

**SUBJECT: ANDROID DEVELOPMENT**

**NAME:**

**CLASS:**

**ROLL NO:**

**SEMESTER/YEAR:-**

**EXAM NO:**

**DATE OF PERFORMANCE:**

**DATE OF SUBMISSION:**

**EXAMINED:-**

**REMARKS:**

**EXPERIMENT NO – 4**

**TITLE: Design a mobile application using image slider to show images.**

**OBJECTIVE:**

1. To study and design a mobile application using image slider

**PREREQUISITE:**

Students should be aware of Android Studio platform and image slider

**TOOLS USED:**

1. Android studio (Electric Eel)

**THEORY:**

Android image slider slides one entire screen to another screen. Image slider is created by ViewPager which is provided by support library. To implement image slider, you need to inherit ViewPager class which extends PagerAdapter. This type of feature is seen in many ecommerce applications which are used to display different types of banners to the customers using auto image slider widget.

The following principles are used to create a Slider:

- The PagerAdapter to populate an ArrayList of slides into a ViewPager.
- A tabbed interface to build a list of slide presentations.
- Set a timer task to control the flow of the slides.

ViewPager gets its data from a PagerAdapter. The PagerAdapter stores the slides in the memory, making it lightning fast to switch between already loaded slides.

### **Need Of image slider In Android:**

Sliders are found on cross-platforms such as website pages, desktop, and mobile apps. They are usually used to highlight important features on home screens. Slides are an excellent way to display such items to the consumer.

### **Conclusion:**

## Code –

### 1. activity\_main.xml

```
<?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="fill_parent"
    android:layout_height="fill_parent"
    android:paddingBottom="10dp"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.imageslider.MainActivity">

    <androidx.viewpager.widget.ViewPager
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:id="@+id/viewPage"/>

</RelativeLayout>
```

### 2. MainActivity.Java

```
package com.example.imageslider;

import androidx.appcompat.app.AppCompatActivity;
import androidx.viewpager.widget.ViewPager;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        ViewPager mViewPager = (ViewPager) findViewById(R.id.viewPage);
        ImageAdapter adapterView = new ImageAdapter(this);
        mViewPager.setAdapter(adapterView);
    }
}
```

### 3. ImageAdapter.Java

```
package com.example.imageslider;

import androidx.annotation.NonNull;
import androidx.viewpager.widget.PagerAdapter;
import androidx.viewpager.widget.ViewPager;
import android.content.Context;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;

public class ImageAdapter extends PagerAdapter{
    Context mContext;

    ImageAdapter(Context context) {
        this.mContext = context;
    }

    @Override
    public boolean isViewFromObject(View view, Object object) {
        return view == ((ImageView) object);
    }

    private final int[] sliderImageId = new int[]{
        R.drawable.image1, R.drawable.image2,R.drawable.image3, R.drawable.image4,
    };

    @NonNull
    @Override
    public Object instantiateItem(@NonNull ViewGroup container, int position) {
        ImageView imageView = new ImageView(mContext);
        imageView.setScaleType(ImageView.ScaleType.CENTER_CROP);
        imageView.setImageResource(sliderImageId[position]);
        ((ViewPager) container).addView(imageView, 0);
        return imageView;
    }

    @Override
```

```
public void destroyItem(@NonNull ViewGroup container, int position, @NonNull Object object) {  
    ((ViewPager) container).removeView((ImageView) object);  
}  
  
@Override  
public int getCount() {  
    return sliderImageId.length;  
}  
}
```

## Output –



