**DAY1**

**Goal:**

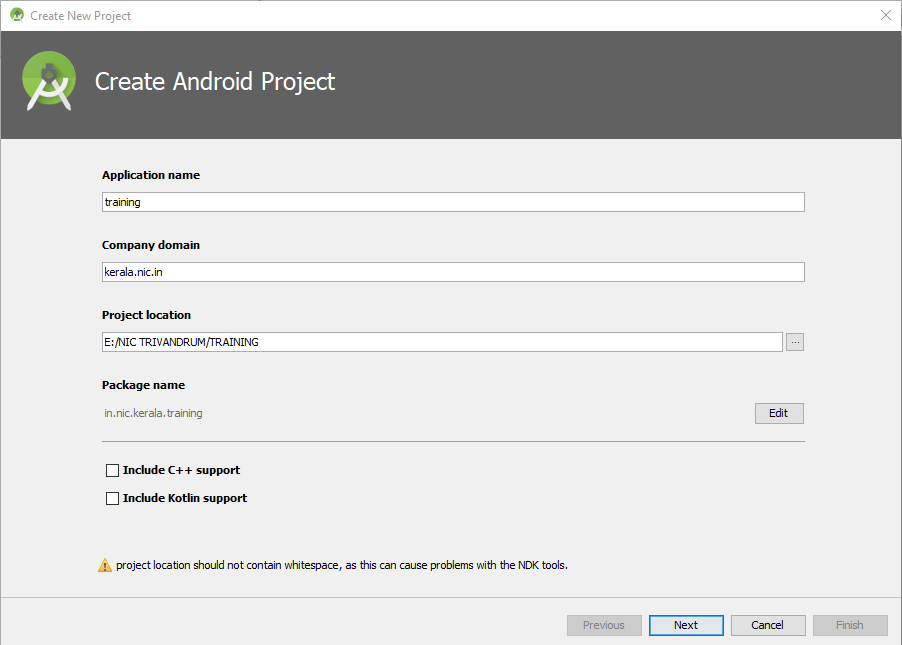
Create a Splash Screen  
Create a Dashboard with grid (These grid items will be the navigation items to further screens)  
Create a navigation Drawer  
Create Floating Action Bar

**You will Learn:**

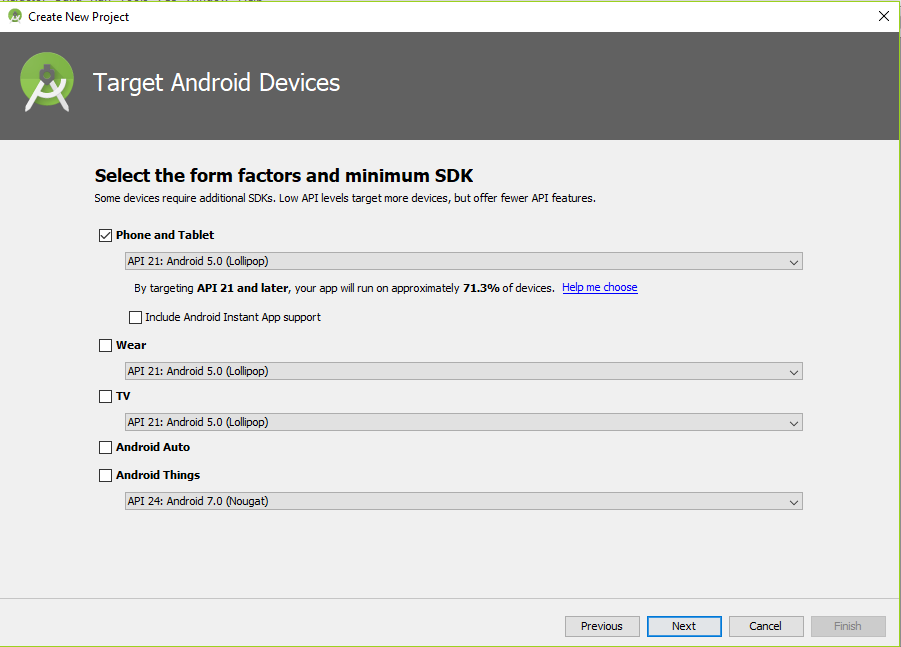
Android Studio basics  
Android project structure  
Android emulator  
Android Activity  
Android Fragments  
Android Adapters  
Android Layouts (grid , various layouts)

# LETS BEGIN

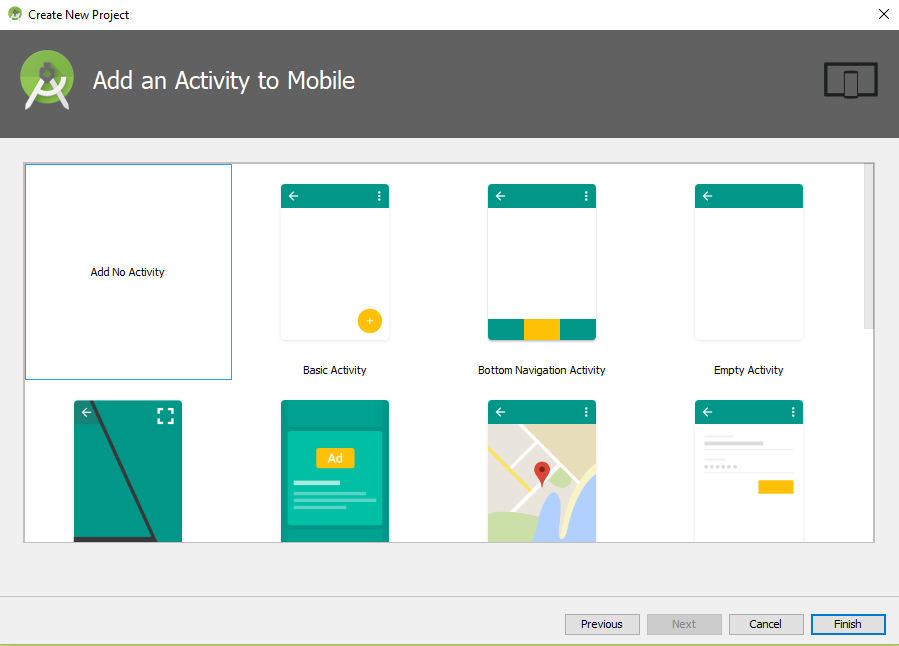
## Create an android studio Project



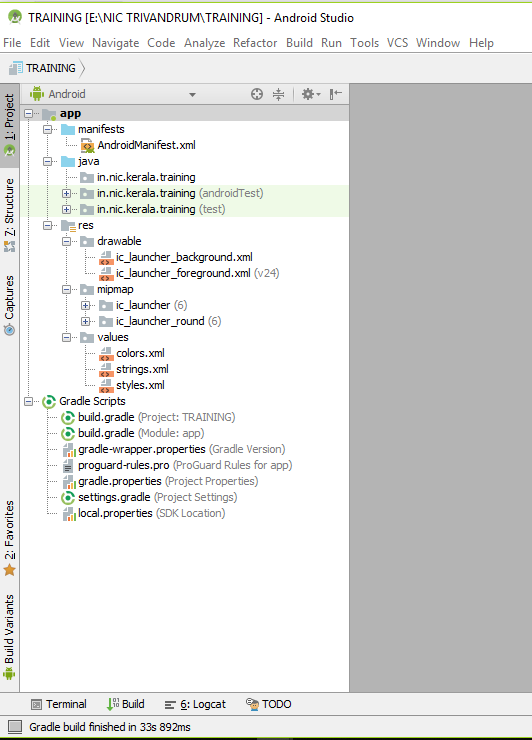
**Step 1**Provide the project details .Notice the package name *training.kerala.nic.in*



**Step 2**Target the minimum android version support which your application is providing. This is important for a reach of big audience.

****

**Step 3**Start with an empty project as shown in the picture. Click on **finish**, you can see the project structure as shown below



Project dependency and build configurations

Launcher  
ICON

Your resources  
Images  
Vectors  
Fonts etc

Your java programs

Android project configuration file

## splashCreate a Splash screen

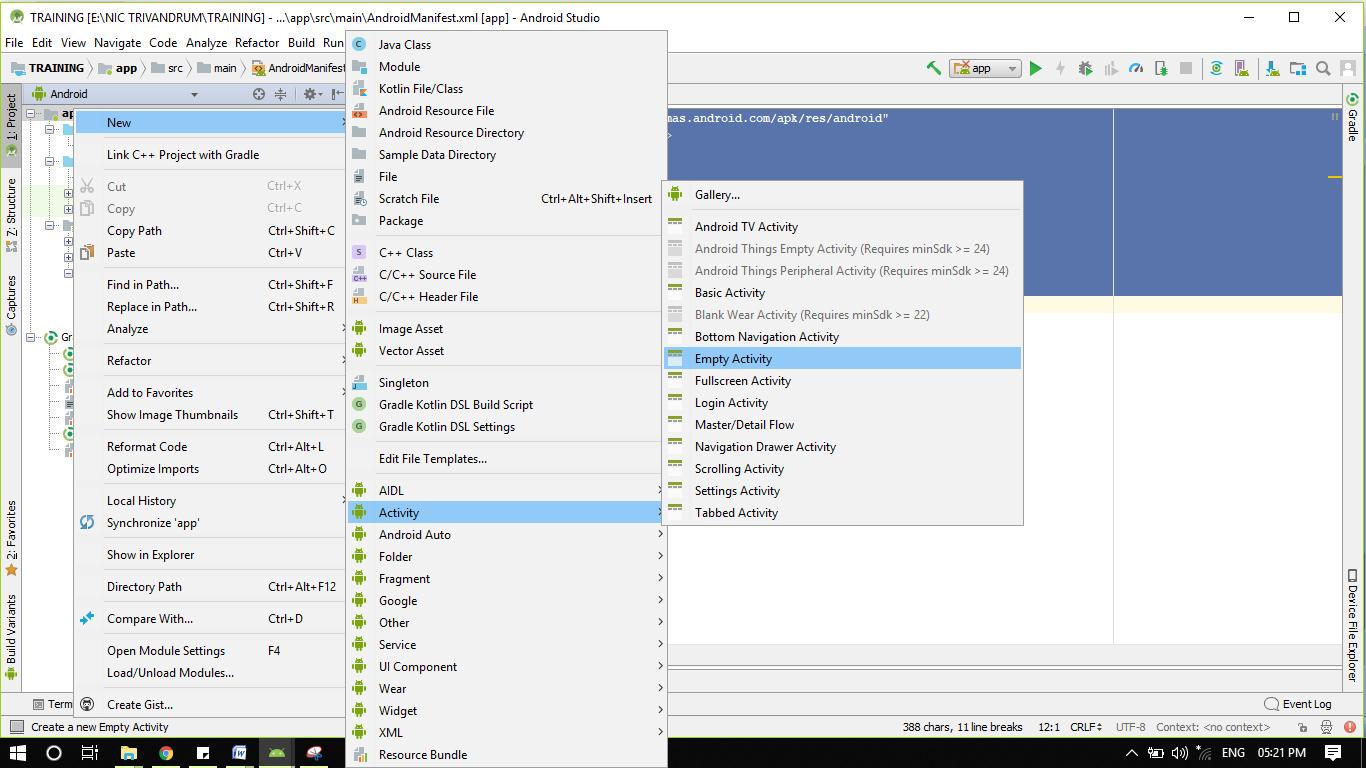
Create a new activity by Right clicking and selecting empty activity

Before starting this session, you may verify the following files in your project

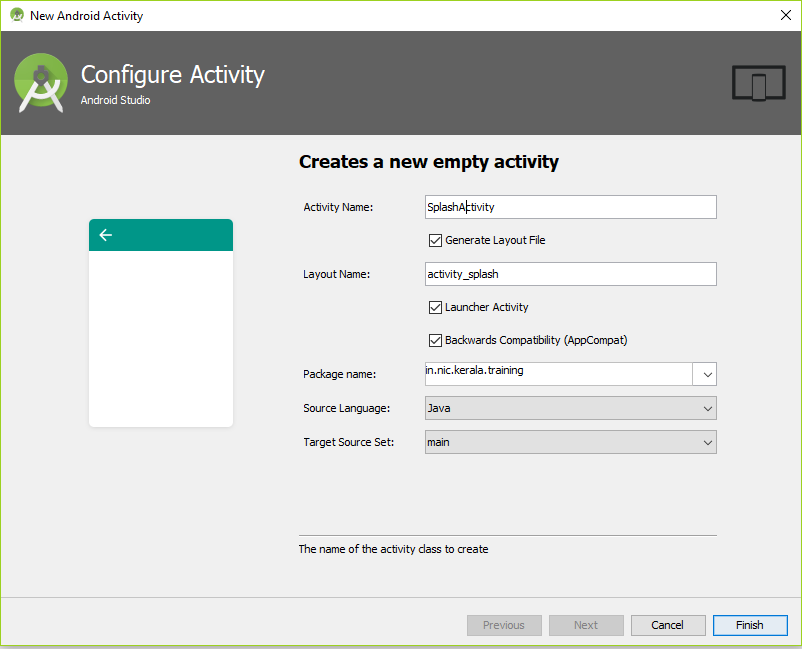
1. AndroidManifest.xml
2. Build.gradle(APP)
3. proguard-rules.pro

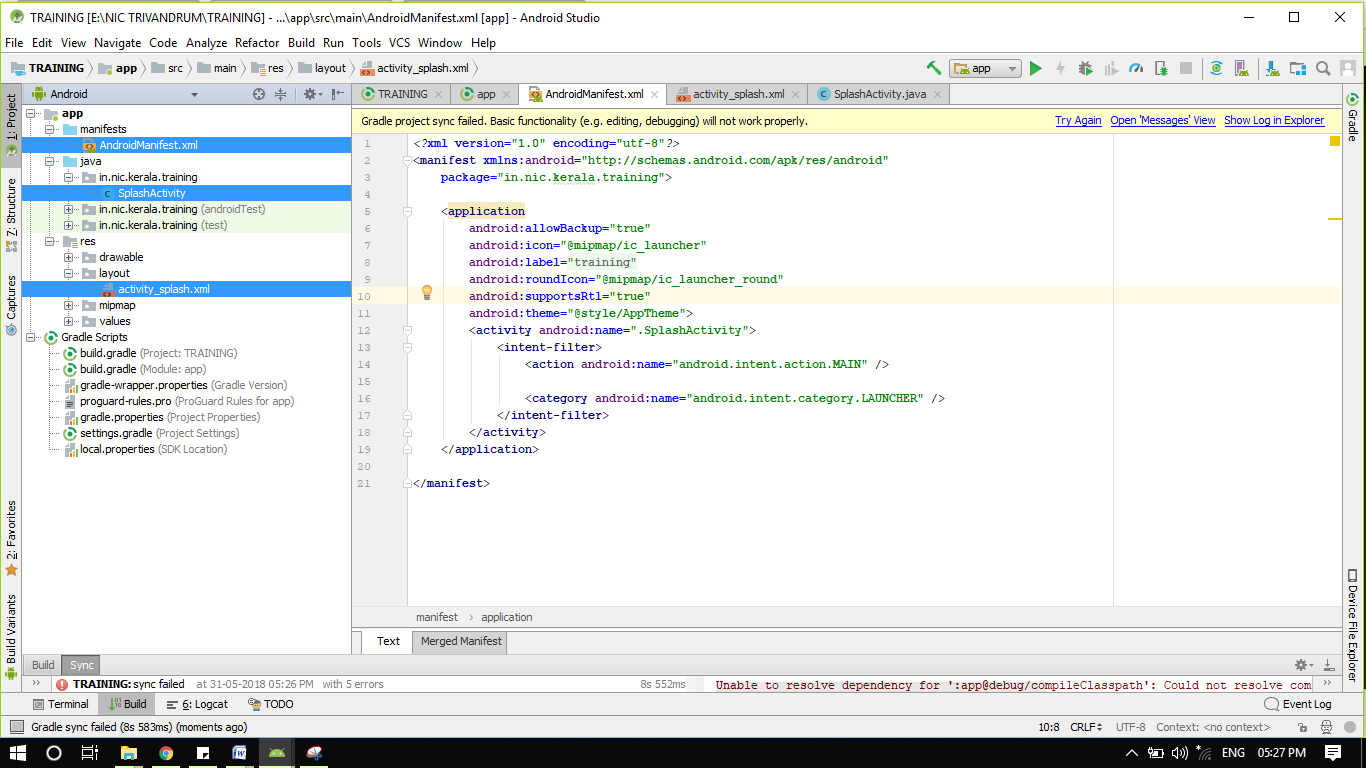
You are about to create a splash screen as shown in the picture

The splash will appear for some time and then navigates user to dashboard



Provide activity name as Splashactivity , layout name as

You can see the following file changes as in the image



a. Add below code in "activity\_splash.xml" file.

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="vertical" android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="#87CEFA"**>  
  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerInParent="true"  
 android:text="Android Training Programme"  
 android:layout\_centerHorizontal="true"  
 android:textSize="20dip"  
 android:textStyle="bold"** />  
  
 <**LinearLayout  
 android:id="@+id/in"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="49dp"  
 android:layout\_alignParentBottom="true"  
 android:background="@drawable/fooo"  
 android:orientation="horizontal"** />  
</**RelativeLayout**>

b. Add below code in "SplashActivity.java" file.

Note: Before doing this step, please go through the present code in the same file. Now overwrite the contents as

**package** in.nic.kerala.training;  
  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.os.Handler;  
**import** android.support.annotation.Nullable;  
**import** android.support.v7.app.AppCompatActivity;  
  
**public class** SplashActivity **extends** AppCompatActivity{  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_splash***);  
 **new** Handler().postDelayed(**new** Runnable() {  
  
 @Override  
 **public void** run() {  
 Intent in = **new** Intent(SplashActivity.**this**, **DashboardActivity.class**);  
 startActivity(in);  
  
  
 finish();  
 }  
 }, 2\*1000);  
  
 }  
  
 @Override  
 **protected void** onDestroy() {  
 **super**.onDestroy();  
 }  
}

Above code is self explanatory, which creates a handler which executes after the mentioned period of time. (Like a Timer). On timeout it calls another activity

c. Changes in manifest file

Open androidmanifest.xml file and add the following property

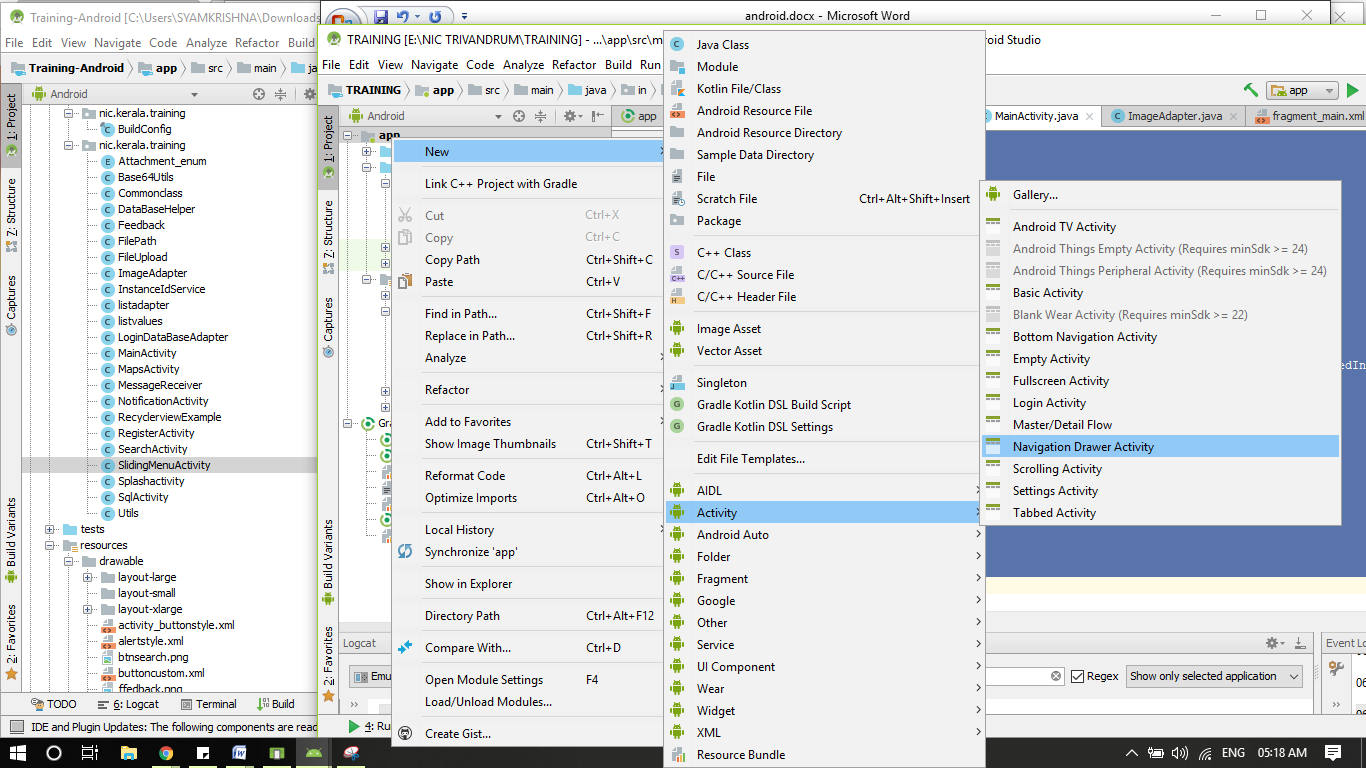
android:theme="@style/AppTheme.NoActionBar"

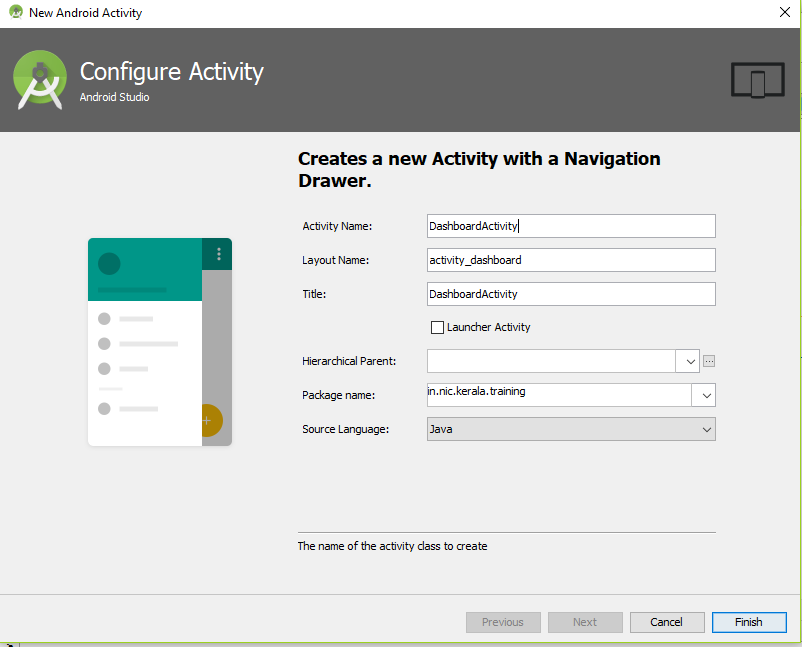
So it looks like

<**activity  
 android:name=".SplashActivity"** android:theme="@style/AppTheme.NoActionBar">  
 <**intent-filter**>  
 <**action android:name="android.intent.action.MAIN"** />  
  
 <**category android:name="android.intent.category.LAUNCHER"** />  
 </**intent-filter**>  
</**activity**>

## Create a Dashboard screen Finally will look like

Create a Navigation Drawer Activity





Provide details as shown in the image.

Change Navigation Drawer contents

Open Res/Menu/activity\_dashboard\_drawer.xml file and remove all contents and paste following contents

*<?***xml version="1.0" encoding="utf-8"***?>*<**menu xmlns:android="http://schemas.android.com/apk/res/android"**>  
  
 <**group android:checkableBehavior="single"**>  
  
 <**item  
 android:id="@+id/search"  
  
 android:title="Search"** />  
  
  
 <**item  
 android:id="@+id/Recyclerview"  
  
  
 android:title="Recyclerview"** />  
 <**item  
 android:id="@+id/fileupload"  
  
 android:title="File Upload"** />  
 <**item  
 android:id="@+id/feed"  
  
  
 android:title="Enter Feedback"** />  
 </**group**>  
 <**item android:title=""**>  
 <**menu**>  
 <**item  
 android:id="@+id/Map"  
  
  
 android:title="Load Map"** />  
  
  
 <**item  
 android:id="@+id/noti"  
  
  
 android:title="Notification"** />  
 <**item  
 android:id="@+id/sql"  
  
  
 android:title="SQL"** />  
  
 </**menu**>  
 </**item**>  
</**menu**>

Update DashboardActivity

Now open DashboardActivity from java folder and update the contents of the function onNavigationItemSelected as

**public boolean** onNavigationItemSelected(MenuItem item) {  
 *// Handle navigation view item clicks here.* **int** id = item.getItemId();  
  
 **if** (id == R.id.***search***) {  
 *// Handle the camera action* } **else if** (id == R.id.***Recyclerview***) {  
  
 } **else if** (id == R.id.***fileupload***) {  
  
 } **else if** (id == R.id.***feed***) {  
  
 } **else if** (id == R.id.***Map***) {  
  
 } **else if** (id == R.id.***noti***) {  
  
 }  
 **else if** (id == R.id.***sql***) {  
  
 }  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 drawer.closeDrawer(GravityCompat.***START***);  
 **return true**;  
}

Add the following variables and functions in the DashboardActivity

**int navItemIndex** = 0;  
**private void** loadHomeFragment() {  
 MainFragment fragment = **new** MainFragment();  
 FragmentTransaction fragmentTransaction = getSupportFragmentManager().beginTransaction();  
 fragmentTransaction.setCustomAnimations(android.R.anim.***fade\_in***,  
 android.R.anim.***fade\_out***);  
 fragmentTransaction.replace(R.id.***frame***, fragment);  
 fragmentTransaction.commitAllowingStateLoss();  
}

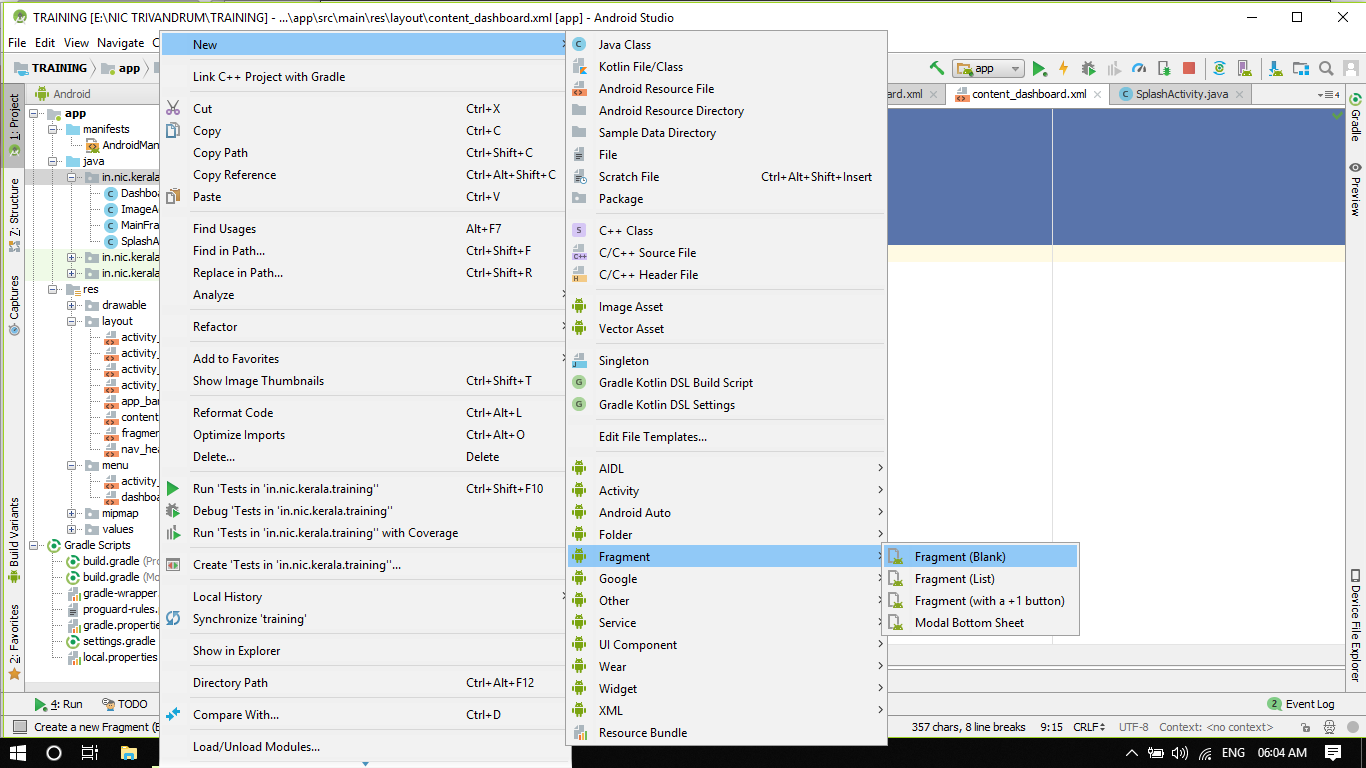
Update content\_dashboard.xml

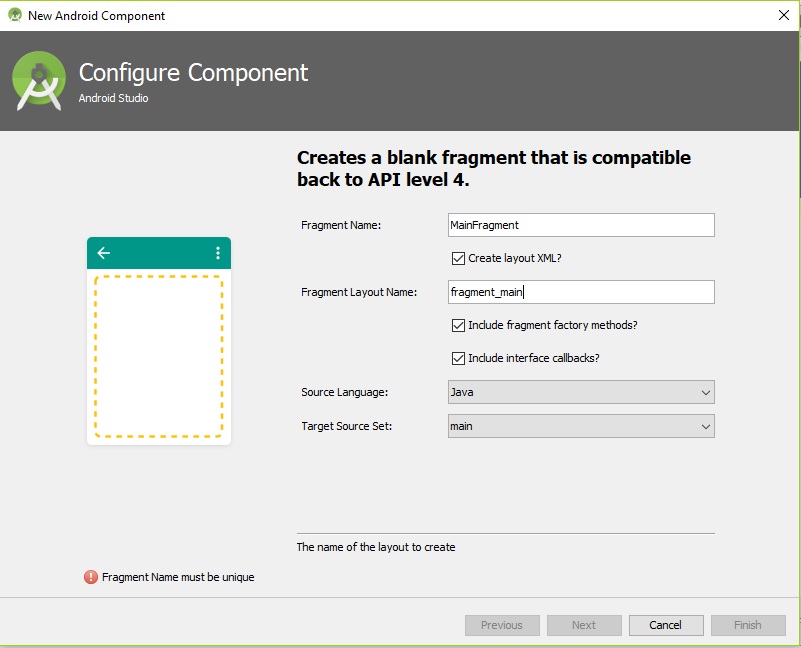
Open content\_dashboard.xml and paste the following code

*<?***xml version="1.0" encoding="utf-8"***?>*<**FrameLayout android:id="@+id/frame"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"**>  
  
</**FrameLayout**>

Now Create a new Fragment

A Fragment represents a behavior or a portion of user interface in an Activity. You can combine multiple fragments in a single activity to build a multi-pane UI and reuse a fragment in multiple activities. You can think of a fragment as a modular section of an activity, which has its own lifecycle, receives its own input events, and which you can add or remove while the activity is running (sort of like a "sub activity" that you can reuse in different activities).





Now Update the contents of MainFragment.java as

**package** in.nic.kerala.training;  
**import** android.support.annotation.Nullable;  
**import** android.support.v4.app.Fragment;  
**import** android.os.Bundle;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.GridView;  
  
**public class** MainFragment **extends** Fragment {  
 GridView **grid**;  
  
 @Nullable  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, @Nullable ViewGroup container, Bundle savedInstanceState) {  
  
 View view = inflater.inflate(R.layout.***fragment\_main***, container, **false**);  
 **grid**=(GridView)view.findViewById(R.id.***grid***);  
 **grid**.setAdapter(**new** ImageAdapter(getActivity()));  
 **return** view;  
 }  
}

Now modify contents of fragment\_main.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainFragment"**>  
 <**include  
 android:id="@+id/ti"  
 android:layout\_height="wrap\_content"  
 android:layout\_width="fill\_parent"  
 layout="@layout/activity\_title"** />  
 <**LinearLayout  
 android:layout\_width="fill\_parent"  
 android:layout\_height="fill\_parent"  
 android:layout\_below="@+id/ti"**>  
 <**GridView  
 android:id="@+id/grid"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:numColumns="2"** />  
 </**LinearLayout**>  
 <**LinearLayout  
 android:id="@+id/in"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="49dp"  
 android:layout\_alignParentBottom="true"  
 android:background="@drawable/fooo"  
 android:orientation="horizontal"** />  
</**RelativeLayout**>

Create a Class Named ImageAdapter.java and Paste following code

This class is used for loading dynamic data to grid. An adapter can feed data to gridview

**package** in.nic.kerala.training;  
  
**import** android.content.Context;  
**import** android.content.Intent;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.BaseAdapter;  
**import** android.widget.ImageView;  
**import** android.widget.ListAdapter;  
**import** android.widget.TextView;  
  
  
**class** ImageAdapter **extends** BaseAdapter {  
  
 **private** Context **context**;  
 **private int** [] **imageId**={R.drawable.***search***,R.drawable.***feedback***,R.drawable.***fileupload***,R.drawable.***map***,R.drawable.***notification***,R.drawable.***sql***};  
 **private** String[]**mobileValues**=**new** String[]{**"Search"**,**"Feedback"**,**"File Upload"**,**"Map"**,**"Notification"**,**"SQL"**};  
 *// Constructor* **public** ImageAdapter(Context c){  
 **this**.**context** = c;  
  
 }  
  
 @Override  
 **public int** getCount() {  
 **return mobileValues**.**length**;  
 }  
  
 @Override  
 **public** Object getItem(**int** position) {  
 **return mobileValues**[position];  
 }  
  
 @Override  
 **public long** getItemId(**int** position) {  
 **return** 0;  
 }  
  
 @Override  
 **public** View getView(**final int** position, View convertView, ViewGroup parent) {  
 LayoutInflater inflater = (LayoutInflater) **context** .getSystemService(Context.***LAYOUT\_INFLATER\_SERVICE***);  
  
 View gridView;  
  
 **if** (convertView == **null**) {  
  
 gridView = **new** View(**context**);  
  
*// get layout from mobile.xml* gridView = inflater.inflate(R.layout.***layout\_custom***, **null**);  
  
*// set value into textview* TextView textView = (TextView) gridView  
 .findViewById(R.id.***grid\_item\_label***);  
 textView.setText(**mobileValues**[position]);  
  
*// set image based on selected text* ImageView imageView = (ImageView) gridView  
 .findViewById(R.id.***grid\_item\_image***);  
  
 imageView.setImageResource(**imageId**[position]);  
 gridView.setOnClickListener(**new** View.OnClickListener() {  
  
 @Override  
 **public void** onClick(View v) {  
  
 */\*  
 if(position==0)  
 {  
 Intent in1 = new Intent(context, SearchActivity.class);  
 context.startActivity(in1);  
  
 }else if(position==1){  
 Intent in2 = new Intent(context, FeedbackActivity.class);  
 context.startActivity(in2);  
  
 }  
 else if(position==2){  
 Intent in2 = new Intent(context, FileUploadActivity.class);  
 context.startActivity(in2);  
  
 }else if(position==3){  
 Intent in2 = new Intent(context, MapActivity.class);  
 context.startActivity(in2);  
  
 }  
 else if(position==4){  
 Intent in4 = new Intent(context, NotificationActivity.class);  
 context.startActivity(in4);  
  
 }  
 else{  
 Intent in4 = new Intent(context, SqlActivity.class);  
 context.startActivity(in4);  
 }  
 \*/* }  
 });  
  
 } **else** {  
 gridView = (View) convertView;  
 }  
  
 **return** gridView;  
 }  
  
}

Now modify create an xml layout layout\_custom.xml under res/layout and paste following code

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:padding="5dp"**>  
  
  
 <**ImageView  
 android:id="@+id/grid\_item\_image"  
 android:layout\_width="80px"  
 android:layout\_height="80px"  
 android:layout\_gravity="center"  
 android:layout\_marginBottom="5dp"  
 android:layout\_marginTop="5dp"**></**ImageView**>  
  
 <**TextView  
 android:id="@+id/grid\_item\_label"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:layout\_marginBottom="5dp"  
 android:layout\_marginTop="5dp"  
 android:text="TextView"  
 android:textSize="15dp"  
 android:textStyle="bold"**></**TextView**>  
  
</**LinearLayout**>

**Android Sliding Menu Using Navigation Drawer**

You might have noticed that lot of android applications introduced a sliding panel menu to navigate between major modules of the application. Previously this kind of UI was done using some third party libraries where a list view and some swiping gestures used to achieve this. But now android itself officially introduced sliding panel menu by introducing a newer concept called Navigation Drawer in which we combine [DrawerLayout](https://developer.android.com/reference/android/support/v4/widget/DrawerLayout.html) and [NavigationView](https://developer.android.com/reference/android/support/design/widget/NavigationView.html) to achieve the desired output. Android studio will create the layout contents for you, which you can override.

**Finally your codes will look like**

**DashboardActivity.java**

**package** in.nic.kerala.training;  
  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.support.design.widget.FloatingActionButton;  
**import** android.support.design.widget.Snackbar;  
**import** android.support.v4.app.FragmentTransaction;  
**import** android.view.View;  
**import** android.support.design.widget.NavigationView;  
**import** android.support.v4.view.GravityCompat;  
**import** android.support.v4.widget.DrawerLayout;  
**import** android.support.v7.app.ActionBarDrawerToggle;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.support.v7.widget.Toolbar;  
**import** android.view.Menu;  
**import** android.view.MenuItem;  
  
**public class** DashboardActivity **extends** AppCompatActivity  
 **implements** NavigationView.OnNavigationItemSelectedListener {  
  
 **int navItemIndex** = 0;  
 **private void** loadHomeFragment() {  
 MainFragment fragment = **new** MainFragment();  
 FragmentTransaction fragmentTransaction = getSupportFragmentManager().beginTransaction();  
 fragmentTransaction.setCustomAnimations(android.R.anim.***fade\_in***,  
 android.R.anim.***fade\_out***);  
 fragmentTransaction.replace(R.id.***frame***, fragment);  
 fragmentTransaction.commitAllowingStateLoss();  
 }  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_dashboard***);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.***toolbar***);  
 setSupportActionBar(toolbar);  
  
 FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.***fab***);  
 fab.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 Snackbar.*make*(view, **"Replace with your own action"**, Snackbar.***LENGTH\_LONG***)  
 .setAction(**"Action"**, **null**).show();  
 }  
 });  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 ActionBarDrawerToggle toggle = **new** ActionBarDrawerToggle(  
 **this**, drawer, toolbar, R.string.***navigation\_drawer\_open***, R.string.***navigation\_drawer\_close***);  
 drawer.addDrawerListener(toggle);  
 toggle.syncState();  
  
 NavigationView navigationView = (NavigationView) findViewById(R.id.***nav\_view***);  
 navigationView.setNavigationItemSelectedListener(**this**);  
  
  
  
 **if** (savedInstanceState == **null**) {  
 **navItemIndex** = 0;  
 loadHomeFragment();  
 }  
 }  
  
 @Override  
 **public void** onBackPressed() {  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 **if** (drawer.isDrawerOpen(GravityCompat.***START***)) {  
 drawer.closeDrawer(GravityCompat.***START***);  
 } **else** {  
 **super**.onBackPressed();  
 }  
 }  
  
 @Override  
 **public boolean** onCreateOptionsMenu(Menu menu) {  
 *// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.***dashboard***, menu);  
 **return true**;  
 }  
  
 @Override  
 **public boolean** onOptionsItemSelected(MenuItem item) {  
 *// Handle action bar item clicks here. The action bar will  
 // automatically handle clicks on the Home/Up button, so long  
 // as you specify a parent activity in AndroidManifest.xml.* **int** id = item.getItemId();  
  
 *//noinspection SimplifiableIfStatement* **if** (id == R.id.***action\_settings***) {  
 **return true**;  
 }  
  
 **return super**.onOptionsItemSelected(item);  
 }  
  
 @SuppressWarnings(**"StatementWithEmptyBody"**)  
 @Override  
 **public boolean** onNavigationItemSelected(MenuItem item) {  
  
  
  
  
*// Handle navigation view item clicks here.* **int** id = item.getItemId();  
  
 **if** (id == R.id.***search***) {  
 **Intent i=new Intent(DashboardActivity.this,SearchActivity.class);  
 startActivity(i);**  
  
  
  
 }  
 **else if** (id == R.id.***Recyclerview***) {  
  
 **Intent i=new Intent(DashboardActivity.this,RecyclerviewExample***Activity***.class);  
 startActivity(i);**  
  
  
 }  
 **else if** (id == R.id.***feed***) {  
  
 **Intent i=new Intent(DashboardActivity.this,Feedback***Activity***.class);  
 startActivity(i);**  
 }  
 **else if** (id == R.id.***fileupload***) {  
 **Intent i=new Intent(DashboardActivity.this,FileUpload***Activity***.class);  
 startActivity(i);**  
  
  
 }  
 **else if** (id == R.id.***Map***) {  
 **Intent i=new Intent(DashboardActivity.this,MapsActivity.class);  
 startActivity(i);**  
  
  
 }  
 **else if** (id == R.id.***sql***) {  
  
 **Intent i=new Intent(DashboardActivity.this,SqlActivity.class);  
 startActivity(i);**  
 }  
 **else if** (id == R.id.***noti***) {  
 **Intent i=new Intent(DashboardActivity.this,NotificationActivity.class);  
 startActivity(i);**  
  
  
 }  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 drawer.closeDrawer(GravityCompat.***START***);  
 loadHomeFragment();  
 **return true**;  
  
  
  
 }  
}

Here we are yet to create other activities (shown in red color), so they may show errors.

app\_bar\_dashboard.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".DashboardActivity"**>  
  
 <**android.support.design.widget.AppBarLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:theme="@style/AppTheme.AppBarOverlay"**>  
  
 <**android.support.v7.widget.Toolbar  
 android:id="@+id/toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="?attr/actionBarSize"  
 android:background="?attr/colorPrimary"  
 app:popupTheme="@style/AppTheme.PopupOverlay"** />  
  
 </**android.support.design.widget.AppBarLayout**>  
  
 <**include layout="@layout/content\_dashboard"** />  
  
 <**android.support.design.widget.FloatingActionButton  
 android:id="@+id/fab"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="bottom|end"  
 android:layout\_margin="@dimen/fab\_margin"  
 app:srcCompat="@android:drawable/ic\_dialog\_email"** />  
  
</**android.support.design.widget.CoordinatorLayout**>

content\_dashboard.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**FrameLayout android:id="@+id/frame"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"**>  
  
</**FrameLayout**>

nav\_header\_dashboard.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="@dimen/nav\_header\_height"  
 android:background="@drawable/side\_nav\_bar"  
 android:gravity="bottom"  
 android:orientation="vertical"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 android:theme="@style/ThemeOverlay.AppCompat.Dark"**>  
  
 <**ImageView  
 android:id="@+id/imageView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:contentDescription="@string/nav\_header\_desc"  
 android:paddingTop="@dimen/nav\_header\_vertical\_spacing"  
 app:srcCompat="@mipmap/ic\_launcher\_round"** />  
  
 <**TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:paddingTop="@dimen/nav\_header\_vertical\_spacing"  
 android:text="@string/nav\_header\_title"  
 android:textAppearance="@style/TextAppearance.AppCompat.Body1"** />  
  
 <**TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/nav\_header\_subtitle"** />  
  
</**LinearLayout**>