

Android Fragments

Android Fragment is the part of activity, it is also known as sub-activity. There can be more than one fragment in an activity. Fragments represent multiple screen inside one activity.

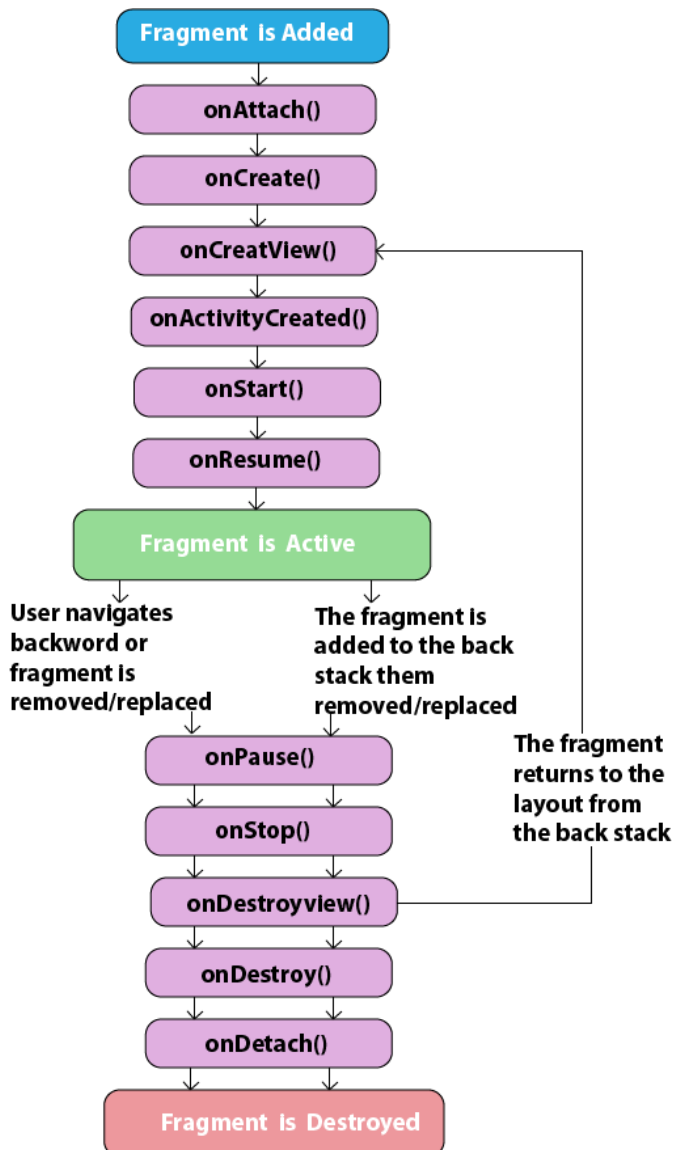
Android fragment lifecycle is affected by activity lifecycle because fragments are included in activity.

Each fragment has its own life cycle methods that is affected by activity life cycle because fragments are embedded in activity.

The **FragmentManager** class is responsible to make interaction between fragment objects.

Android Fragment Lifecycle

The lifecycle of android fragment is like the activity lifecycle. There are 12 lifecycle methods for fragment.



Android Fragment Lifecycle Methods

| No. | Method | Description |
|-----|---|--|
| 1) | onAttach(Activity) | it is called only once when it is attached with activity. |
| 2) | onCreate(Bundle) | It is used to initialize the fragment. |
| 3) | onCreateView(LayoutInflater, ViewGroup, Bundle) | creates and returns view hierarchy. |
| 4) | onActivityCreated(Bundle) | It is invoked after the completion of onCreate() method. |
| 5) | onViewStateRestored(Bundle) | It provides information to the fragment that all the saved state of fragment view hierarchy has been restored. |
| 6) | onStart() | makes the fragment visible. |
| 7) | onResume() | makes the fragment interactive. |
| 8) | onPause() | is called when fragment is no longer interactive. |
| 9) | onStop() | is called when fragment is no longer visible. |
| 10) | onDestroyView() | allows the fragment to clean up resources. |
| 11) | onDestroy() | allows the fragment to do final clean up of fragment state. |
| 12) | onDetach() | It is called immediately prior to the fragment no longer being associated with its activity. |

Android Fragment Example

Let's have a look at the simple example of android fragment.

activity_main.xml

File: activity_main.xml

1. `<?xml version="1.0" encoding="utf-8"?>`
2. `<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"`
3. `xmlns:app="http://schemas.android.com/apk/res-auto"`

```

4.   xmlns:tools="http://schemas.android.com/tools"
5.   android:layout_width="fill_parent"
6.   android:layout_height="fill_parent"
7.   tools:context="example.javatpoint.com.fragmentexample.MainActivity">
8.
9.   <fragment
10.    android:id="@+id/fragment1"
11.    android:name="example.javatpoint.com.fragmentexample.Fragment1"
12.    android:layout_width="0px"
13.    android:layout_height="match_parent"
14.    android:layout_weight="1"
15.  />
16.
17.  <fragment
18.    android:id="@+id/fragment2"
19.    android:name="example.javatpoint.com.fragmentexample.Fragment2"
20.    android:layout_width="0px"
21.    android:layout_height="match_parent"
22.    android:layout_weight="1"
23.  />
24.
25. </LinearLayout>

```

File: fragment_fragment1.xml

```

1.  <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
2.    xmlns:tools="http://schemas.android.com/tools"
3.    android:layout_width="match_parent"
4.    android:layout_height="match_parent"
5.    android:background="#F5F5DC"
6.    tools:context="example.javatpoint.com.fragmentexample.Fragment1">
7.
8.    <!-- TODO: Update blank fragment layout -->
9.    <TextView
10.     android:layout_width="match_parent"
11.     android:layout_height="match_parent"
12.     android:text="@string/hello_blank_fragment" />
13.
14. </FrameLayout>

```

File: fragment_fragment2.xml

```

1. <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
2.     xmlns:tools="http://schemas.android.com/tools"
3.     android:layout_width="match_parent"
4.     android:layout_height="match_parent"
5.     android:background="#F0FFFF"
6.     tools:context="example.javatpoint.com.fragmentexample.Fragment2">
7.
8.     <!-- TODO: Update blank fragment layout -->
9.     <TextView
10.         android:layout_width="match_parent"
11.         android:layout_height="match_parent"
12.         android:text="@string/hello_blank_fragment" />
13.
14. </FrameLayout>

```

MainActivity class

File: MainActivity.java

```

1. import android.support.v7.app.AppCompatActivity;
2. import android.os.Bundle;
3.
4. public class MainActivity extends AppCompatActivity {
5.
6.     @Override
7.     protected void onCreate(Bundle savedInstanceState) {
8.         super.onCreate(savedInstanceState);
9.         setContentView(R.layout.activity_main);
10.    }
11.}

```

File: Fragment1.java

```

1. import android.os.Bundle;
2. import android.support.v4.app.Fragment;
3. import android.view.LayoutInflater;
4. import android.view.View;
5. import android.view.ViewGroup;
6.
7. public class Fragment1 extends Fragment {
8.

```

```

9.     @Override
10.    public void onCreate(Bundle savedInstanceState) {
11.        super.onCreate(savedInstanceState);
12.    }
13.
14.    @Override
15.    public View onCreateView(LayoutInflater inflater, ViewGroup container,
16.                            Bundle savedInstanceState) {
17.        // Inflate the layout for this fragment
18.        return inflater.inflate(R.layout.fragment_fragment1, container, false);
19.    }
20. }

```

File: Fragment2.java

```

1. import android.os.Bundle;
2. import android.support.v4.app.Fragment;
3. import android.view.LayoutInflater;
4. import android.view.View;
5. import android.view.ViewGroup;
6.
7. public class Fragment2 extends Fragment {
8.
9.     @Override
10.    public void onCreate(Bundle savedInstanceState) {
11.        super.onCreate(savedInstanceState);
12.    }
13.
14.    @Override
15.    public View onCreateView(LayoutInflater inflater, ViewGroup container,
16.                            Bundle savedInstanceState) {
17.        // Inflate the layout for this fragment
18.        return inflater.inflate(R.layout.fragment_fragment2, container, false);
19.    }
20.
21. }

```

Output:

