Assignment No: 07

Aim:- Create an android app that demonstrate Screen Navigation Using the App Bar and Tabs.

Code:-

Basics

Typically, an implementation of tabs in Android consists of:

1. Swipe views
2. Tabs UI element

These are two independent navigation patterns, but they can be combined with each other.

In general: swipe views *can* be combined with tabs, and tabs be combined with swipe views. However, tabs benefit tremendously from being combined with swipe views, as explained below.

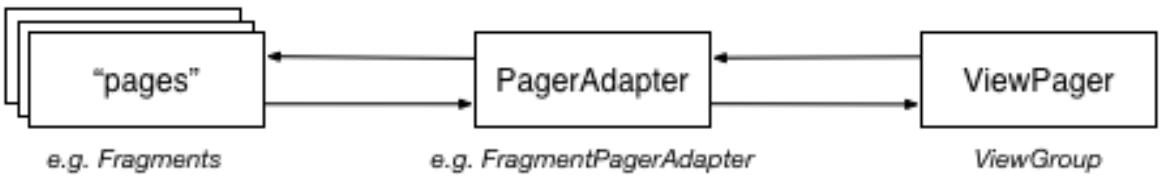
Swipe Views

Swipe views allow to flip through a set of "pages" by swiping horizontally on the screen.

Combined with tabs, this allows the user to switch to the next or previous tab by just swiping anywhere on the screen, rather than having to click on the tab itself.

- Swipe views are implemented by the ViewPager ViewGroup (declared in the activity's layout XML)
- A page is typically implemented as a Fragment
- A PagerAdapter supplies the ViewPager with the pages (fragments) to display. In the case of using fragments as pages, this PagerAdapter is a FragmentPagerAdapter or FragmentStatePagerAdapter

The swipe views architecture is illustrated below:



TabLayout Tabs

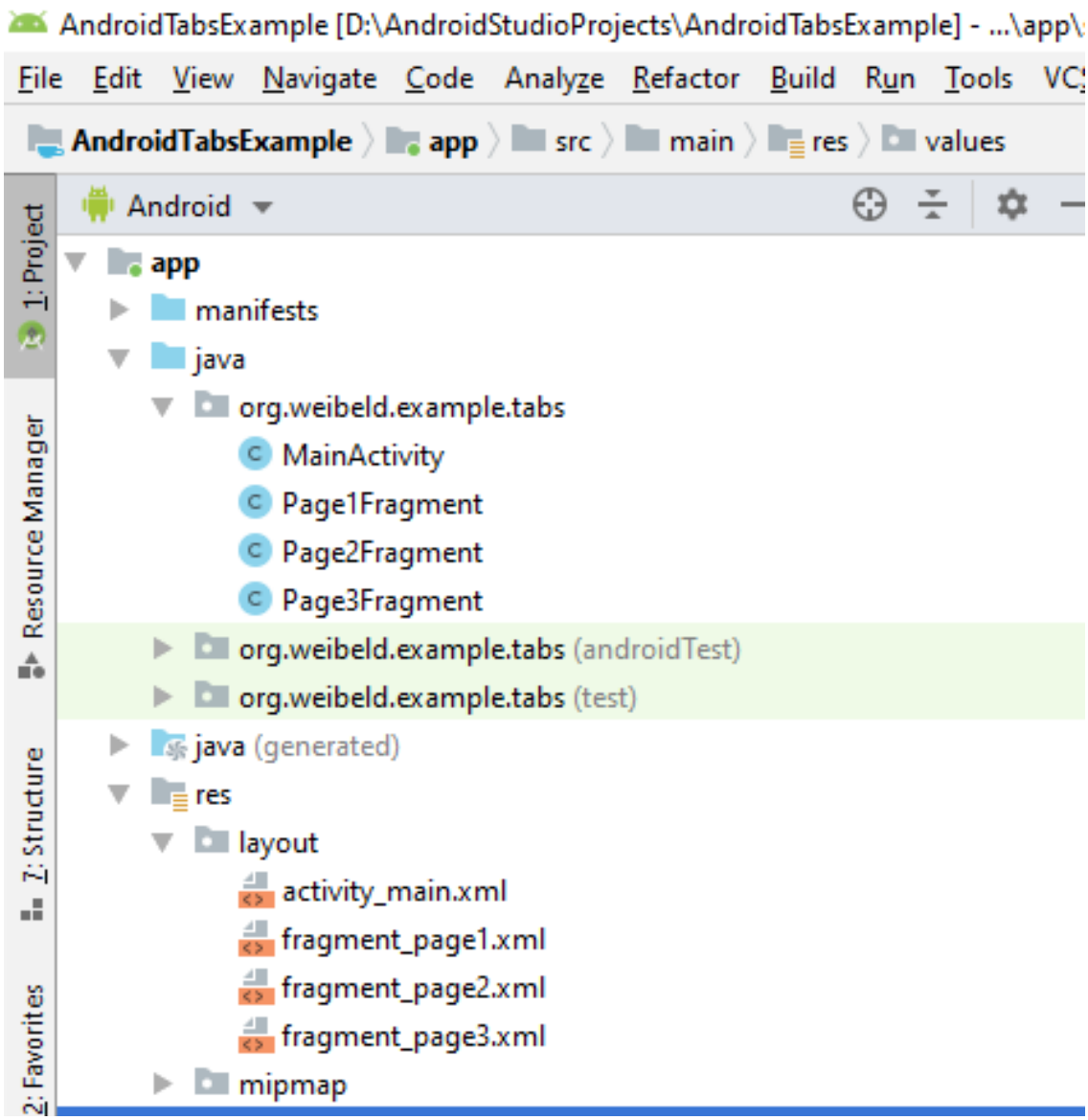
This is the preferred approach as it's easier to implement than ActionBar tabs and does not rely on an ActionBar.

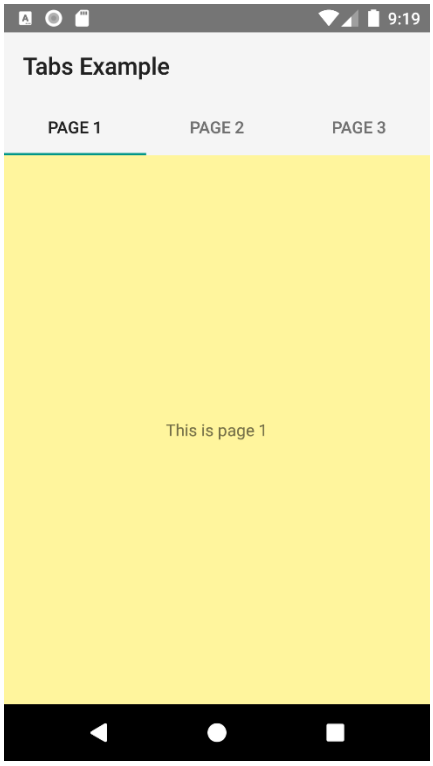
TabLayout tabs are especially easy to implement if they are used in combination with a formerly implemented ViewPager (see Swipe Views), because then the tabs can be automatically populated by the ViewPager (i.e. no need to create and add the individual tabs manually). This is shown in the following:

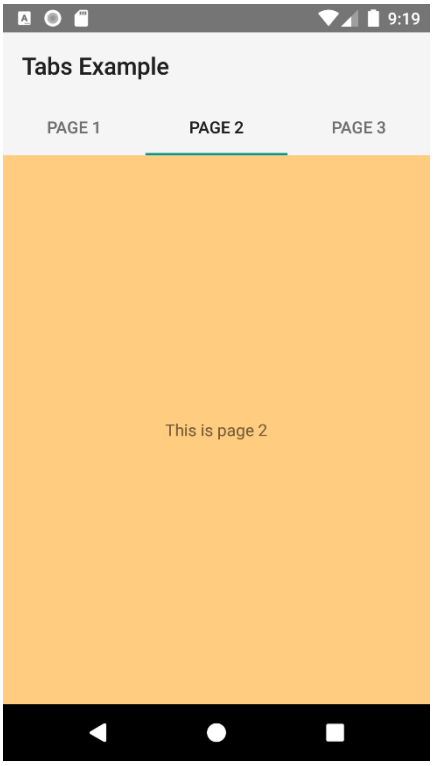
1. In the activity's layout XML, add a TabLayout element above the ViewPager element (and below the Toolbar element, if a Toolbar is used)
2. In the activity's onCreate method, call setupWithViewPager(ViewPager) on the TabLayout to populate and integrate the TabLayout with the ViewPager (requires that getPageTitle of the PagerAdapter is overriden)

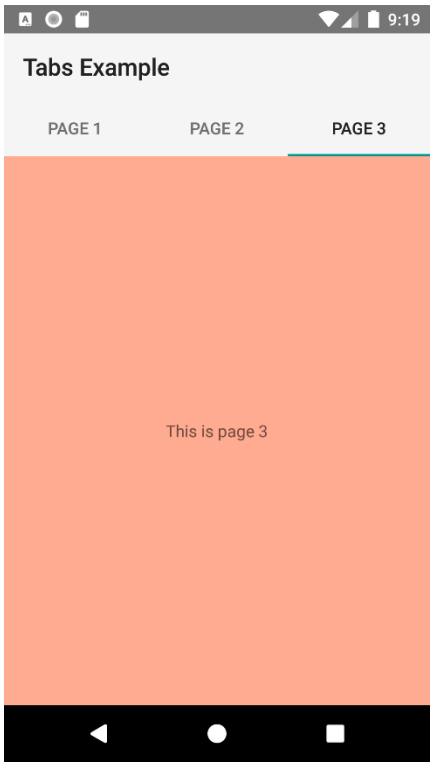
Note: to use TabLayout, the *Design Support Library* must be added to the project. In the module build.gradle file, do:

```
dependencies {
   compile 'com.android.support:support-v13:24.2.1'
}
```









activity_main.xml

```
<?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:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context="org.weibeld.example.tabs.MainActivity">

    <android.support.v7.widget.Toolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:minHeight="?attr/actionBarSize"
        android:background="?attr/colorPrimary"
        />

    <android.support.design.widget.TabLayout
        android:id="@+id/tab_layout"
        android:layout_height="wrap_content"
        android:layout_width="match_parent"
        android:background="?attr/colorPrimary"
        />

    <android.support.v4.view.ViewPager
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:id="@+id/viewpager"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        />

</LinearLayout>
```

fragment_page1.xml

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

<FrameLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/pageColor1">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="This is page 1" />

</FrameLayout>
```

fragment_page2.xml

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

<FrameLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:background="@color/pageColor2">

   <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_gravity="center"
      android:text="This is page 2" />

</FrameLayout>
```

fragment_page3.xml

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

<FrameLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:background="@color/pageColor3">

   <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_gravity="center"
      android:text="This is page 3" />

</FrameLayout>
```

Page1Fragment.java

```
package org.weibeld.example.tabs;

import android.app.Fragment;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

import org.weibeld.example.R;

/* Fragment used as page 1 */
public class Page1Fragment extends Fragment {
```

```
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState) {
        View rootView = inflater.inflate(R.layout.fragment_page1, container, false);
        return rootView;
    }
}
```

Page2Fragment.java

```
package org.weibeld.example.tabs;

import android.app.Fragment;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

import org.weibeld.example.R;

/* Fragment used as page 2 */
public class Page2Fragment extends Fragment {

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState) {
        View rootView = inflater.inflate(R.layout.fragment_page2, container, false);
        return rootView;
    }

}
```

Page3Fragment.java

```
package org.weibeld.example.tabs;

import android.app.Fragment;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

import org.weibeld.example.R;

/* Fragment used as page 3 */
public class Page3Fragment extends Fragment {

    @Override
```

```
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState) {
        View rootView = inflater.inflate(R.layout.fragment_page3, container, false);
        return rootView;
    }
}
```

MainActivity.java

```
package org.weibeld.example.tabs;

import android.app.Fragment;
import android.app.FragmentManager;
import android.os.Bundle;
import android.support.design.widget.TabLayout;
import android.support.v13.app.FragmentPagerAdapter;
import android.support.v4.view.ViewPager;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;

import org.weibeld.example.R;

public class MainActivity extends AppCompatActivity {

    private final String LOG_TAG = MainActivity.class.getSimpleName();

    // Titles of the individual pages (displayed in tabs)
    private final String[] PAGE_TITLES = new String[] {
            "Page 1",
            "Page 2",
            "Page 3"
    };

    // The fragments that are used as the individual pages
    private final Fragment[] PAGES = new Fragment[] {
            new Page1Fragment(),
            new Page2Fragment(),
            new Page3Fragment()
    };

    // The ViewPager is responsible for sliding pages (fragments) in and out upon user input
    private ViewPager mViewPager;

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

        // Set the Toolbar as the activity's app bar (instead of the default ActionBar)
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
```

```
        setSupportActionBar(toolbar);

        // Connect the ViewPager to our custom PagerAdapter. The PagerAdapter supplies the
pages
        // (fragments) to the ViewPager, which the ViewPager needs to display.
        mViewPager = (ViewPager) findViewById(R.id.viewpager);
        mViewPager.setAdapter(new MyPagerAdapter(getFragmentManager()));

        // Connect the tabs with the ViewPager (the setupWithViewPager method does this for
us in
        // both directions, i.e. when a new tab is selected, the ViewPager switches to this page,
        // and when the ViewPager switches to a new page, the corresponding tab is selected)
        TabLayout tabLayout = (TabLayout) findViewById(R.id.tab_layout);
        tabLayout.setupWithViewPager(mViewPager);
    }


    /* PagerAdapter for supplying the ViewPager with the pages (fragments) to display. */
    public class MyPagerAdapter extends FragmentPagerAdapter {

        public MyPagerAdapter(FragmentManager fragmentManager) {
            super(fragmentManager);
        }

        @Override
        public Fragment getItem(int position) {
            return PAGES[position];
        }

        @Override
        public int getCount() {
            return PAGES.length;
        }

        @Override
        public CharSequence getPageTitle(int position) {
            return PAGE_TITLES[position];
        }

    }
}
```