Experiment No. - 9

Student Name: Aman Gokul
UID: 20BCS5449
Branch: BE-CSE
Section/Group: 20BCS-607/A
Semester: 6th
Date of Performance: 03/05/2023
Subject Name: Mobile Application Development Lab
Subject Code: 20CSP-356

1. **Aim:**
   Design the Android application using menus and action bar.
2. **Objective:**
   Understanding and analyse the specific requirement, possibilities and challenges when developing for a mobile application context.

3. **System Requirements:**
   - Microsoft Windows 7/8/10 (32-bit or 64-bit)
   - 4 GB RAM minimum, 8 GB RAM recommended (plus 1 GB for the Android Emulator)
   - 2 GB of available disk space minimum, 4 GB recommended (500 MB for IDE plus 1.5 GB for Android SDK and emulator system image)
   - 1280 x 800 minimum screen resolution
   - Java JDK5 or later version
   - Java Runtime Environment (JRE) 6 or higher.
4. **Steps/Program:**
   - You will use Android Studio IDE to create an Android application and name it as tutorials point under a package in.innovateria.wa9application
   - Modify src/MainActivity.java and add the appropriate code for the drawer and option menu.
   - Override the 2 Boolean method for option menu implementation such as onCreateOptionsMenu() and onOptionsItemSelected().
   - Modify layout XML file res/layout/activity_main.xml add any GUI component if required. I'm adding a simple GUI to implement all type of Layout such as linear, relative, absolute.
   - No need to define default string constants at res/values/strings.xml. Android studio takes care of default constants.
   - No need to Modify AndroidManifest.xml and Add the Permissions.
   - Run the application to launch Android emulator and verify the result of the changes done in the application.

5. **Code:**

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.drawerlayout.widget.DrawerLayout 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=".MainActivity"
    android:fitsSystemWindows="true"
```

```
    tools:openDrawer="start"
    android:id="@+id/drawer">

    <include layout="@layout/toolbar"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>

    <com.google.android.material.navigation.NavigationView
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:layout_gravity="start"
        android:fitsSystemWindows="true"
        android:id="@+id/navigationView"
        app:headerLayout="@layout/header"
        app:menu="@menu/drawer_menu"/>

</androidx.drawerlayout.widget.DrawerLayout>
```

MainActivity.java

```
package in.innovateria.wa9application;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;

import android.os.Bundle;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.Toast;

import com.google.android.material.navigation.NavigationView;

import in.innovateria.wa9application.Fragments.FirstFragment;
import in.innovateria.wa9application.Fragments.SecondFragment;

public class MainActivity extends AppCompatActivity {
    DrawerLayout drawerLayout;
    NavigationView navigationView;
    Toolbar toolbar;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        drawerLayout = findViewById(R.id.drawer);
        navigationView = findViewById(R.id.navigationView);
        toolbar = findViewById(R.id.toolbar);

        setSupportActionBar(toolbar);

        ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(this, drawerLayout, toolbar,
R.string.open, R.string.close);

        drawerLayout.addDrawerListener(toggle);

        toggle.syncState();

        getSupportFragmentManager().beginTransaction()
                .add(R.id.nav_host_fragment, new FirstFragment())
                .commit();

        navigationView.setNavigationItemSelectedListener(new NavigationView.OnNavigationItemSelectedLis-
tener() {
            @Override
            public boolean onNavigationItemSelected(@NonNull MenuItem item) {
                int id = item.getItemId();
                if (id == R.id.home) {
                    loadFragment(new FirstFragment());
                } else if (id == R.id.account) {
                    loadFragment(new SecondFragment());
                }
                drawerLayout.closeDrawer(GravityCompat.START);

                return true;
            }
        });
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        new MenuInflater(this).inflate(R.menu.option_menu,menu);
        return super.onCreateOptionsMenu(menu);
    }

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        int itemId = item.getItemId();
        if(itemId==R.id.notification){
            Toast.makeText(this, "Notification is Selected", Toast.LENGTH_SHORT).show();
        }else if(itemId==R.id.close){
```

```
            Toast.makeText(this, "Close is Selected", Toast.LENGTH_SHORT).show();
        }else if(itemId==R.id.save){
            Toast.makeText(this, "Save is Selected", Toast.LENGTH_SHORT).show();
        }
        return super.onOptionsItemSelected(item);
    }

    private void loadFragment(Fragment fragment) {
        FragmentManager fm = getSupportFragmentManager();
        FragmentTransaction ft = fm.beginTransaction();
        ft.add(R.id.nav_host_fragment, fragment);
        ft.commit();
    }
}
```

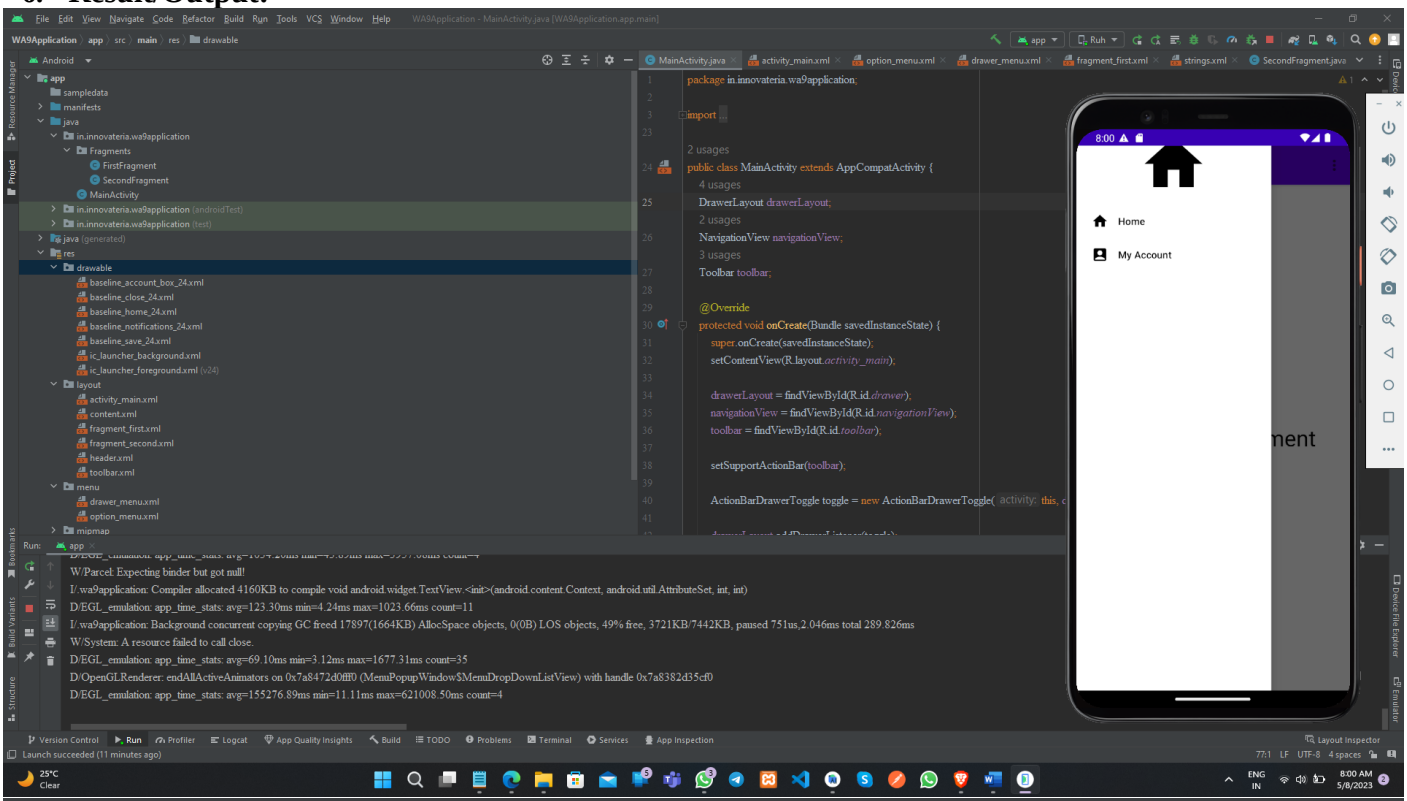
drawer_menu.xml

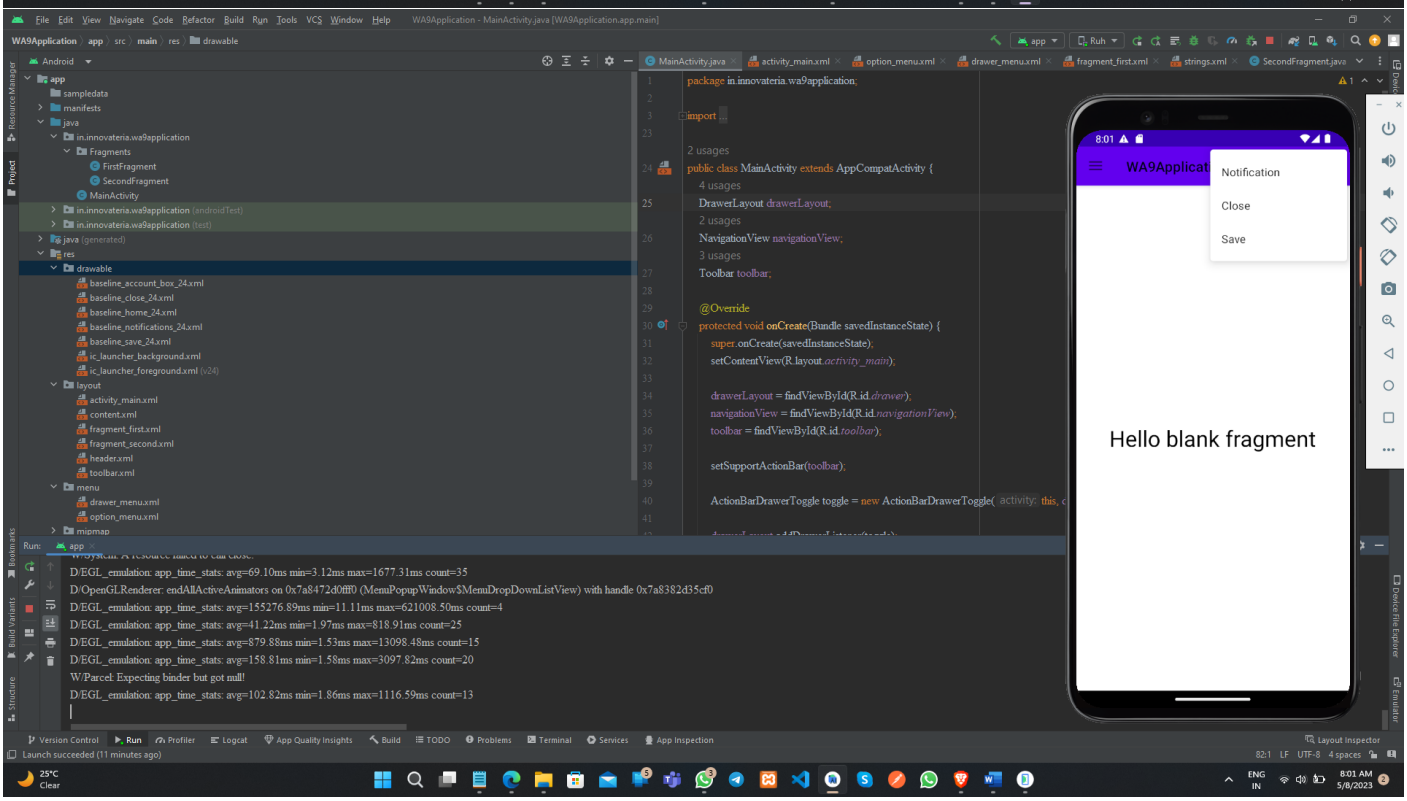
```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <item
        android:id="@+id/home"
        android:icon="@drawable/baseline_home_24"
        android:title="Home" />
    <item
        android:id="@+id/account"
        android:icon="@drawable/baseline_account_box_24"
        android:title="My Account" />
</menu>
```

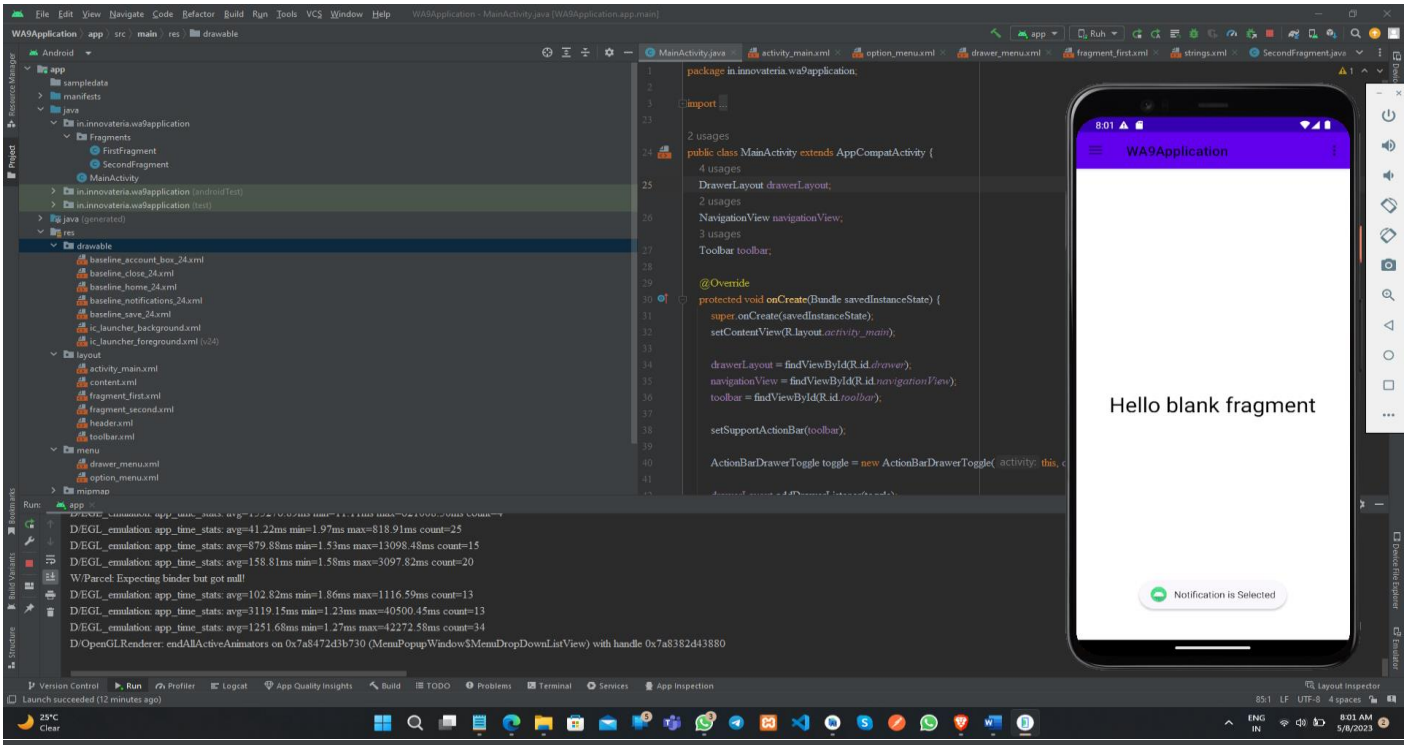
option_menu.xml

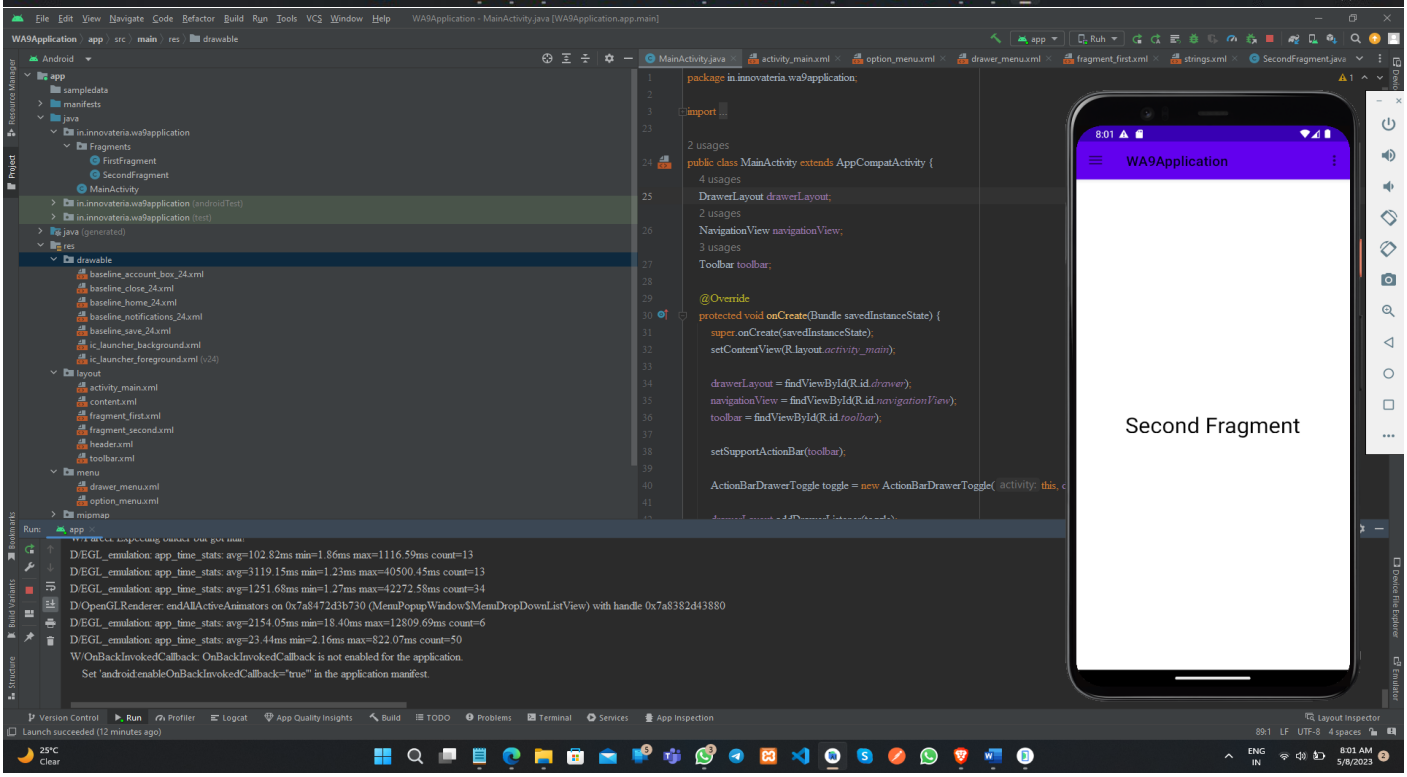
```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <item
        android:id="@+id/notification"
        android:icon="@drawable/baseline_notifications_24"
        android:title="Notification"
        app:showAsAction="ifRoom"/>
    <item
        android:id="@+id/close"
        android:icon="@drawable/baseline_close_24"
        android:title="Close"
        app:showAsAction="collapseActionView"/>
    <item
        android:id="@+id/save"
        android:icon="@drawable/baseline_save_24"
        android:title="Save"
        app:showAsAction="collapseActionView"/>
</menu>
```

6. Result/Output:









Learning outcomes (What I have learnt):

- To design an android application which uses Drawer and Option in android studio.
- Learnt about running application on android studio.
- Creating Application by Implementing Drawer and Option Menu App.