

ICP8

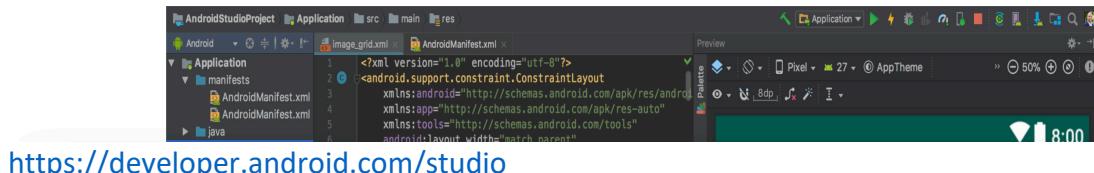
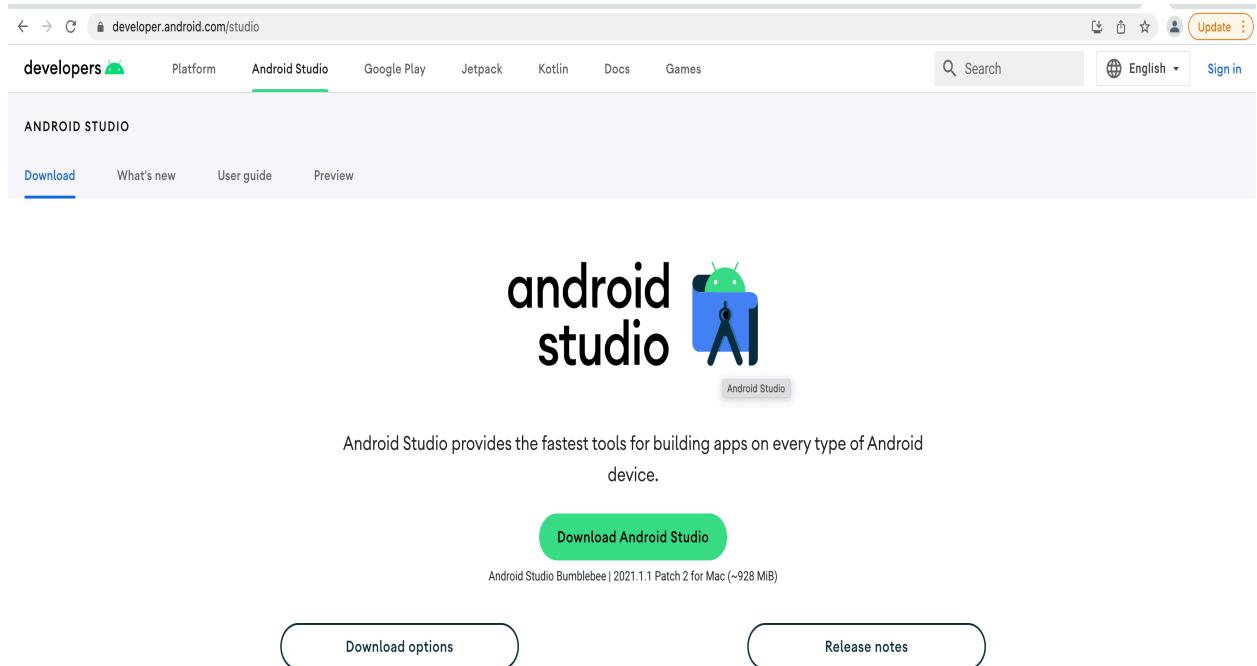
CHENNA SRINIVAS – 16307995

https://github.com/sc7vc/Web_Applications/tree/main/Web/ICP8

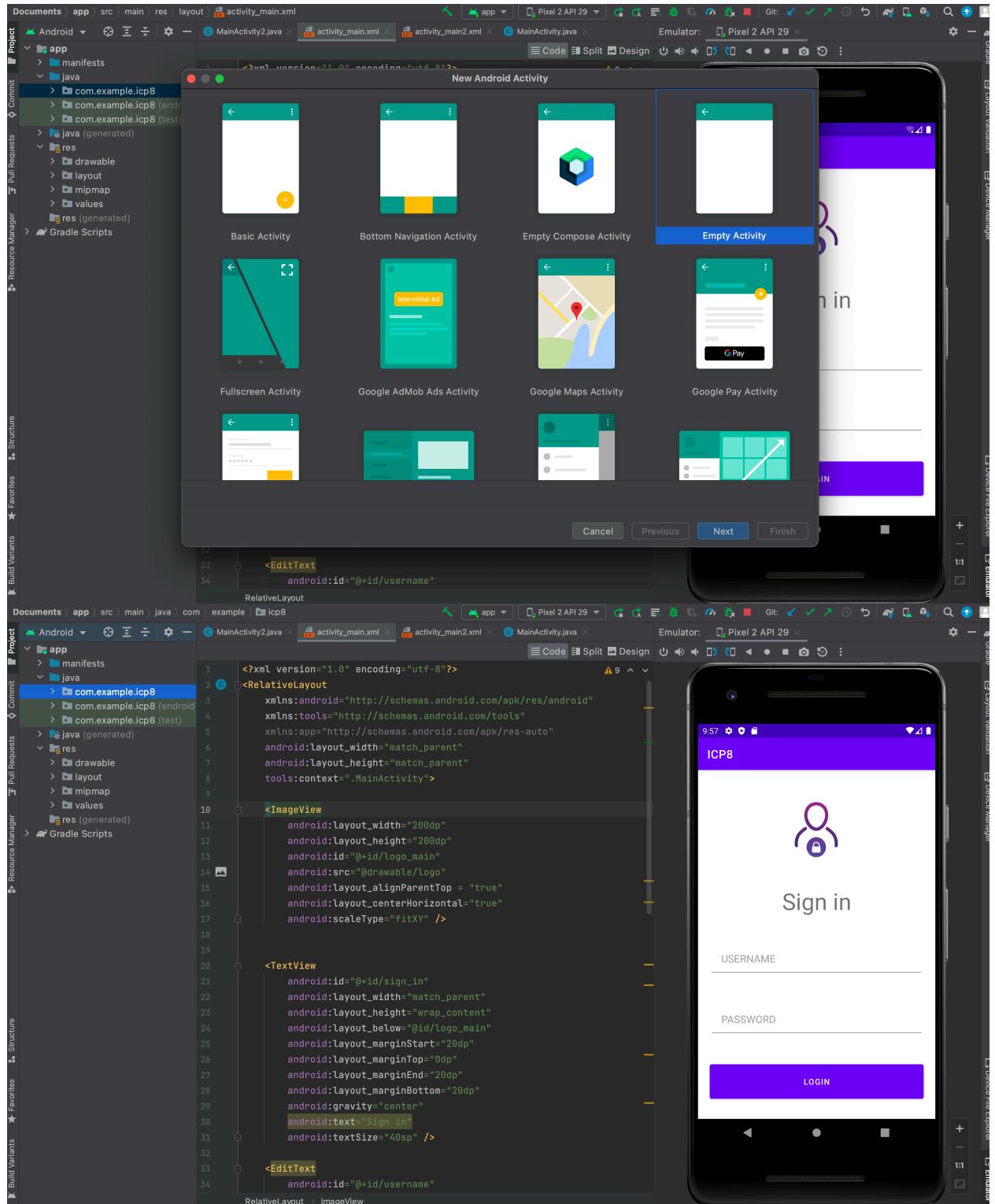
MAMJALA MANISHA – 16307984

https://github.com/mmg6m/Web/tree/main/Web_ICP/ICP8

1. First download Android Studio from below website and install it



2. After installing the studio create a new project by selecting Java as the language and select an empty activity and select a Google Pixel2 phone with android 11 version.



3. Now write the structure in activity_main.xml as below

The screenshot shows the Android Studio interface with the XML code for `activity_main.xml` in the center. The code defines a `RelativeLayout` containing a `ImageView` for the logo and a `TextView` for the sign-in button. Below the `TextView` is an `EditText` for the username. To the right, a smartphone emulator displays the sign-in screen with a purple header, a logo icon, and the text "Sign in". It also shows fields for "USERNAME" and "PASSWORD" and a purple "LOGIN" button.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <ImageView
        android:layout_width="200dp"
        android:layout_height="200dp"
        android:id="@+id/logo_main"
        android:src="@drawable/logo"
        android:layout_alignParentTop = "true"
        android:layout_centerHorizontal="true"
        android:scaleType="fitXY" />

    <TextView
        android:id="@+id/sign_in"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/logo_main"
        android:layout_marginStart="20dp"
        android:layout_marginTop="0dp"
        android:layout_marginEnd="20dp"
        android:layout_marginBottom="20dp"
        android:gravity="center"
        android:text="Sign in"
        android:textSize="40sp" />

    <EditText
        android:id="@+id/username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"/>

```

4. Now write the user specific conditions as below

The screenshot shows the Java code for `MainActivity.java` in the center. It contains an `onClick` listener for a button. The code checks if both the username and password are empty and shows a toast message. If either field is empty, it shows a different toast message. If both are valid (both are "admin"), it starts an activity and shows a success toast. If the username is invalid ("User"), it shows another error toast. The right side shows the same sign-in screen as before, with the "Sign in" button now being purple.

```
    indViewById(R.id.password);
    indViewById(R.id.btn);
    OnClickListener( new View.OnClickListener() {
        @Override
        void onClick(View v) { //Username and Password both are missing
            if(usn.getText().toString().isEmpty() && psw.getText().toString().isEmpty())
                Toast.makeText( context: MainActivity.this, text: "Please provide valid credentials"
            } else {
                if ((usn.getText().toString().equals("admin")) && (psw.getText().toString().equals("admin")))
                    Intent intent = new Intent( packageContext: MainActivity.this, Main.class );
                    startActivity(intent);
                    Toast.makeText( context: MainActivity.this, text: "Successfully Signed In"
                } else { //Invalid Credentials
                    if (!usn.getText().toString().equals("admin"))
                        if (!psw.getText().toString().equals("admin"))
                            Toast.makeText( context: MainActivity.this, text: "Username or Password is Invalid"
                        } else {
                            Toast.makeText( context: MainActivity.this, text: "Password is Invalid"
                        }
                }
            }
        }
    });

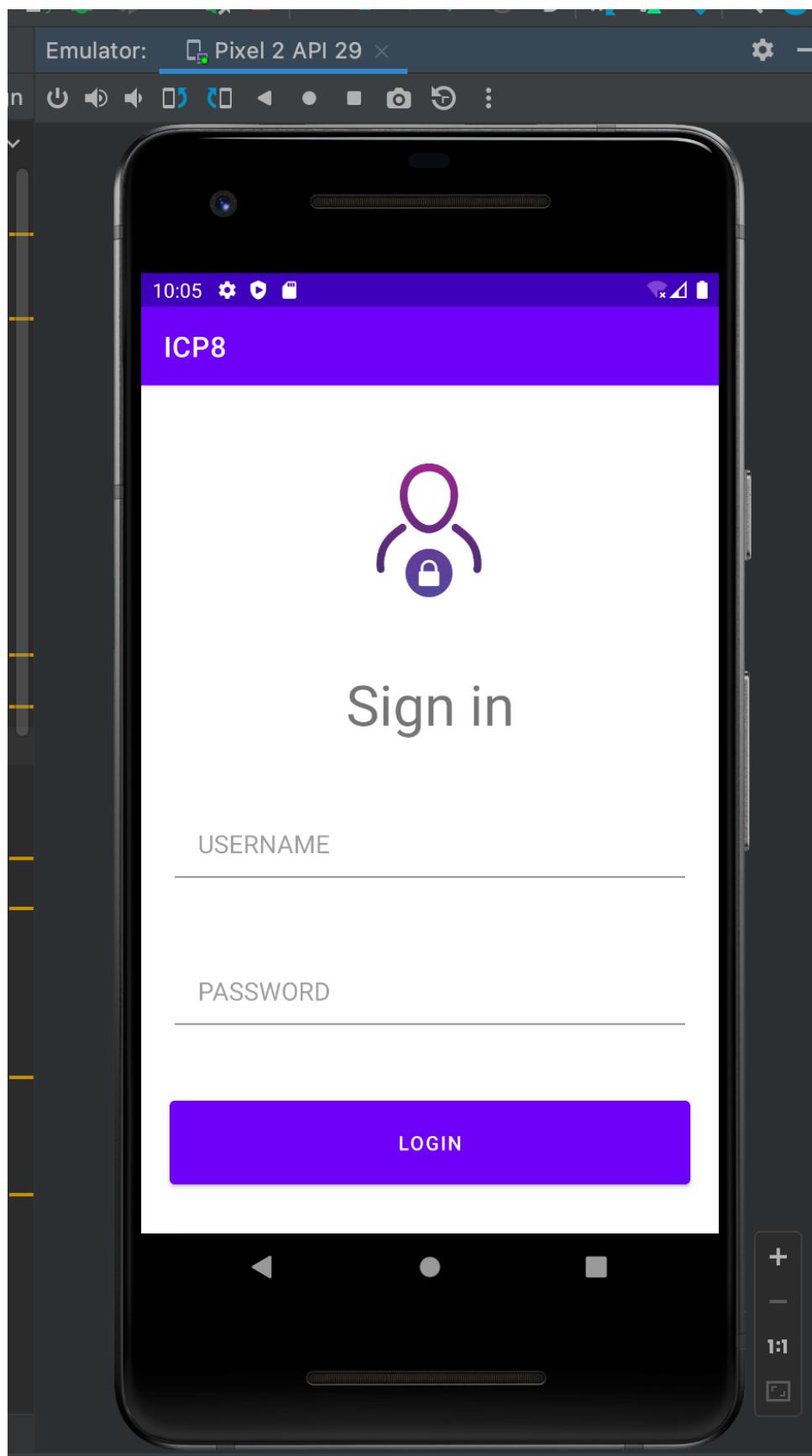
```

5. We can include a logo as below

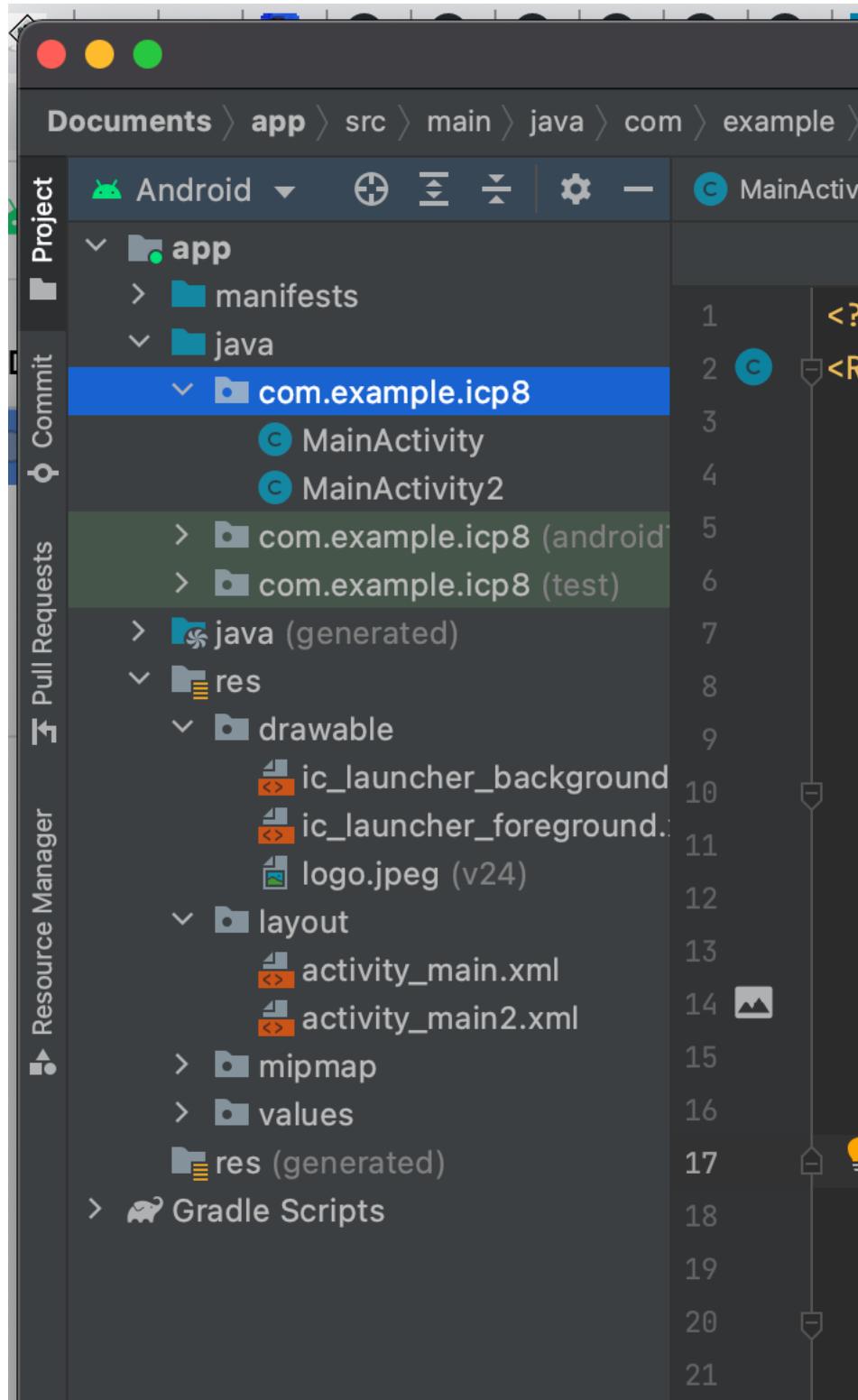
```
<ImageView  
    android:layout_width="200dp"  
    android:layout_height="200dp"  
    android:id="@+id/logo_main"  
    android:src="@drawable/logo"  
    android:layout_alignParentTop = "true"  
    android:layout_centerHorizontal="true"  
    android:scaleType="fitXY" />
```

Here @drawable is the location and logo is our logo name

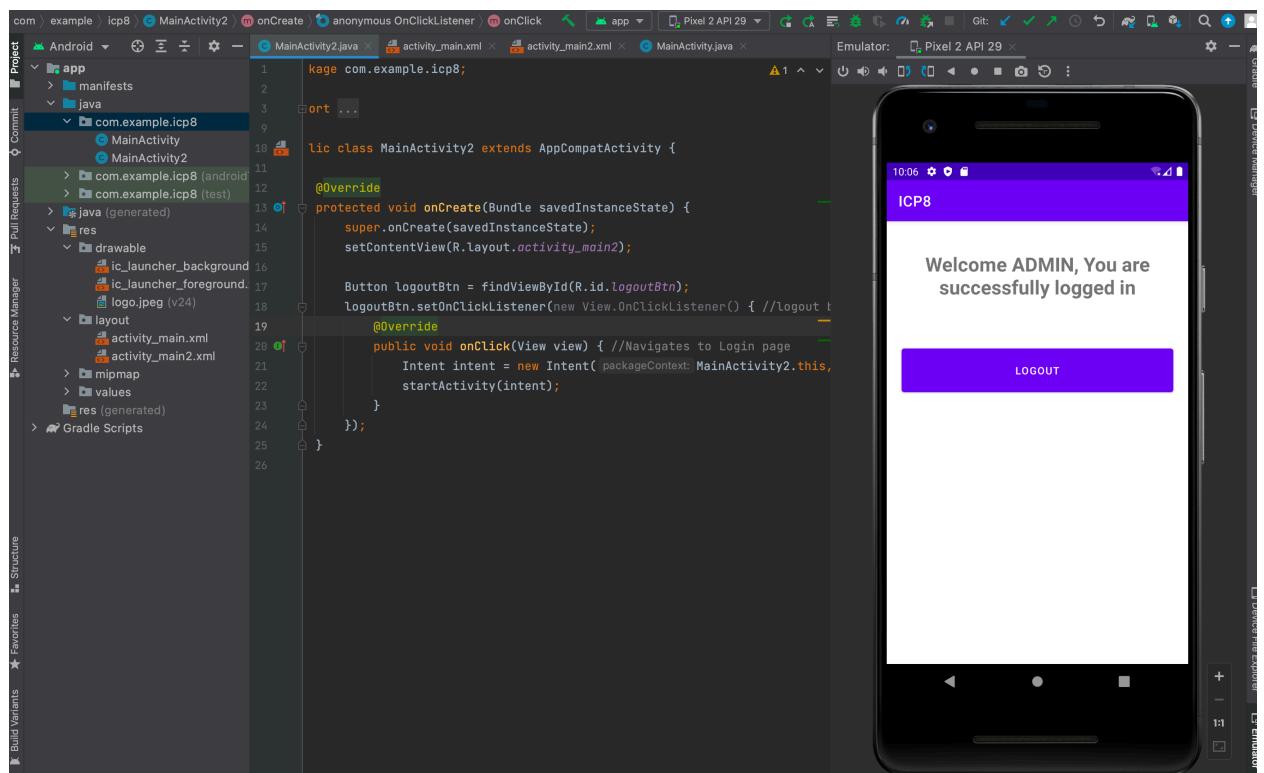
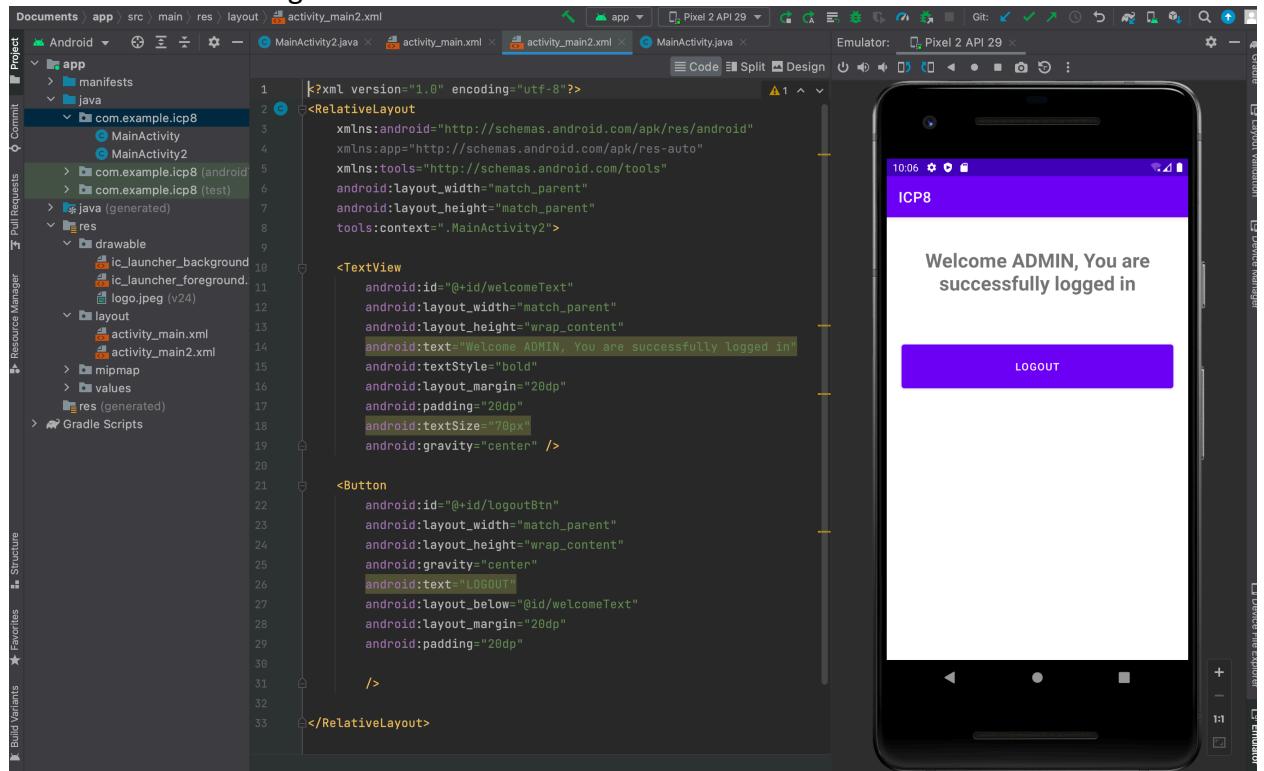
6. Now our Login page is created as below



7. Now let's create the logout page , in which we need to take new blank activity, then our structure will look like below



8. Now lets write our Logout function as below



9. Now we need to link the Login page to Logout page. When user clicks on Login, the user should navigate to Logout page as below

```
if ((usn.getText().toString().equals("admin")) && (psw.getText().toString().equals("admin"))){  
    Intent intent = new Intent( packageContext: MainActivity.this, MainActivity2.class);  
    startActivity(intent);
```

10. Now lets link Logout page to Login where when user clicks Logout, user will be navigated to Login page

```
public void onClick(View view) { //Navigates to Login page  
    Intent intent = new Intent( packageContext: MainActivity2.this, MainActivity.class);  
    startActivity(intent);
```

11. Final Output:-

