PRACTICAL NO. 4

Practical 4(i)-Create an application to create Image Flipper and Image Gallery .On click on the screen the image changes.

STEP 1:-Working with the activity_main.xml file

Navigate to the app > res > layout > activity_main.xml and paste the following code to activity_main.xml file.Below is the code for the activity_main.xml file. Comments are added inside the code to understand the code in more detail.

Code:-

```
<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout

   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">

   <!--on below line we are adding view pager -->

   <androidx.viewpager.widget.ViewPager

       android:id="@+id/idViewPager"

       android:layout_width="343dp"

       android:layout_height="272dp"

       android:layout_centerInParent="true"

       android:layout_gravity="center"

       android:layout_marginStart="10dp"

       android:layout_marginLeft="10dp"

       android:layout_marginTop="10dp"

       android:layout_marginEnd="10dp"

       android:layout_marginRight="10dp"

       android:layout_marginBottom="10dp"

       android:contentDescription="hello students enjoying android " />

   android:layout_margin="10dp" />

</RelativeLayout>
```

STEP 2:- Create a layout file for ImageView in View Pager

Navigate to the app > res > layout > Right-click on it > New > Layout Resource file and specify the name as image_slider_item. Paste the following code to the image_slider_item file. Comments are added in the code to understand the code in detail.

Code:-

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
   android:layout_width="match_parent"
   android:layout_height="match_parent">
   <!--on below line we are creating an image view-->
   <ImageView
       android:id="@+id/idIVImage"
       android:layout_width="200dp"
       android:layout_height="200dp"
       android:layout_centerInParent="true" />
</RelativeLayout>
```

STEP 3:-Create a new kotlin class for the adapter of our ViewPager

Navigate to the app > kotlin+java > your file name/your package name(here our file file name is practical_no_4> Right-click on it > New > Java/Kotlin class and name it as ViewPagerAdapter . Delete all the lines in that file except the 1st line and paste the below code to it. Comments are added in the code to understand the code in detail.

Code:-

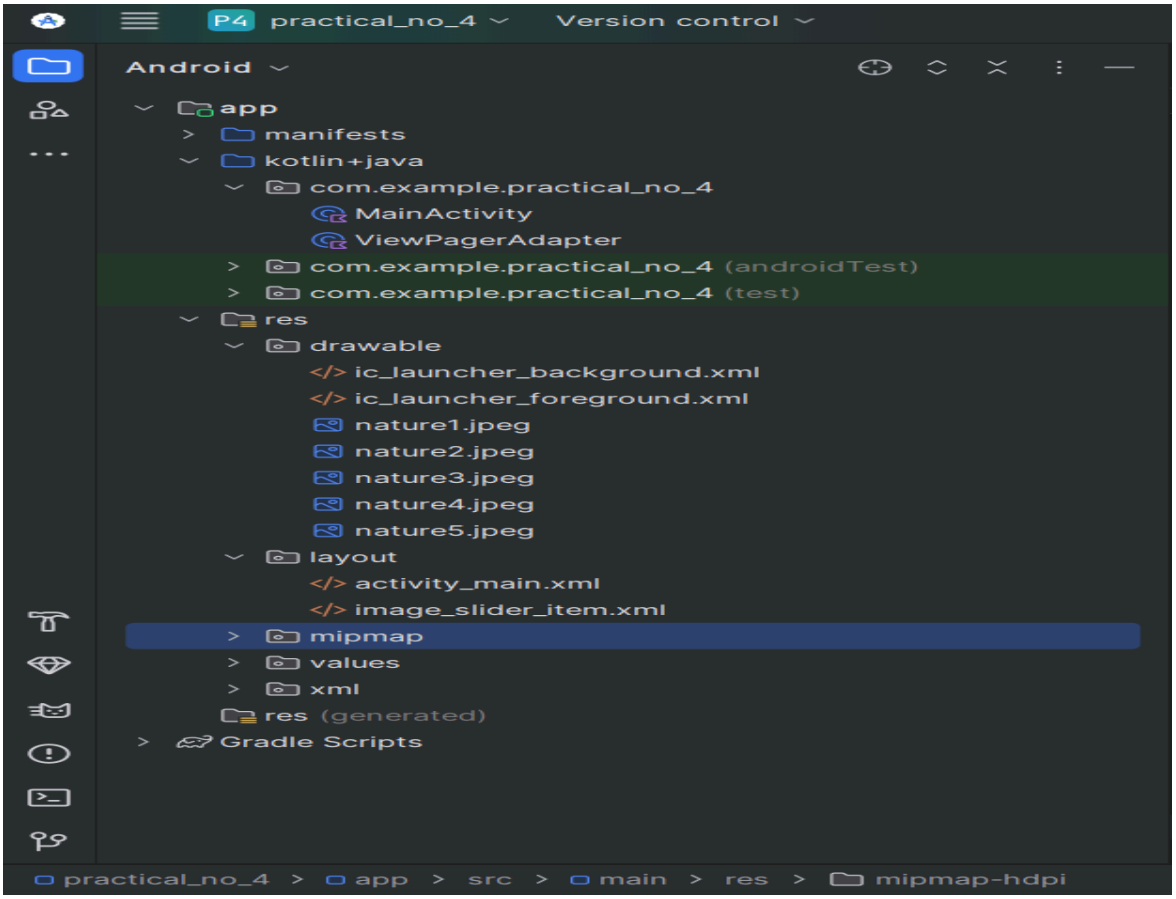
```
package com.example.practical_no_4

import android.content.Context
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.ImageView
import android.widget.RelativeLayout
import androidx.viewpager.widget.PagerAdapter
import java.util.*
class ViewPagerAdapter(val context: Context, val imageList: List<Int>)
: PagerAdapter() {  // on below line we are creating a method
   // as get count to return the size of the list.
   override fun getCount(): Int {
       return imageList.size
   }
   // on below line we are returning the object
   override fun isViewFromObject(view: View, `object`: Any): Boolean {
       return view === `object` as RelativeLayout
   }
   // on below line we are initializing
   // our item and inflating our layout file
   override fun instantiateItem(container: ViewGroup, position: Int):
Any {
       // on below line we are initializing
       // our layout inflater.
       val mLayoutInflater =
           context.getSystemService(Context.LAYOUT_INFLATER_SERVICE) as
LayoutInflater
       // on below line we are inflating our custom
       // layout file which we have created.
```

```
        val itemView: View =
mLayoutInflater.inflate(R.layout.image_slider_item, container, false)
        // on below line we are initializing
        // our image view with the id.
        val imageView: ImageView =
itemView.findViewById<View>(R.id.idIVImage) as ImageView
        // on below line we are setting
        // image resource for image view.
        imageView.setImageResource(imageList.get(position))
        // on the below line we are adding this
        // item view to the container.
        Objects.requireNonNull(container).addView(itemView)
        // on below line we are simply
        // returning our item view.
        return itemView
    }
    // on below line we are creating a destroy item method.
    override fun destroyItem(container: ViewGroup, position: Int,
`object`: Any) {  // on below line we are removing view
        container.removeView(`object` as RelativeLayout)
    }
}
```

STEP 4:- Add images to the drawable folder

Select the images which you want to add copy them Navigate to app > res > drawable and right-click on it. Simply paste it and add all the images to the drawable folder.
(for us it looks like this when we add all the images to a drawable folder.)



STEP 5:- Working with the MainActivity.kt file

Go to the MainActivity.kt file and paste the following code. Below is the code for the **MainActivity.kt** file.Delete all the lines except the first line of this file and paste the code. Comments are added inside the code to understand the code in more detail.

Code:-

```
package com.example.practical_no_4

import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity
import androidx.viewpager.widget.ViewPager
class MainActivity : AppCompatActivity() {
   // on below line we are creating variable for view pager,
   // viewpager adapter and the image list.
   lateinit var viewPager: ViewPager
   lateinit var viewPagerAdapter: ViewPagerAdapter
   lateinit var imageList: List<Int>
   override fun onCreate(savedInstanceState: Bundle?) {
       super.onCreate(savedInstanceState)
       setContentView(R.layout.activity_main)
       // initializing variables
       // of below line with their id.
       viewPager = findViewById(R.id.idViewPager)
       // on below line we are initializing
       // our image list and adding data to it.
       imageList = ArrayList<Int>()
       imageList = imageList + R.drawable.nature1 //your image name
       imageList = imageList + R.drawable.nature2 //your image name
       imageList = imageList + R.drawable.nature3 //your image name
       imageList = imageList + R.drawable.nature4 //your image name
       imageList = imageList + R.drawable.nature5 //your image name
       // on below line we are initializing our view
       // pager adapter and adding image list to it.
       viewPagerAdapter = ViewPagerAdapter(this@MainActivity, imageList)
       // on below line we are setting
       // adapter to our view pager.
       viewPager.adapter = viewPagerAdapter
   }
}
```

STEP 6:- Run the application.

Click the play button at the top or use the shortcut ‘SHIFT + F10’.

Just click on the screen and your image changes.