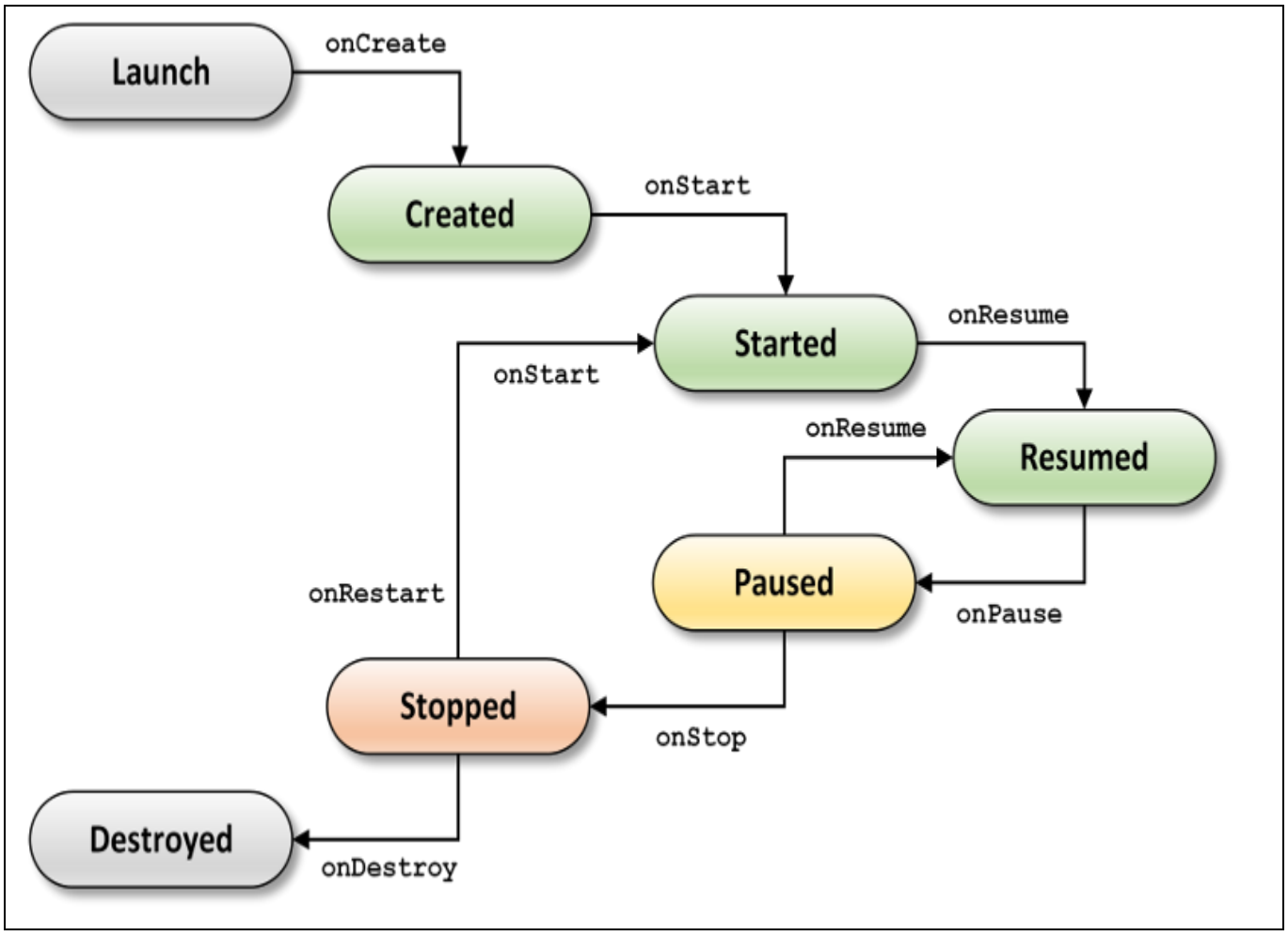
Practical 3

Programming Activities and fragments

Activity Life Cycle, Activity methods, Multiple Activities, Life Cycle of fragments and multiple fragments.

a) Activity Life Cycle

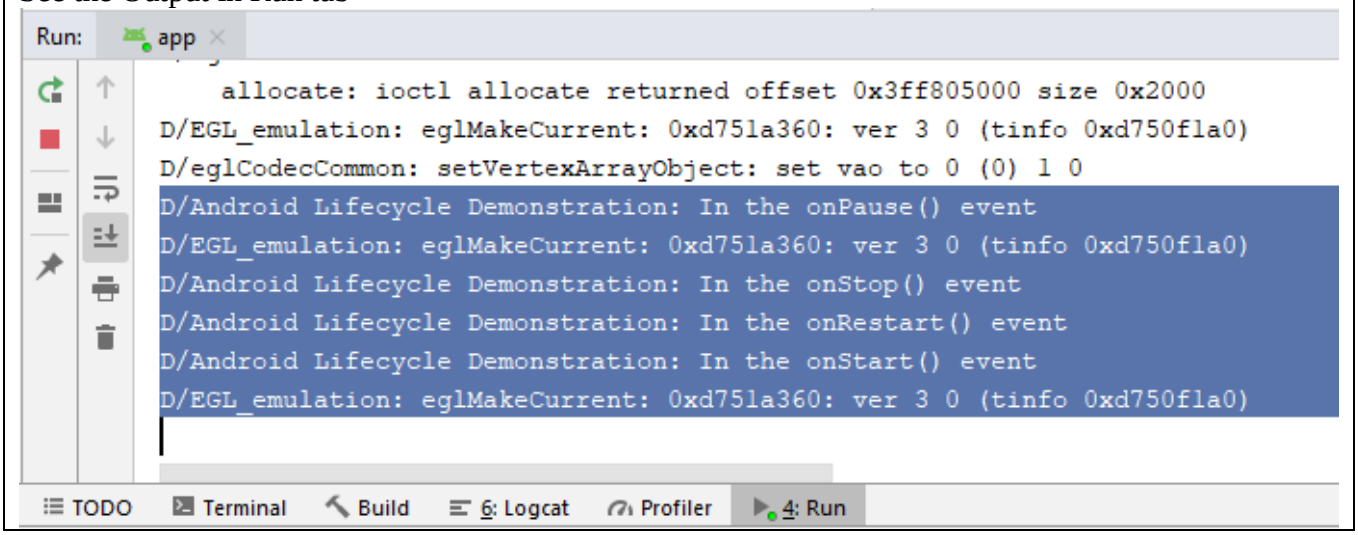
The various methods to be created are onStart(), onRestart(), onStop(), onResume(),onDestroy() and onPause() as shown in the figure.



Create a new project and go to, **MainActivity.java**

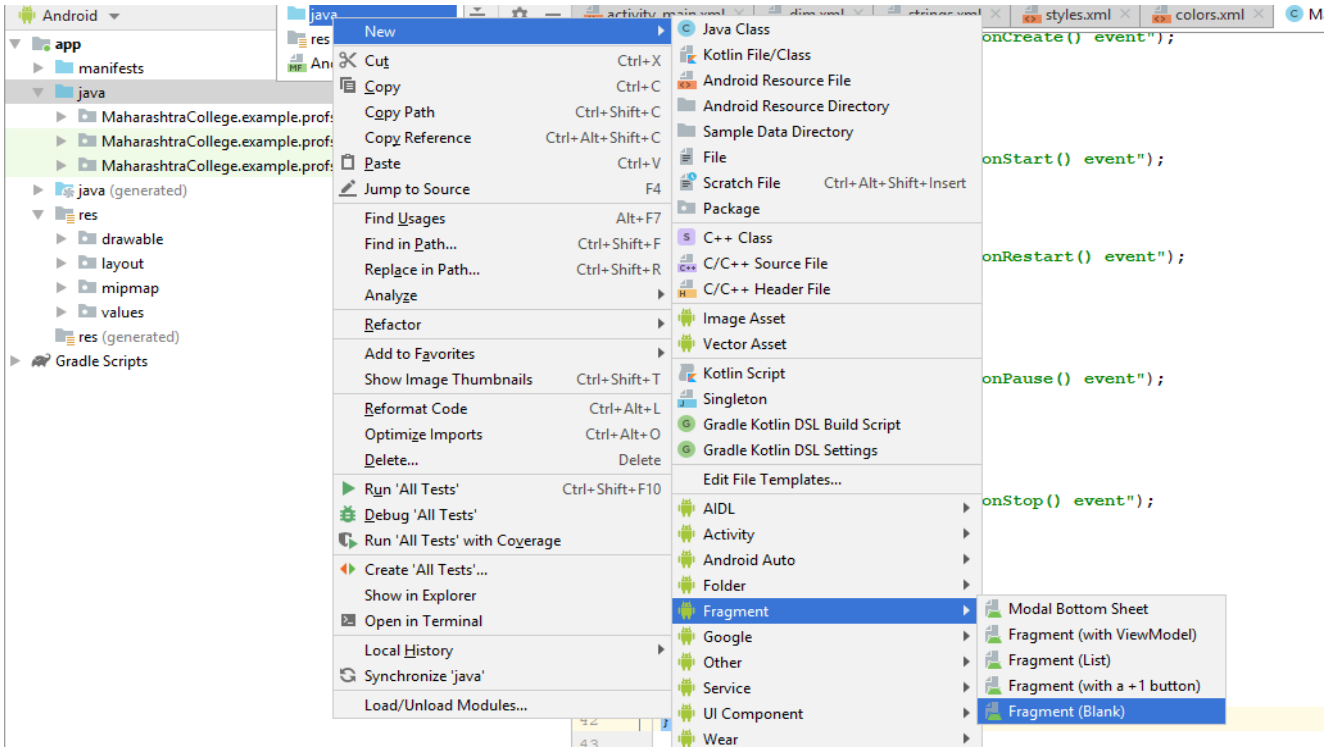
```
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.util.Log;
public class MainActivity extends AppCompatActivity {

    String tag= "Android Lifecycle Demonstration";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Log.d(tag,"In the onCreate() event");
    }
    public void onStart()
    {
        super.onStart();
        Log.d(tag,"In the onStart() event");
    }
    public void onRestart()
    { super.onRestart();
        Log.d(tag,"In the onRestart() event");
    }
    public void onPause()
    {
        super.onPause();
        Log.d(tag,"In the onPause() event");
    }
    public void onStop()
    {
        super.onStop();
        Log.d(tag,"In the onStop() event");
    }
    public void onDestroy()
    {
        super.onDestroy();
        Log.d(tag,"In the onDestroy() event");
    }


}
```

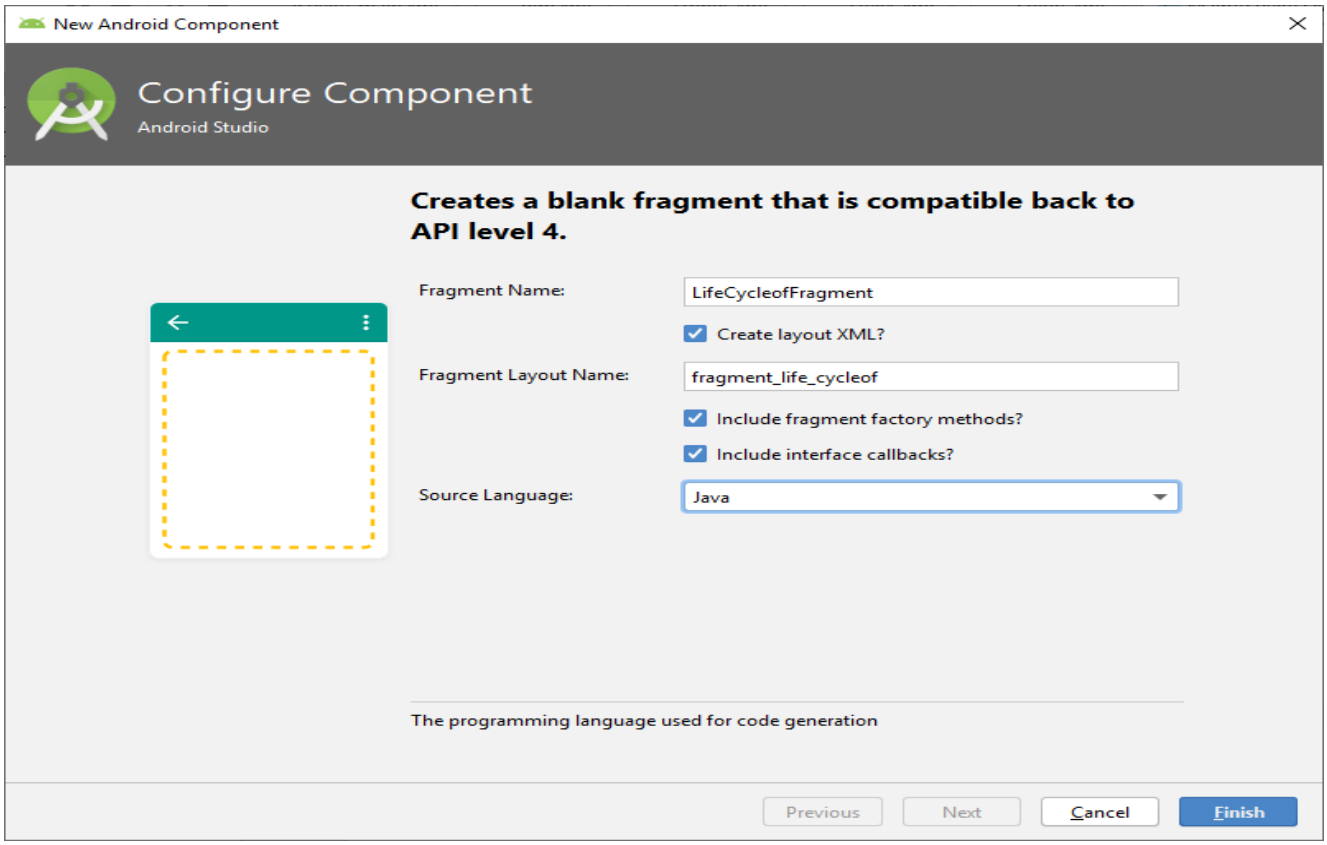
Output
See the Output in Run tab



b) Life Cycle of fragments

Create a new project
Add a new Fragment
ProjectName>App>src>main>Java
3. Right click on Java and select **'Add Fragment'>Add Fragment(Blank)**
Give a name to the fragment and click Finish


Android
app
manifests
java
MaharashtraCollege.example.profs
MaharashtraCollege.example.profs
MaharashtraCollege.example.profs
java (generated)
res
drawable
layout
mipmap
values
res (generated)
Gradle Scripts
New
Cut Ctrl+X
Copy Ctrl+C
Copy Path Ctrl+Shift+C
Copy Reference Ctrl+Alt+Shift+C
Paste Ctrl+V
Jump to Source F4
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Analyze
Refactor
Add to Favorites
Show Image Thumbnails Ctrl+Shift+T
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Run 'All Tests' Ctrl+Shift+F10
Debug 'All Tests'
Run 'All Tests' with Coverage
Create 'All Tests'...
Show in Explorer
Open in Terminal
Local History
Synchronize 'java'
Load/Unload Modules...
Java Class
Kotlin File/Class
Android Resource File
Android Resource Directory
Sample Data Directory
File
Scratch File Ctrl+Alt+Shift+Insert
Package
C++ Class
C/C++ Source File
C/C++ Header File
Image Asset
Vector Asset
Kotlin Script
Singleton
Gradle Kotlin DSL Build Script
Gradle Kotlin DSL Settings
Edit File Templates...
AIDL
Activity
Android Auto
Folder
Fragment
Google
Other
Service
UI Component
Wear
Modal Bottom Sheet
Fragment (with ViewModel)
Fragment (List)
Fragment (with a +1 button)
Fragment (Blank)
styles.xml
colors.xml
onCreate() event");
onStart() event");
onRestart() event");
onPause() event");
onStop() event");


New Android Component
Configure Component
Android Studio
Creates a blank fragment that is compatible back to API level 4.
Fragment Name:
LifeCycleofFragment
Create layout XML?
Fragment Layout Name:
fragment_life_cycleof
Include fragment factory methods?
Include interface callbacks?
Source Language:
Java
The programming language used for code generation
Previous
Next
Cancel
Finish

```
import android.content.Context;
import android.net.Uri;
import android.os.Bundle;

import androidx.fragment.app.Fragment;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.util.Log;

public class LifeCycleofFragment extends Fragment {
    String tag = "Life Cycle of Fragment";
    // TODO: Rename parameter arguments, choose names that match
    // the fragment initialization parameters, e.g. ARG_ITEM_NUMBER
    private static final String ARG_PARAM1 = "param1";
    private static final String ARG_PARAM2 = "param2";

    // TODO: Rename and change types of parameters
    private String mParam1;
    private String mParam2;

    private OnFragmentInteractionListener mListener;

    public LifeCycleofFragment() {

    // TODO: Rename and change types and number of parameters
    public static LifeCycleofFragment newInstance(String param1, String
param2) {
        LifeCycleofFragment fragment = new LifeCycleofFragment();
        Bundle args = new Bundle();
        args.putString(ARG_PARAM1, param1);
        args.putString(ARG_PARAM2, param2);
        fragment.setArguments(args);
        return fragment;
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
```

```
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.fragment_life_cycleof, container,
false);
    }

    // TODO: Rename method, update argument and hook method into UI event
    public void onButtonPressed(Uri uri) {
        if (mListener != null) {
            mListener.onFragmentInteraction(uri);
        }
    }

    @Override
    public void onAttach(Context context) {
        super.onAttach(context);
        if (context instanceof OnFragmentInteractionListener) {
            mListener = (OnFragmentInteractionListener) context;
        } else {
            throw new RuntimeException(context.toString()
                    + " must implement OnFragmentInteractionListener");
        }
    }

    @Override
    public void onDetach() {
        super.onDetach();
        mListener = null;
    }

    public interface OnFragmentInteractionListener {
        // TODO: Update argument type and name
        void onFragmentInteraction(Uri uri);
    }

    public void onStart()
    {
        super.onStart();
        Log.d(tag,"In the onStart() event");
    }
    public void onRestart()
    {

        Log.d(tag,"In the onRestart() event");
    }
    public void onPause()
    {
        super.onPause();
        Log.d(tag,"In the onPause() event");
    }
    public void onStop()
    {
        super.onStop();
        Log.d(tag,"In the onStop() event");
    }
```

```
    public void onDestroy()
    {
        super.onDestroy();
        Log.d(tag,"In the onDestroy() event");
    }
}
```

Output