\_MODE");  
 AirplaneReceiver br = new AirplaneReceiver();  
 registerReceiver(br, intentFilter);  
  
 }  
}

Fragment



First Fragment

package com.mastercoding.fragmentsapp;  
  
import …  
  
public class FirstFragment extends Fragment {  
  
 @Override  
 public void onAttach(@NonNull Context context) {  
 super.onAttach(context);  
 Toast.*makeText*(context,  
 "OnAttach() is Called", Toast.*LENGTH\_SHORT*).show();  
 }  
  
 @Override  
 public void onCreate(@Nullable Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 Toast.*makeText*(getActivity(),  
 "OnCreate() is Called", Toast.*LENGTH\_SHORT*).show();  
 }  
  
 @Override  
 public void onResume() {  
 super.onResume();  
 Toast.*makeText*(getActivity(),  
 "OnResume() is Called", Toast.*LENGTH\_SHORT*).show();  
 }  
  
  
 @Nullable  
 @Override  
 public View onCreateView(@NonNull LayoutInflater inflater,  
 @Nullable ViewGroup container,  
 @Nullable Bundle savedInstanceState) {  
  
 View view = inflater.inflate(R.layout.*fragment\_first*, container, false);  
  
 Button firstBtn = view.findViewById(R.id.*btn\_frag1*);  
 TextView text = view.findViewById(R.id.*text\_frag1*);  
  
 firstBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Toast.*makeText*(getActivity(),  
 "Welcome to the First Fragment",  
 Toast.*LENGTH\_SHORT*).show();  
 }  
 });  
  
 return view;  
 }  
}

Fragment 2

package com.mastercoding.fragmentsapp;  
  
import …  
public class SecondFragment extends Fragment {  
  
 @Nullable  
 @Override  
 public View onCreateView(@NonNull LayoutInflater inflater,  
 @Nullable ViewGroup container,  
 @Nullable Bundle savedInstanceState) {  
  
 View view = inflater.inflate(R.layout.*fragment\_second*,  
 container,  
 false);  
  
 Button btn = view.findViewById(R.id.*second\_btn*);  
 TextView txt = view.findViewById(R.id.*text\_frag2*);  
  
 btn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Toast.*makeText*(getActivity(),  
 "Good Bye From Second Fragment",  
 Toast.*LENGTH\_SHORT*).show();  
 }  
 });  
  
 return view;  
 }  
}

Main Activity

package com.mastercoding.fragmentsapp;  
  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.fragment.app.Fragment;  
import androidx.fragment.app.FragmentManager;  
import androidx.fragment.app.FragmentTransaction;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
  
  
public class MainActivity extends AppCompatActivity {  
  
 Button btn1,btn2;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 btn1 = findViewById(R.id.*btn1*);  
 btn2 = findViewById(R.id.*btn2*);  
  
 btn1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 loadFragment(new FirstFragment());  
 }  
 });  
 btn2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 loadFragment(new SecondFragment());  
 }  
 });  
 }  
  
 public void loadFragment(Fragment fragment){  
 // responsible for all runtime management of fragments  
 // including adding, removing, hiding, showing  
 // and navigating between fragments  
 FragmentManager fm = getSupportFragmentManager();  
  
 FragmentTransaction ft = fm.beginTransaction();  
  
 // Replace the framelayout with new fragment  
 ft.replace(R.id.*frameLayout*, fragment);  
 ft.commit();  
 }  
}

ViewPager

Add androidx dependencies put in the gradle

Fragment just keep the constructor and onCreateView

MyViewPagerAdapter

package com.mastercoding.viewpagerapp;  
  
import androidx.annotation.NonNull;  
import androidx.fragment.app.Fragment;  
import androidx.fragment.app.FragmentManager;  
import androidx.lifecycle.Lifecycle;  
import androidx.viewpager2.adapter.FragmentStateAdapter;  
  
import java.util.ArrayList;  
  
public class MyViewPagerAdapter extends FragmentStateAdapter {  
  
 private ArrayList<Fragment> fragmentList = new ArrayList<>();  
  
  
 public MyViewPagerAdapter(@NonNull FragmentManager fragmentManager, @NonNull Lifecycle lifecycle) {  
 super(fragmentManager, lifecycle);  
 }  
  
 @NonNull  
 @Override  
 public Fragment createFragment(int position) {  
 // responsible for creating and returning a  
 // fragment for a specific position within ViewPager2  
  
 return fragmentList.get(position);  
 }  
  
 @Override  
 public int getItemCount() {  
 return fragmentList.size();  
 }  
  
  
 public void addFragment(Fragment fragment){  
 fragmentList.add(fragment);  
 }  
}

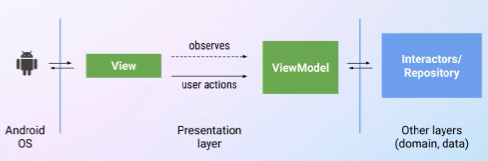
Main Activity

package com.mastercoding.viewpagerapp;

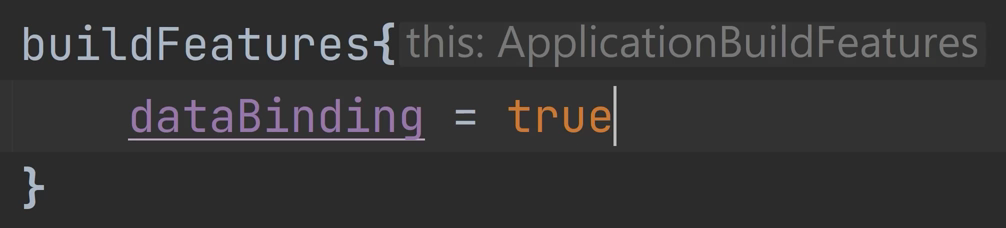
import …

public class MainActivity extends AppCompatActivity {  
 ViewPager2 viewpager;  
 MyViewPagerAdapter myAdapter;  
 TabLayout tabLayout;  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 tabLayout = findViewById(R.id.*tablayout*);  
  
  
  
 myAdapter = new MyViewPagerAdapter(  
 getSupportFragmentManager(),  
 getLifecycle()  
 );  
  
 // Adding Fragments to the list in the Adapter Class  
 myAdapter.addFragment(new Fragment1());  
 myAdapter.addFragment(new Fragment2());  
 myAdapter.addFragment(new Fragmen3());  
  
 // Set the orientation in ViewPager2  
 viewpager = findViewById(R.id.*viewPager2*);  
 viewpager.setOrientation(ViewPager2.*ORIENTATION\_HORIZONTAL*);  
  
 // Connecting the Adapter with ViewPager2  
 viewpager.setAdapter(myAdapter);  
  
 // Connecting TabLayout with ViewPager  
 new TabLayoutMediator(  
 tabLayout,  
 viewpager,  
 new TabLayoutMediator.TabConfigurationStrategy() {  
 @Override  
 public void onConfigureTab(@NonNull TabLayout.Tab tab, int position) {  
 tab.setText("Fragment " +(position +1));  
 }  
 }  
 ).attach();  
 }  
}

Data Binding



Enable data binding library, in gradle file



Use layout tag in xml and move the xmlns

<?xml version="1.0" encoding="utf-8"?>  
  
<layout  
 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">  
  
  
 <data>  
 <variable  
 name="person"  
 type="com.mastercoding.databindingapp.Person" />  
  
 <variable  
 name="clickHandler"  
 type="com.mastercoding.databindingapp.MyClickHandler" />  
  
 </data>  
  
  
 <androidx.constraintlayout.widget.ConstraintLayout  
  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@{person.name}"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <Button  
 android:onClick="@{clickHandler::onButton1Clicked}"  
 android:id="@+id/button3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="166dp"  
 android:text="Click Me!"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
  
 </androidx.constraintlayout.widget.ConstraintLayout>  
  
</layout>

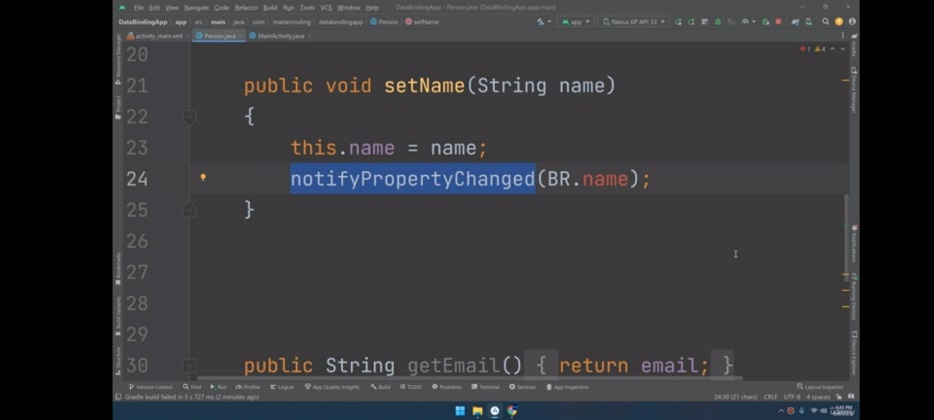
My Click Handler

package com.mastercoding.databindingapp;  
  
import   
public class MyClickHandler {  
  
 Context context;  
  
 public MyClickHandler(Context context) {  
 this.context = context;  
 }  
  
 public void onButton1Clicked(View view){  
 Toast.*makeText*(  
 context,  
 "First Button is Clicked",  
 Toast.*LENGTH\_SHORT*).show();  
 }

**Activity Main**

package com.mastercoding.databindingapp;  
  
import   
public class MainActivity extends AppCompatActivity {  
  
 private ActivityMainBinding activityMainBinding;  
 private MyClickHandler myClickHandler;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 // Data Binding with TextView  
 Person p1 = new Person("Jack","jack@gmail.com");  
  
 activityMainBinding = DataBindingUtil.*setContentView*(  
 this, R.layout.*activity\_main*);  
  
 activityMainBinding.setPerson(p1);  
  
 // Binding the Handler for Click Events  
 myClickHandler = new MyClickHandler(this);  
 activityMainBinding.setClickHandler(myClickHandler);  
  
 }  
}

**Two Way Data Binding, Quadratic App**

****

Extend Base Observable to notify property changed.  
BR = Bindable Resource, add @indable  
Data source as String to prevent error

In the layout, normal display = @{var.name},   
edit text biderctional @={var.name}  
Method ex onClick @{var:method}

To connect the method with button, use view parameter.

<?xml version="1.0" encoding="utf-8"?>  
  
<layout  
 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"  
 >  
  
 <data>  
 <variable  
 name="myEquation"  
 type="com.example.quadraticapp.Equation" />  
 </data>  
  
<androidx.constraintlayout.widget.ConstraintLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/title"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="30dp"  
 android:gravity="center"  
 android:text="Quadratic Formula Calculator"  
 android:textSize="32sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <EditText  
 android:text="@={myEquation.a}"  
 android:id="@+id/edit1"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="136dp"  
 android:hint="enter A"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <EditText  
 android:text="@={myEquation.b}"  
 android:id="@+id/edit2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="34dp"  
 android:hint="Enter B"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/edit1" />  
  
 <EditText  
 android:text="@={myEquation.c}"  
 android:id="@+id/edit3"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="32dp"  
 android:hint="Enter C"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/edit2" />  
  
 <Button  
 android:onClick="@{myEquation::solveEquation}"  
 android:id="@+id/button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="44dp"  
 android:text="Calculate"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/edit3" />  
  
 <TextView  
 android:id="@+id/result"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="167dp"  
 android:text="Result :"  
 android:textSize="32sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 android:layout\_marginStart="30sp"/>  
  
</androidx.constraintlayout.widget.ConstraintLayout>  
  
</layout>

My Equation

package com.example.quadraticapp;  
  
import …  
public class Equation extends BaseObservable {  
 String a;  
 String b;  
 String c;  
 ActivityMainBinding binding;  
  
 public Equation(ActivityMainBinding binding) {  
 this.binding = binding;  
 }  
  
 public Equation() {  
 }  
  
 @Bindable  
 public String getA() {  
 return a;  
 }  
  
 public void setA(String a) {  
 this.a = a;  
 }  
  
 @Bindable  
 public String getB() {  
 return b;  
 }  
  
 public void setB(String b) {  
 this.b = b;  
 }  
  
 @Bindable  
 public String getC() {  
 return c;  
 }  
  
 public void setC(String c) {  
 this.c = c;  
 }  
  
 public void solveEquation(View view){  
 int a = Integer.*parseInt*(getA());  
 int b = Integer.*parseInt*(getB());  
 int c = Integer.*parseInt*(getC());  
  
 double D = b\*b - 4\*a\*c;  
 double x1, x2;  
  
 if (D > 0){  
 x1 = (-b + Math.*sqrt*(D))/(2\*a);  
 x2 = (-b - Math.*sqrt*(D))/(2\*a);  
  
 binding.result.setText("Result : \n"+ x1 + "\n" + x2);  
  
 } else if (D < 0) {  
  
 binding.result.setText("Result : no real roots");  
 } else {  
 x1 = (-b + Math.*sqrt*(D))/(2\*a);  
 binding.result.setText("Result : \n"+ x1 );  
 }  
  
  
 }  
  
}

Main Activity

public class MainActivity extends AppCompatActivity {  
  
 ActivityMainBinding mainBinding;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 mainBinding = DataBindingUtil.*setContentView*(  
 this,  
 R.layout.*activity\_main* );  
  
 Equation equation = new Equation(mainBinding);  
  
 mainBinding.setMyEquation(equation);  
 }  
}

View Model + Live Data

Add dependencies

*implementation*(platform("org.jetbrains.kotlin:kotlin-bom:1.8.0"))  
  
val lifecycle\_version = "2.6.2"  
  
// ViewModel  
*implementation*("androidx.lifecycle:lifecycle-viewmodel-ktx:$lifecycle\_version")

public class MyViewModel extends ViewModel {  
 private MutableLiveData<Integer> counter = new MutableLiveData<>();  
  
 public void increaseCounter(){  
 int currentValue = counter.getValue() != null ? counter.getValue() : 0;  
  
 counter.setValue(currentValue+1);  
 }  
  
 public LiveData<Integer> getCounter() {  
 return counter;  
 }  
}

Main Activity

public class MainActivity extends AppCompatActivity {  
 ActivityMainBinding mainBinding;  
 MyViewModel viewModel;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 mainBinding = DataBindingUtil.*setContentView*(this, R.layout.*activity\_main*);  
  
 //connect the view model with an activity or fragment then make a view model based on the class given  
 viewModel = new ViewModelProvider(this).get(MyViewModel.class);  
  
 //Observing live data  
 viewModel.getCounter().observe(  
 this,  
 new Observer<Integer>() {  
 @Override  
 public void onChanged(Integer counter) {  
  
 mainBinding.count.setText(""+counter);  
 }  
 }  
 );  
 }  
}

MVVM

A reference variable can be declared as a class type or an interface type.If the variable is declared as an interface type, it can reference any object of any class that implements the interface.