MOBILE APPLICATION DEVELOPMENT LABORATORY

PRACTICAL REPORT FILE

Submitted in partial fulfillment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

(Information Technology)

Submitted By: **Submitted To:**
Name : Kishan Tiwari   **Prof.** Ranjodh Kaur
URN: 1905354
CRN: 1921056

GURU NANAK DEV ENGINEERING COLLEGE
LUDHIANA - 141006 , INDIA

Contents

1 To study design aspects of development environment like Android, iOS.

1.1 Android Studio

Android is an open source and Linux-based Operating System for mobile devices such as smart-phones and tablet computers. Android was developed by the Open Handset Alliance, led by Google, and other companies.
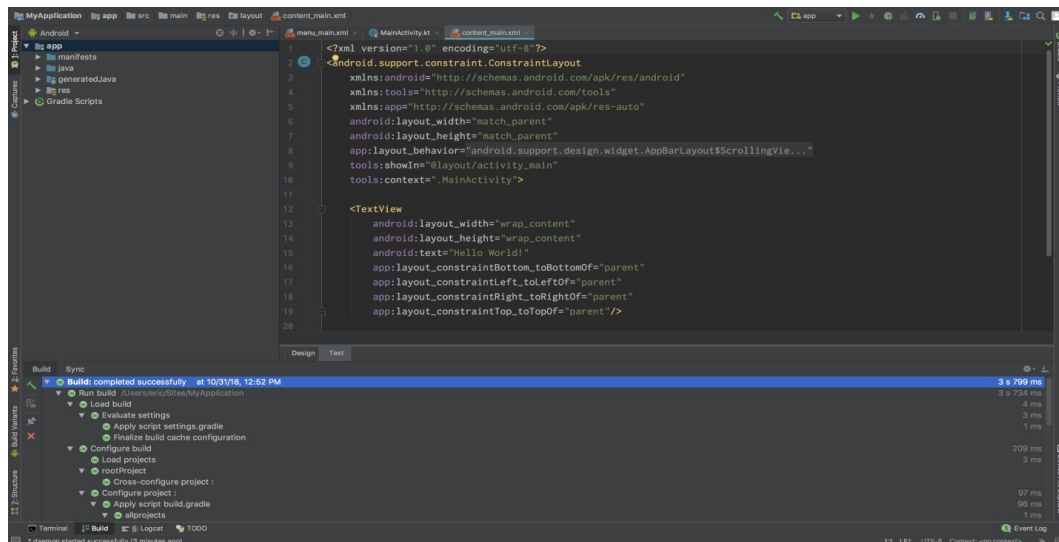
Figure 1: Android Studio

To start coding an Android App, you need an Android development environment. Android Studio is an IDE dedicated to creating and maintaining Android apps Fortunately, Alex did a great job explaining the ins and outs of how it works. As a designer, you're going to want to get comfortable working in the following files:

- **Colors.xml**
  Much like Sass variables or CSS Custom Properties, you're going to want to abstract as much of your code as possible. Regardless of coding language or medium, this kind of practice helps keep your code consistent, flexible, and easy to maintain.In Android, colors.xml is used to specify all your app's color values. For example, a color called sunset would described like this: FA5A41.After turning their palette into something Android-friendly, we can now use it in the various pieces that make up the app. Access the contents of colors.xml by first typing @color/, then the color's name:

  ```
      android:id="@+id/background"``
  android:background="@color/grayLight"``
  ``
  />
  ```

- **Dimens.xml**

  We want to abstract our measurements, much like with our color values. In Android, dimens.xml is where we place that code. Density-independent pixels .Android has a unit called density-independent pixels (dp). It's a technical solution that plays to the strengths of their many devices having different display sizes and densities. Unlike declaring a regular static pixel value, a device pixel declaration tells the Android operating system to draw the pixels making up a UI proportionately based on a device's reported display properties.

  1. Scale-independent pixels

  Scale-independent pixels (sp) are much like density-independent pixels, only they are used for type. The main difference is that they scale along with a user's font size preference. This is great for helping to make your app accessible for those who benefit from a larger type size.

  2. Abstraction

  The client's design system used a suite of standardized spacing values (10px, 20px, 30px, etc.). Since Sketch pixels don't directly translate to Android density-independent pixels, the trick is to abstract these Sketch measurements and codify them as a series of distances:
  2dp
  4dp
  8dp
  10dp
  12dp

  3. Start from the top

  There's a trick in web design where you consistently apply spacing to only the top or bottom of all your interface elements. The idea is it makes a consistent appearance easier, as you don't have to constantly undo and redo your margin declarations and wreck the cascade. Personally, I prefer applying spacing from the top, as an infinite vertically-scrolling canvas means there's always going to be more room below.

- **Styles.xml**

  Styling components in Android is a highly declarative process. There isn't a real cascade, like what you'd get with CSS, meaning that you're going to spend a lot of time re-describing things like color and fonts between different interface components.

```
<style name="InputLabel" parent="Label"> ` ``
@font/lato_bold``` ` ``@color/sunset``` ` `
`true``` ` ``18sp``` </style>``
This collection of styling instructions can
then be via a declaration of style:
<TextView``
```

```
android:id="@+id/labelName"``
android:layout_marginTop="@dimen/spacing_small"``
style="@style/InputLabel"``
``
/>
```

- **Strings.xml**

  strings.xml is used to contain the content of your app. If you're picking up on a pattern here, it's that this is yet another centralizing file that controls things. Strings are declared in a fashion similar to colors and measurement: Login as an admin. They are declared by typing android:text="", then @string/ inside the double quotes, then the string's name:

```
<Button``
android:id="@+id/button_login_as_admin"``
android:layout_marginTop="@dimen/spacing_largest"``
android:text="@string/login_as_admin"``
style="@style/ButtonPrimary"``
``
/>
```

1. Commenting and prefixing
Keeping your app code organized helps you quickly locate things and avoid writing duplicate code. This is especially important as the scope of your app grows.
As more code is added to the repo, what we can do is liberally apply comments to help break up the walls of code. I personally like to use multiline comments to separate the larger sections, to more quickly identify what's what when scrolling through the page.

2. Get a phone
Sketch and Android Studio can't directly communicate, meaning that there's the opportunity for drift to occur between your design files and the actual implementation. Android studio has two tools to help combat this: a Design view and a virtual device emulator.

3. Design view and emulated Android devices
The Design view gives you an in-pane preview of the code you're writing for a specific view. It's good for a quick check, to see if the layout you coded appears properly. However, I've found that when it comes to reviewing fine details, Android Studio simply can't hack it.

4. Physical hardware
As the saying goes, nothing beats the real thing. On a decent computer, the build time for compiling to an emulator or a physical device was basically the same.

5. Wrapping up
The client met their deadline, and thanks to Alex's help, I got to learn some new skills.

Excellent all around! If you're a designer finding yourself undertaking a similar challenge, hopefully this advice can help you out as well.

1.2 IOS

IOS is a mobile operating system for Apple-manufactured devices. iOS runs on the iPhone, iPad, iPod Touch and Apple TV.IOS is best known for serving as the underlying software that allows iPhone users to interact with their phones using gestures such as swiping, tapping and pinching. These finger actions are typically performed on multitouch capacitive touch screen displays, which provide fast response and accept inputs from multiple fingers.
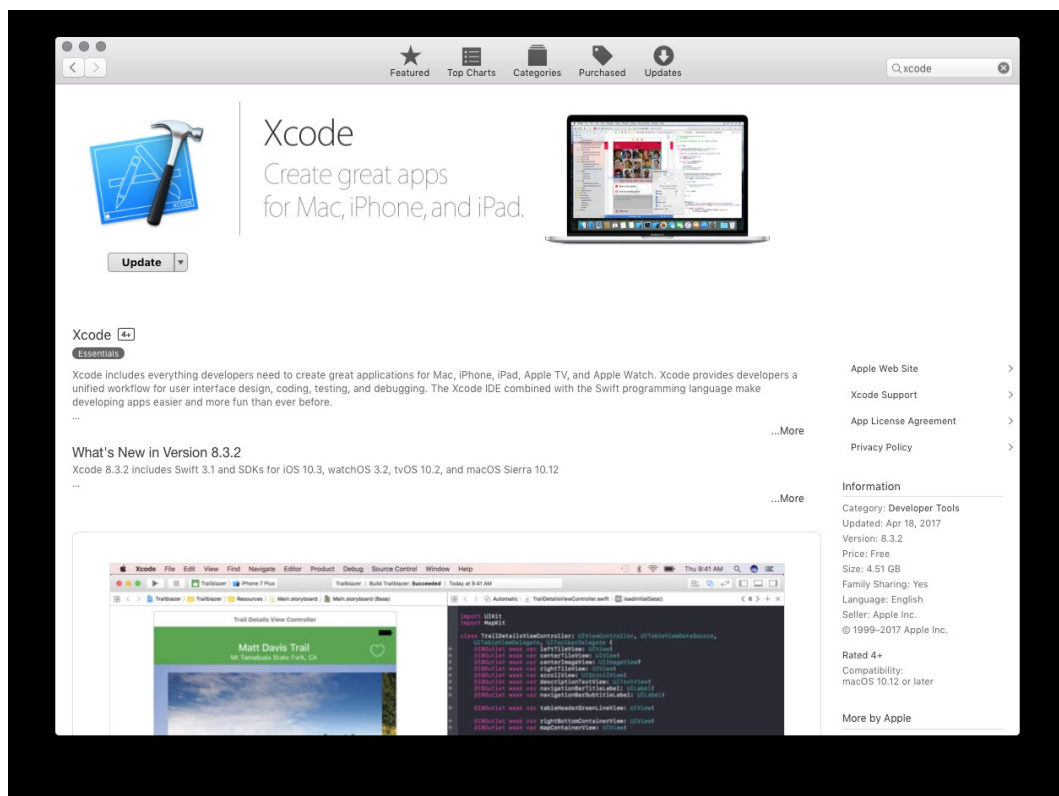
Figure 2: IOS Interface

1. The iPhone X safe area
The iPhone X requires some special attention from designers. The width of the screen is the same as on the iPhone 6, 7 and 8, but it's also 145pt taller. When designing apps for the iPhone X you have to be aware of the safe zone and layout margins to ensure your app user interface doesn't get clipped by the device's sensors or corners.

2. Automatically applied effects
App icons assets are generally added to the application package as plain, squared PNG files in various dimensions. When rendered on a device, iOS applies various effects to app icons.

- **Rounded Corners**

  The old simple radii values for rounded corners are gone. Since iOS 7, app icons have been using the shape of a superellipse. When designing app icons for iOS, I recommend to use the official app icon templates provided by Apple.

- **Border stroke (in some situations)**
  If the app icon you are using has a white background, a 1 pixel gray border stroke will be applied to make it easier to recognize the edges of the icon. This is only done in the settings app (if your application is listed there) and the AppStore.

3. Legacy effects (iOS 6 and previous versions)
On older iOS versions, these effects are applied automatically: rounded corners (not the same shape as iOS 7+ icons are using), drop shadows on the home screen and a gloss effect that can be disabled.

4. Keyboards
There are various keyboard types available to provide the best possible keyboard for a specific text input. While it is possible to build your own completely custom keyboard, default keyboards cannot be customized in style or size.

Figure 3: Grid System

Apple developed a golden ratio grid system that can be used to size and align elements on your icon correctly. Nevertheless, even Apple designers are not following the grid system very strictly with the native apps' icons. Feel free to break the rules if your icon simply works better without aligning all elements strictly to the grid.

5. Typography
The default system font on all iOS versions previous iOS 9 is Helvetica Neue. With the release of iOS 9, Apple introduced a brand new font called San Francisco, which replaced Helvetica Neue as the default font. San Francisco comes in two shapes: SF UI Display" and SF UI Text", while Display" is primarily used for UI components, Text" features a wider letter spacing and should be used for longer texts. You can download the San Francisco fonts here if you are a member of Apple's Developer program. In addition to the default font, many alternative font

faces are available to use. You can find a complete list of pre-installed typefaces here.

6. Toolbar
A toolbar contains a set of actions for managing or manipulating the content of the current view. On the iPhone, it will always appear aligned at the bottom edge of the screen, while on the iPad, it can also be displayed aligned at the top of the screen.

Similarly to the navigation bar, the background fill of toolbars can be modified, is translucent and blurs the underlaying content by default.

7. Search Bar
Search bars come in two different styles by default: prominent and minimal. Both versions do have the same functionality.
As long as no text was entered by the user, a placeholder text is shown inside the bar, and, optionally, a bookmarks icon that can be used to access recent or saved searches.
Once a search term is entered, the placeholder disappears, and a clear button to delete the entered value appears on the right edge.

8. Table View
Table views are used to display small to large amounts of list style information in a single or multiple columns and with the option to divide several rows into separate sections or to group them.
There are two basic table view types that should be used, depending on the type of data you are presenting.

9. Alerts
The purpose of alerts is to inform the user about critical information and optionally to force the user to make a decision about some action.
An alert view does always contain a title text, which should not be longer than one line and one (for pure informational alerts, e.g., OK") or two (for alerts that require a decision, e.g., Send" and Cancel") buttons.

10. Modals
Modals are a useful view for tasks that require multiple commands or inputs by the user. They appear on top of everything else, and, while open, block interaction with any other interactive elements underneath.
The typical modal usually provides:

- a title to describe the task;

- a button to close the modal without saving or performing any other actions;

- a button to save or submit any entered information

2 To setup Android studio2 and study its basic components.

2.1 Steps to set up Android Studio:-

Step-1 Head over to this link https://developer.android.com/studio/downloads to get the Android Studio executable or zip file .

Step-2 Click on the download android studio button .
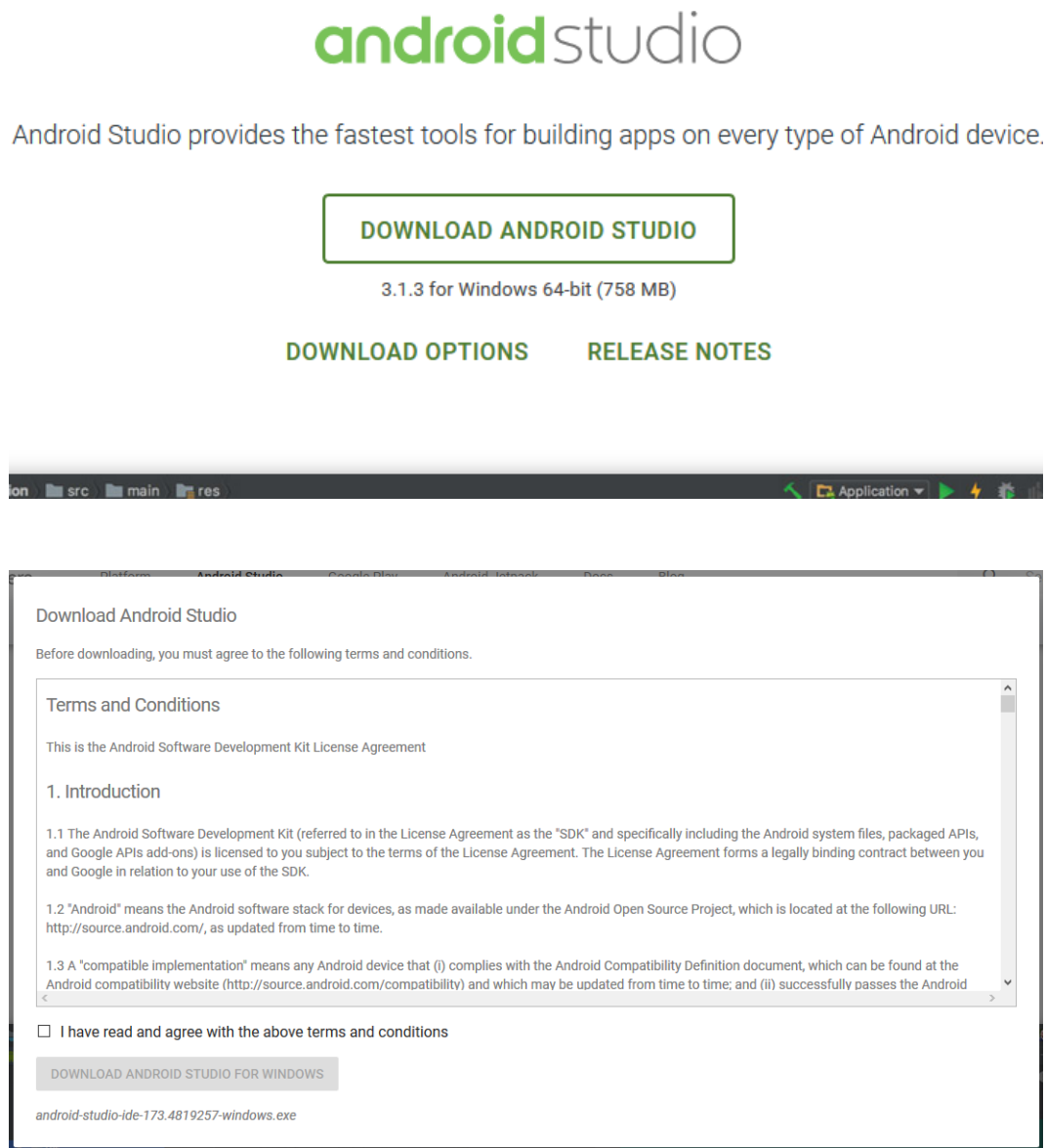
Click on Save file button in the appeared prompt box and the file will start downloading .

Welcome to Android Studio Setup

Setup will guide you through the installation of Android Studio.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.
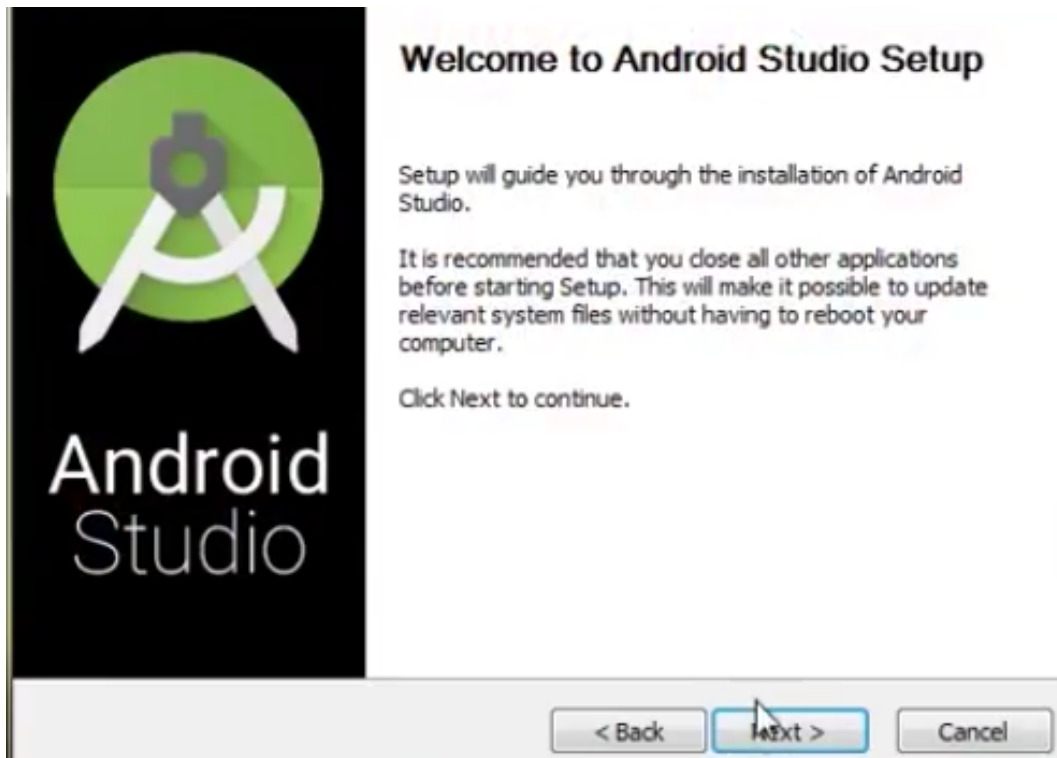
Step-3 After the downloading has finished, open the file from downloads and run it . It will prompt the following dialogue box .

In the next prompt it'll ask for a path for installation. Choose a path and hit next.
Note :The installation path should have the required minimum space.

Step-4 It will start the installation, and once it is completed, it will be like the image shown below .

Installation Complete
Setup was completed successfully.

Completed

Show details
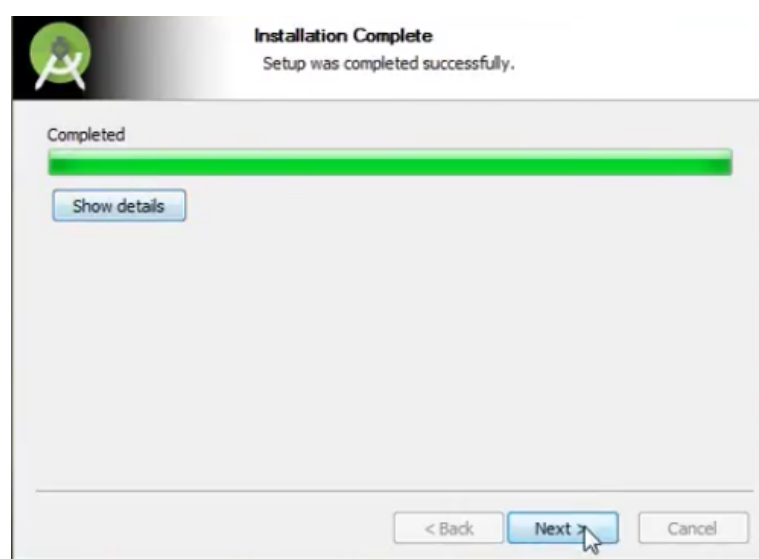
Click on next .

Step – 6 This will start the Android Studio. Meanwhile it will be finding the available SDK components .
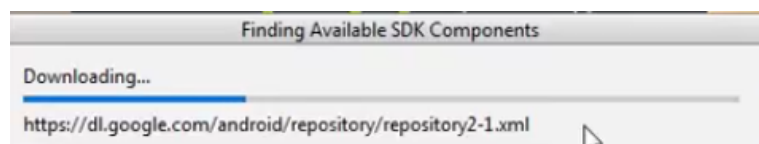
Step – 7 After it has found the SDK components, it will redirect to the Welcome dialog box .
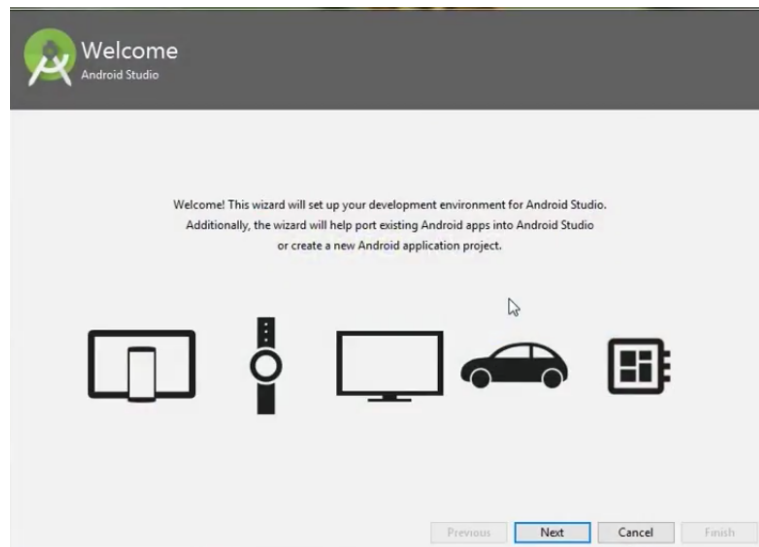
Click on next .
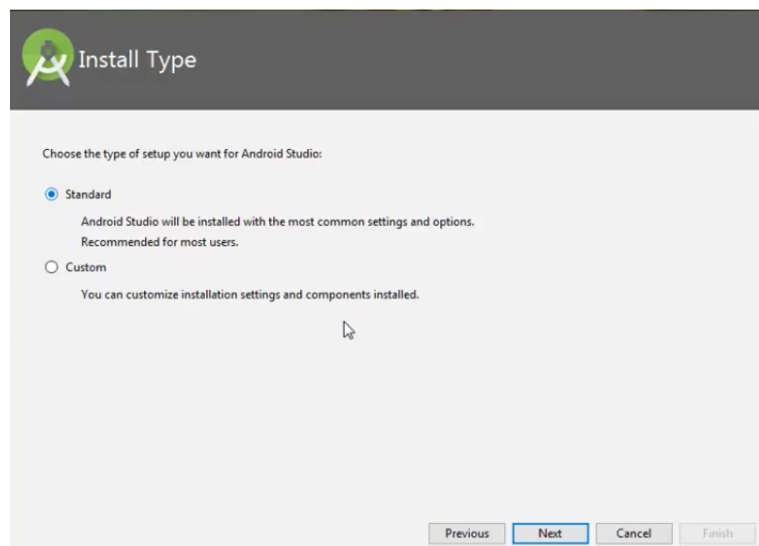
Choose Standard and click on Next. Now choose the theme, whether Light theme or the Dark one .The light one is called the IntelliJ theme whereas the dark theme is called Darcula . Choose as required.

Click on the Next button.

Step – 8 Now it is time to download the SDK components .
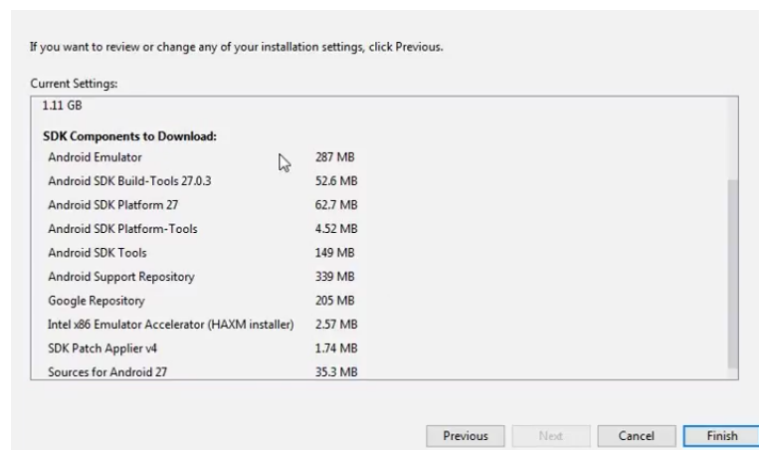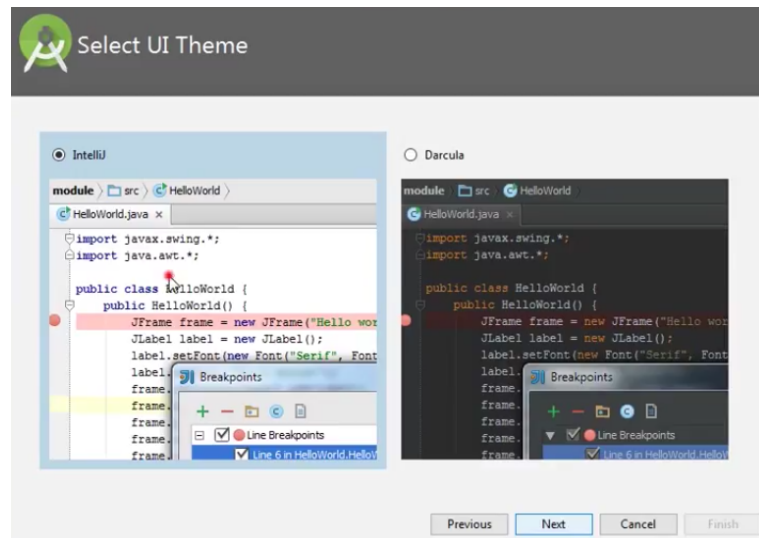
Click on Finish .

It has started downloading Components.
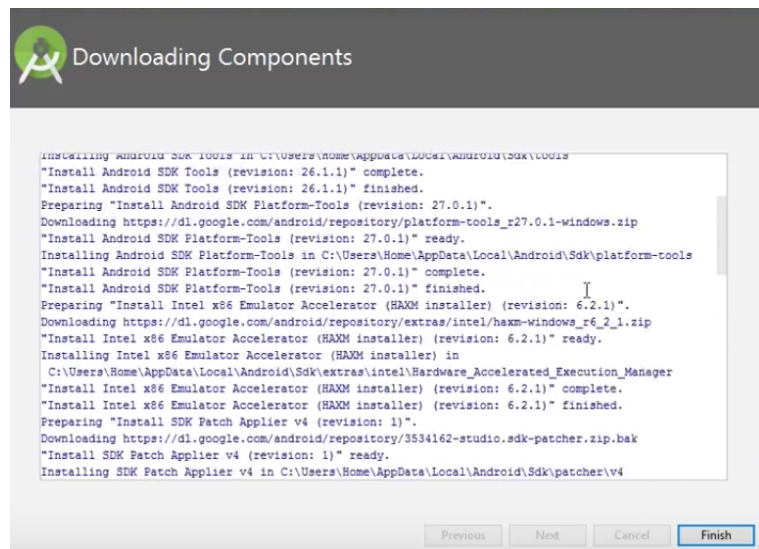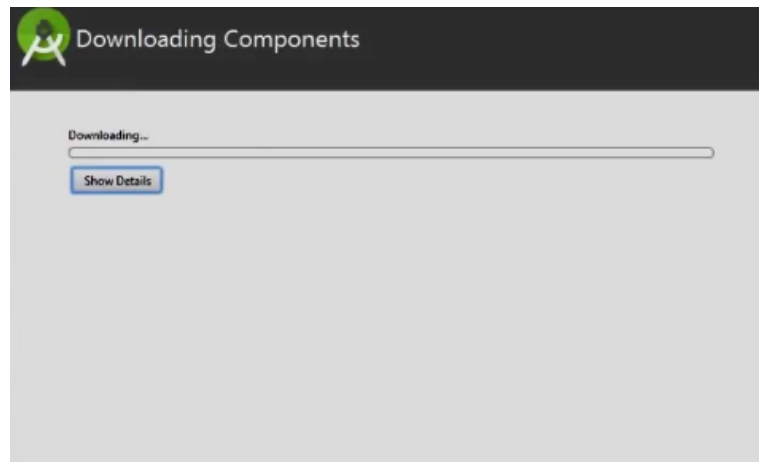
The Android Studio has been successfully configured. Now it's time to launch and build apps.
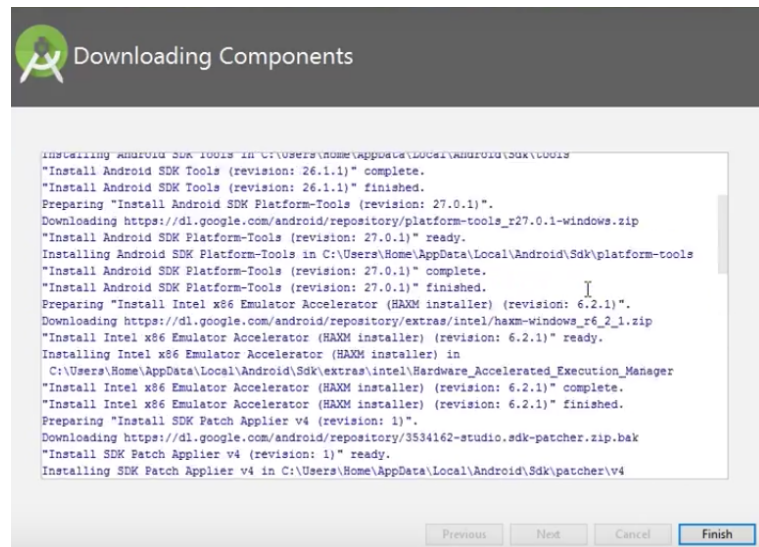
Click on the Finish button to launch it.
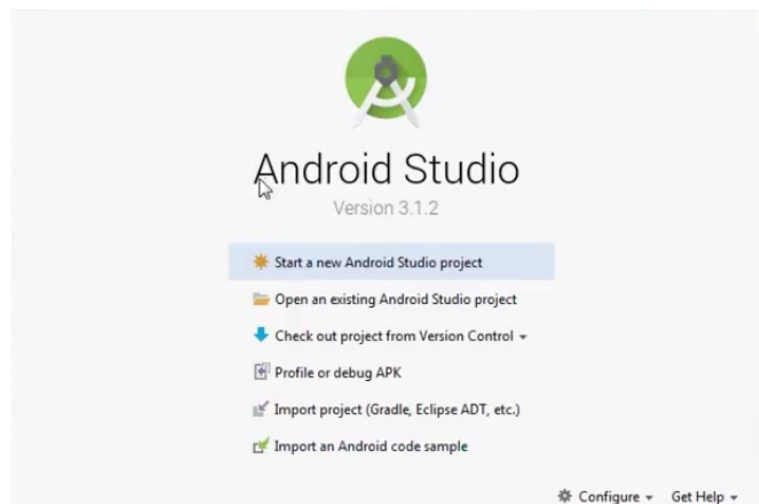
Step – 9 Click on 'Start new android project' to build a new app.

2.2   Components of an Android Application

There are some necessary building blocks that an Android application consists of. These loosely coupled components are bound by the application manifest file which contains description of each component and how they interact. The manifest file also contains the app's metadata, its hardware configuration and platform requirements, external libraries and required permissions.

There are the following main components of an android app:-

1 .Activities- Activities are said to be the presentation layer of our applications. The UI of our application is build around one or more extensions of the Activity class. By using Frag-

ments and Views, activities set the layout and display the output and also respond to the users actions. An activity is implemented as a subclass of class Activity.

These are like invisible workers of our app. These components run at backend, updating your data sources and Activities, triggering Notification and also broadcast Intents. They also perform some tasks when applications are not active. A service can be used as a subclass of class Service:

2.Services- These are like invisible workers of our app. These components run at backend, updating your data sources and Activities, triggering Notification and also broadcast Intents. They also perform some tasks when applications are not active. A service can be used as a subclass of class Service:

```
brightness_5 public class ServiceName extends Service { }
```

3.Content Providers- It is used to manage and persist the application data also typically interact with SQL database. They are also responsible for sharing the data beyond the application boundaries. The Content Providers of a particular application can be configured to allow access from other applications, and the Content Providers exposed by other applications can also be configured. A content provider should be a sub class of the class ContentProvider.

```
brightness_5 public class contentProviderName extends ContentProvide
```

4.Intents- It is a powerful inter-application message-passing framework. They are extensively used throughout Android. Intents can be used to start and stop Activities and Services, to broadcast messages system-wide or to an explicit Activity, Service or Broadcast Receiver or to request an action be performed on a particular piece of data.

5.Broadcast Receivers- They are known to be intent listeners as they enable your application to listen the Intents that satisfy the matching criteria specified by us. Broadcast Receivers make our application to react to any received Intent thereby making them perfect for creating event driven applications.

6.Widgets- These are the small visual application components that you can find on the home screen of the devices. They are special variation of Broadcast Receivers that allow us to create dynamic, interactive application components for users to embed on their Home Screen.

7. Notifications- - Notifications are the application alerts that are used to draw user's attention to some particular app event without stealing focus or interrupting the current Activity of the user. They are generally used to grab user's attention when the application is not visible or active, particularly from within a Service or Broadcast Receiver. Examples: E-mail popups, Messenger popups etc.

3 Android User Interface Design: Implement "Your Name" in Andrid Studio.

3.1 activitymain.xml

```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"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="Kishan Tiwari"
        android:textSize="40dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</LinearLayout>
```

3.2 MainActivity.xml

```java
package com.t.fragment;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```
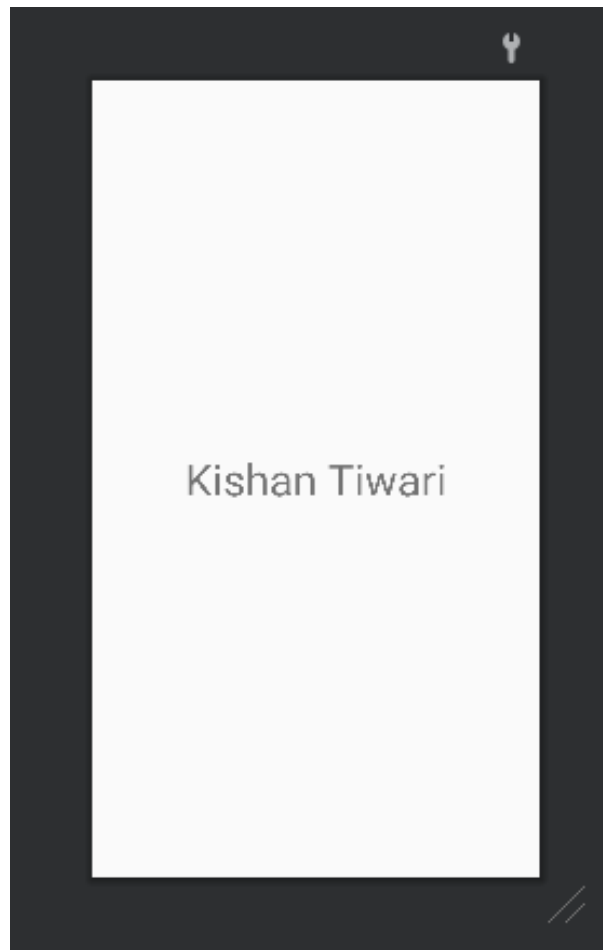
Figure 4: "Output of your Name"

4 Write any text on Button Eg. round shape button

4.1 activitymain.xml

```xml
    <?xml version="1.0" encoding="utf-8"?>
<?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"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="20dp"
    android:layout_marginRight="20dp"
    android:layout_marginBottom="20dp"
    android:background="@drawable/background"
    android:orientation="vertical"
    tools:context=".MainActivity">


    <EditText
        android:id="@+id/id1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="60dp"
        android:layout_marginTop="20dp"
        android:layout_marginRight="60dp"
        android:layout_marginBottom="20dp"
        android:background="@drawable/edit_text_border"
        android:ems="10"
        android:inputType="number"
        android:minHeight="48dp"
        android:paddingLeft="8dp"
        tools:ignore="SpeakableTextPresentCheck" />

    <EditText
        android:id="@+id/id2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="60dp"
        android:layout_marginTop="5dp"
        android:layout_marginRight="60dp"
        android:layout_marginBottom="20dp"
        android:background="@drawable/edit_text_border"
        android:ems="10"
```

```
        android:inputType="number"
        android:minHeight="48dp"
        android:paddingLeft="8dp"
        tools:ignore="SpeakableTextPresentCheck" />

    <TextView
        android:id="@+id/result"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="60dp"
        android:layout_marginTop="5dp"
        android:layout_marginRight="60dp"
        android:layout_marginBottom="25dp"
        android:background="@drawable/edit_text_border"
        android:maxHeight="40dp"
        android:minHeight="40dp"
        android:paddingLeft="8dp"
        android:paddingTop="10dp"
        android:text="TextView" />

    <Button
        android:id="@+id/add"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="40dp"
        android:layout_marginTop="10dp"
        android:layout_marginRight="40dp"
        android:layout_marginBottom="5dp"
        android:background="@drawable/button_border"
        android:text="Addition" />

    <Button
        android:id="@+id/sub"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="40dp"
        android:layout_marginTop="10dp"
        android:layout_marginRight="40dp"
        android:layout_marginBottom="5dp"
        android:background="@drawable/button_border"
        android:text="Subtraction" />

    <Button
        android:id="@+id/multi"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```xml
        android:layout_marginLeft="40dp"
        android:layout_marginTop="10dp"
        android:layout_marginRight="40dp"
        android:layout_marginBottom="5dp"
        android:background="@drawable/button_border"
        android:text="Multiplication" />

    <Button
        android:id="@+id/div"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="40dp"
        android:layout_marginTop="10dp"
        android:layout_marginRight="40dp"
        android:layout_marginBottom="5dp"
        android:background="@drawable/button_border"
        android:text="Division" />

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="77dp"
        android:layout_height="347dp"
        android:layout_gravity="fill_vertical"
        android:layout_marginLeft="5dp"
        android:foregroundGravity="bottom"
        app:srcCompat="@drawable/kt" />


</LinearLayout>
```

4.2 MainActivity.xml

```java
    package com.t.fragment;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity
package com.t.kcalculators;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
```

```java
public class MainActivity extends AppCompatActivity {
    EditText id1,id2;
    Button add,sub,multi,div;
    TextView result;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        id1=findViewById(R.id.id1);
        id2=findViewById(R.id.id2);
        add=findViewById(R.id.add);
        sub=findViewById(R.id.sub);
        multi=findViewById(R.id.multi);
        div=findViewById(R.id.div);
        result=findViewById(R.id.result);

        add.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                int f,s,a;
                f=Integer.parseInt(id1.getText().toString());
                s=Integer.parseInt(id2.getText().toString());
                a=f+s;
                result.setText("Result =" +a);
            }
        });

        sub.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                int f,s,a;
                f=Integer.parseInt(id1.getText().toString());
                s=Integer.parseInt(id2.getText().toString());
                a=f-s;
                result.setText("Result ="+a);
            }
        });

        multi.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                int f,s,a;
                f=Integer.parseInt(id1.getText().toString());
```

```java
                s=Integer.parseInt(id2.getText().toString());
                a=f*s;
                result.setText("Result ="+a);
            }
        });


    div.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            int f,s,a;
            f=Integer.parseInt(id1.getText().toString());
            s=Integer.parseInt(id2.getText().toString());
            a=f/s;
            result.setText("Result ="+a);
        }
    });
    }
}
```

Figure 5: "Round Shape Button"

5 Change the Icon of your APK file in Android Studio.

5.1 AndroidManifest.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Kcalculators"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name=
                "android.intent.category.LAUNCHER" />
            </intent-filter>

            <meta-data
                android:name="android.app.lib_name"
                android:value="" />
        </activity>
    </application>

</manifest>
```
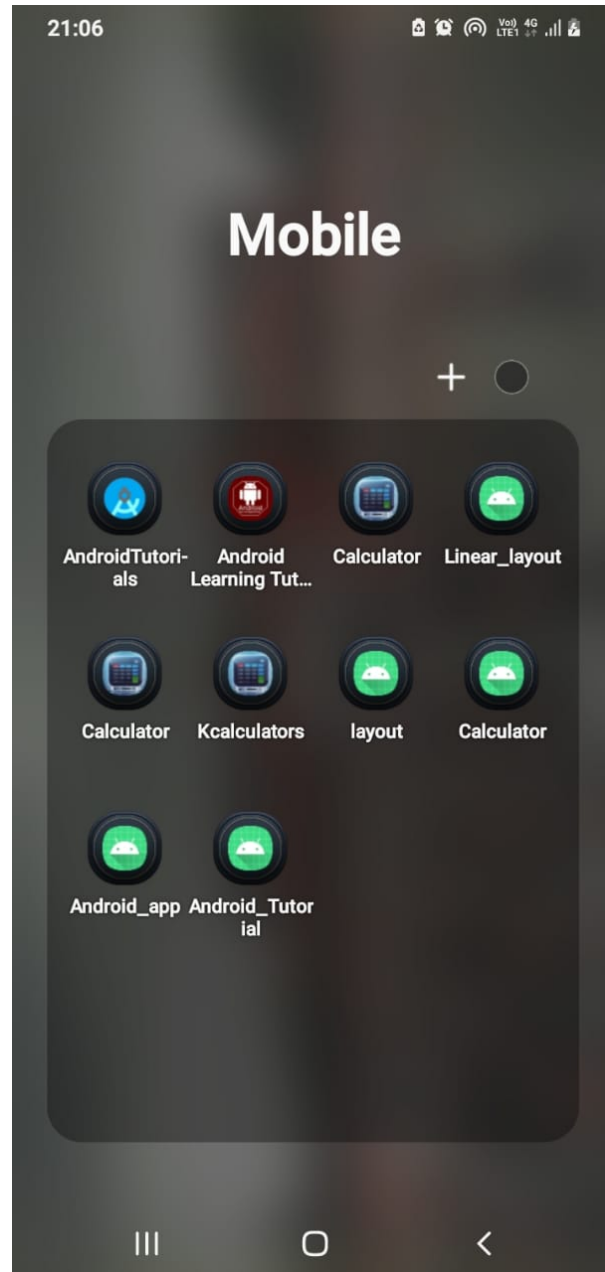
Figure 6: "Icon Of APK files"

6 Create an APK of Second Page Layout like Android Tutorial app Assignment.

6.1 ActicityMain.xml

```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"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/layout2"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:background="@color/purple_200"
        android:orientation="horizontal">

        <Button
            android:id="@+id/tutorial"
            android:layout_width="wrap_content"
            android:layout_height="60dp"
            android:layout_weight="1"
            android:background="@color/purple_200"
            android:text="TUTORIAL"
            android:textColor="@color/white" />

        <Button
            android:id="@+id/example"
            android:layout_width="wrap_content"
            android:layout_height="60dp"
            android:layout_weight="1"
            android:background="@color/purple_200"
            android:text="EXAMPLES"
            android:textColor="@color/white" />

        <Button
            android:id="@+id/quiz"
            android:layout_width="wrap_content"
            android:layout_height="60dp"
            android:layout_weight="1"
            android:background="@color/purple_200"
            android:text="QUIZ"
```

```
            android:textColor="@color/white" />


    </LinearLayout>

    <LinearLayout
        android:id="@+id/main_layout"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">


        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="160dp"
            android:orientation="horizontal">

            <ImageView
                android:id="@+id/imageView"
                android:layout_width="wrap_content"
                android:layout_height="match_parent"
                android:layout_weight="1"
                android:padding="20dp"
                app:srcCompat="@drawable/androidlogo" />

            <View
                android:layout_width="3dp"
                android:layout_height="wrap_content"
                android:layout_gravity="center"
                android:background="@color/purple_200" />


            <ImageView
                android:id="@+id/imageView2"
                android:layout_width="wrap_content"
                android:layout_height="match_parent"
                android:layout_weight="1"
                android:padding="20dp"
                app:srcCompat="@drawable/project_structure" />


        </LinearLayout>

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="50dp"
```

```
        android:orientation="horizontal">

        <TextView
            android:id="@+id/textView4"
            android:layout_width="150dp"
            android:layout_height="38dp"
            android:layout_weight="1"
            android:gravity="center"
            android:text="Introduction"
            android:textColor="@color/black"
            android:textSize="20dp" />


        <View
            android:layout_width="3dp"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:background="@color/purple_200" />

        <TextView
            android:id="@+id/textView5"
            android:layout_width="wrap_content"
            android:layout_height="38dp"
            android:layout_weight="1"
            android:gravity="center"
            android:text="Project structure"
            android:textColor="@color/black"
            android:textSize="20dp" />
    </LinearLayout>

    <View
        android:layout_width="wrap_content"
        android:layout_height="3dp"
        android:background="@color/purple_200" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="160dp"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/imageView6"
            android:layout_width="wrap_content"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:padding="20dp"
```

```
            app:srcCompat="@drawable/layouts" />

        <View
            android:layout_width="3dp"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:background="@color/purple_200" />


        <ImageView
            android:id="@+id/imageView7"
            android:layout_width="wrap_content"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:padding="20dp"
            app:srcCompat="@drawable/widget" />


    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/textView8"
            android:layout_width="98dp"
            android:layout_height="38dp"
            android:layout_weight="1"
            android:gravity="center"
            android:text="Layouts"
            android:textColor="@color/black"
            android:textSize="20dp" />


        <View
            android:layout_width="3dp"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:background="@color/purple_200" />

        <TextView
            android:id="@+id/textView9"
            android:layout_width="wrap_content"
            android:layout_height="38dp"
```

```
            android:layout_weight="1"
            android:gravity="center"
            android:text="UI Widgets"
            android:textColor="@color/black"
            android:textSize="20dp" />
</LinearLayout>

<View
    android:layout_width="wrap_content"
    android:layout_height="3dp"
    android:background="@color/purple_200" />


<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="160dp"
    android:orientation="horizontal">

    <ImageView
        android:id="@+id/imageView10"
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:padding="20dp"
        app:srcCompat="@drawable/activities" />

    <View
        android:layout_width="3dp"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:background="@color/purple_200" />


    <ImageView
        android:id="@+id/imageView11"
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:padding="20dp"
        app:srcCompat="@drawable/services" />


</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
```

```xml
        android:layout_height="50dp"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/textView12"
            android:layout_width="77dp"
            android:layout_height="38dp"
            android:layout_weight="1"
            android:gravity="center"
            android:text="Activities"
            android:textColor="@color/black"
            android:textSize="20dp" />


        <View
            android:layout_width="3dp"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:background="@color/purple_200" />

        <TextView
            android:id="@+id/textView13"
            android:layout_width="wrap_content"
            android:layout_height="38dp"
            android:layout_weight="1"
            android:gravity="center"
            android:text="Services"
            android:textColor="@color/black"
            android:textSize="20dp" />
    </LinearLayout>

    <View
        android:layout_width="wrap_content"
        android:layout_height="3dp"
        android:background="@color/purple_200" />
    </LinearLayout>
</LinearLayout>
```

6.2 MainActivity.xml

```java
package com.t.android_tutorial;

import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentTransaction;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
```

```
import android.view.View;
import android.widget.Button;
import android.widget.LinearLayout;


public class MainActivity extends AppCompatActivity {
    Button btntutorial,btnexample,btnquiz;
    LinearLayout main_layout;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btntutorial=findViewById(R.id.tutorial);
        btnexample=findViewById(R.id.example);
        btnquiz=findViewById(R.id.quiz);
        main_layout=findViewById(R.id.main_layout);
}
}
```
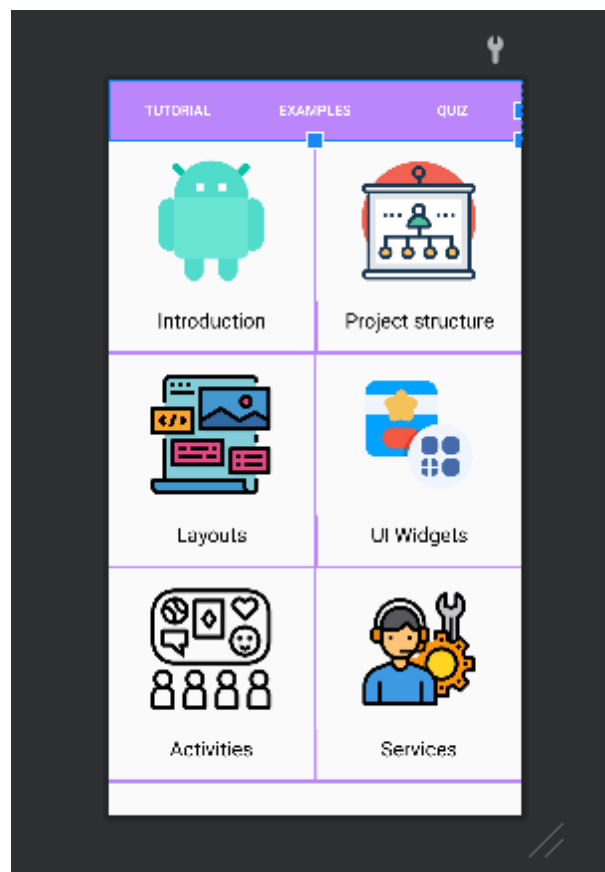

Figure 7: "Second Page Layout Of Android Tutorial"

7 Create Apk file with button and on button click i.e on click task 1 it will move to other task 2

7.1 ActicityMain.xml

```xml
<?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:id="@+id/register"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginLeft="5dp"
    android:layout_marginTop="5dp"
    android:layout_marginRight="5dp"
    android:layout_marginBottom="5dp"
    android:background="@drawable/bg"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/signuptitle"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="25dp"
        android:gravity="center"
        android:text="Sign Up"
        android:textColor="@color/white"
        android:textSize="35dp"
        android:textStyle="bold" />

    <EditText
        android:id="@+id/username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/signuptitle"
        android:layout_marginStart="10dp"
        android:layout_marginTop="10dp"
        android:layout_marginEnd="10dp"
        android:layout_marginBottom="10dp"
        android:background="#30ffffff"
        android:drawableLeft="@drawable/ic_baseline_person_outline_24"
        android:drawablePadding="20dp"
        android:hint="Username"
        android:padding="20dp"
        android:textColor="@color/white"
        android:textColorHint="@color/white" />
```

```xml
<EditText
    android:id="@+id/email"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/username"
    android:layout_marginStart="10dp"
    android:layout_marginTop="10dp"
    android:layout_marginEnd="10dp"
    android:layout_marginBottom="10dp"
    android:background="#30ffffff"
    android:drawableLeft="@drawable/ic_baseline_email_24"
    android:drawablePadding="20dp"
    android:hint="Email"
    android:padding="20dp"
    android:textColor="@color/white"
    android:textColorHint="@color/white" />


<EditText
    android:id="@+id/password"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/email"
    android:layout_marginStart="10dp"
    android:layout_marginTop="10dp"
    android:layout_marginEnd="10dp"
    android:layout_marginBottom="10dp"
    android:background="#30ffffff"
    android:drawableLeft="@drawable/ic_baseline_password_24"
    android:drawablePadding="20dp"
    android:hint="Password"
    android:inputType="textPassword"
    android:padding="20dp"
    android:textColor="@color/white"
    android:textColorHint="@color/white" />

<EditText
    android:id="@+id/rpassword"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/password"
    android:layout_marginStart="10dp"
    android:layout_marginTop="10dp"
    android:layout_marginEnd="10dp"
    android:layout_marginBottom="10dp"
```

```xml
        android:background="#30ffffff"
        android:drawableLeft="@drawable/ic_baseline_password_24"
        android:drawablePadding="20dp"
        android:hint="Confirm Password "
        android:inputType="textPassword"
        android:padding="20dp"
        android:textColor="@color/white"
        android:textColorHint="@color/white" />

    <Button
        android:id="@+id/regis"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/rpassword"
        android:layout_marginLeft="80dp"
        android:layout_marginTop="40dp"
        android:layout_marginRight="60dp"
        android:gravity="center"
        android:text="Register" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/info"
        android:text="or Sign Up with"
        android:layout_centerHorizontal="true"
        android:layout_above="@id/socialicons"
        android:textColor="@color/white"
        android:textSize="15dp"
        android:textStyle="bold"/>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/socialicons"
        android:layout_alignParentBottom="true"
        android:gravity="center">

        <ImageView
            android:layout_width="48dp"
            android:layout_height="48dp"
            android:src="@drawable/google"
            android:layout_margin="20dp"/>

        <ImageView
            android:layout_width="48dp"
```

```xml
            android:layout_height="48dp"
            android:src="@drawable/fb"
            android:layout_margin="20dp"/>

        <ImageView
            android:layout_width="48dp"
            android:layout_height="48dp"
            android:src="@drawable/twitter"
            android:layout_margin="20dp"/>

    </LinearLayout>
</RelativeLayout>
```

7.2 MainActivity.xml

```java
package com.t.signinform;

import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Button;
import android.widget.Toast;


public class MainActivity extends AppCompatActivity {
    EditText username;
    EditText email;
    EditText password;
    Button regis;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        database g=new database(this);

        //SQLiteDatabase db=g.getReadableDatabase();

        username = findViewById(R.id.username);
        password = findViewById(R.id.password);
        email=findViewById(R.id.email);
        regis = findViewById(R.id.regis);
```

```java
            regis.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    String n=username.getText().toString();
                    String m=email.getText().toString();
                    String p=password.getText().toString();
                    if(n.equals("") || m.equals("") || p.equals(""))
                    {
                        Toast.makeText(MainActivity.this
                        ,"Enter All The Fields",
                        Toast.LENGTH_SHORT).show();
                    }
                    else if(n.equals("kishan") && p.equals("tiwari"))
                    {
                        Intent i=new Intent(MainActivity.
                        this,Kcalculator.class);
                        startActivity(i);
                    }
                    else
                    {
                                Boolean i=g.insert_data(n,m,p);
                                if(i==true)
                                {
                                    Toast.makeText(MainActivity.
                                    this,"Successful",Toast.
                                    LENGTH_SHORT).show();
                                     }
                                else{
                                    Toast.makeText(MainActivity.
                                    this,"Not Successful",Toast
                                    .LENGTH_SHORT).show();
                                    }
                    }
                }
            });
            }
        }
```

7.3   activitykcalculator.xml

```java
package com.t.signinform;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
```

```java
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class Kcalculator extends AppCompatActivity {
    EditText id1,id2;
    Button add,sub,multi,div;
    TextView result;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_kcalculator);
        id1=findViewById(R.id.id1);
        id2=findViewById(R.id.id2);
        add=findViewById(R.id.add);
        sub=findViewById(R.id.sub);
        multi=findViewById(R.id.multi);
        div=findViewById(R.id.div);
        result=findViewById(R.id.result);

        add.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                String p=id1.getText().toString();
                String q=id2.getText().toString();


                if(p.equals("24") && q.equals("@"))
                {
                    Intent intent =new Intent();
                    intent.setAction(Intent.ACTION_VIEW);
                    intent.setData(Uri.parse("https:
                    //www.google.com/"));
                    startActivity(intent);
                }
                else
                {
                    int f,s,a;
                    f=Integer.parseInt(id1.getText().toString());
                    s=Integer.parseInt(id2.getText().toString());
```

```java
                        a=f+s;
                        result.setText("Result ="+a);
                    }

            }
        });

        sub.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                int f,s,a;
                f=Integer.parseInt(id1.getText().toString());
                s=Integer.parseInt(id2.getText().toString());
                a=f-s;
                result.setText("Result ="+a);
            }
        });

        multi.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                int f,s,a;
                f=Integer.parseInt(id1.getText().toString());
                s=Integer.parseInt(id2.getText().toString());
                a=f*s;
                result.setText("Result ="+a);
            }
        });

        div.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                int f,s,a;
                f=Integer.parseInt(id1.getText().toString());
                s=Integer.parseInt(id2.getText().toString());
                a=f/s;
                result.setText("Result ="+a);
            }
        });
    }
}

package com.t.signinform;

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

```java
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class Kcalculator extends AppCompatActivity {
    EditText id1,id2;
    Button add,sub,multi,div;
    TextView result;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_kcalculator);
        id1=findViewById(R.id.id1);
        id2=findViewById(R.id.id2);
        add=findViewById(R.id.add);
        sub=findViewById(R.id.sub);
        multi=findViewById(R.id.multi);
        div=findViewById(R.id.div);
        result=findViewById(R.id.result);

        add.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                String p=id1.getText().toString();
                String q=id2.getText().toString();


                if(p.equals("24") && q.equals("@"))
                {
                    Intent intent =new Intent();
                    intent.setAction(Intent.ACTION_VIEW);
                    intent.setData(Uri.parse("
                    https://www.google.com/"));
                    startActivity(intent);
                }
                else
                {
                    int f,s,a;
                    f=Integer.parseInt(id1.getText().toString());
```

```java
                s=Integer.parseInt(id2.getText().toString());
                a=f+s;
                result.setText("Result ="+a);
            }
        }
    });

    sub.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            int f,s,a;
            f=Integer.parseInt(id1.getText().toString());
            s=Integer.parseInt(id2.getText().toString());
            a=f-s;
            result.setText("Result ="+a);
        }
    });

    multi.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            int f,s,a;
            f=Integer.parseInt(id1.getText().toString());
            s=Integer.parseInt(id2.getText().toString());
            a=f*s;
            result.setText("Result ="+a);
        }
    });

    div.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            int f,s,a;
            f=Integer.parseInt(id1.getText().toString());
            s=Integer.parseInt(id2.getText().toString());
            a=f/s;
            result.setText("Result ="+a);
        }
    });
}
}
```
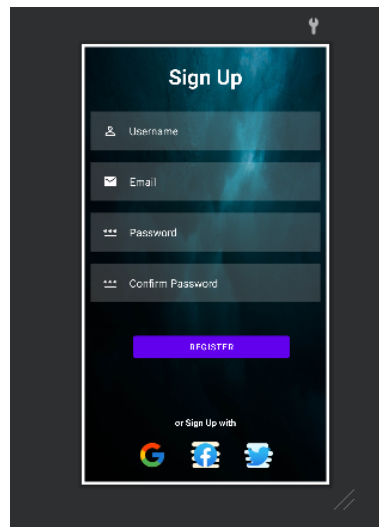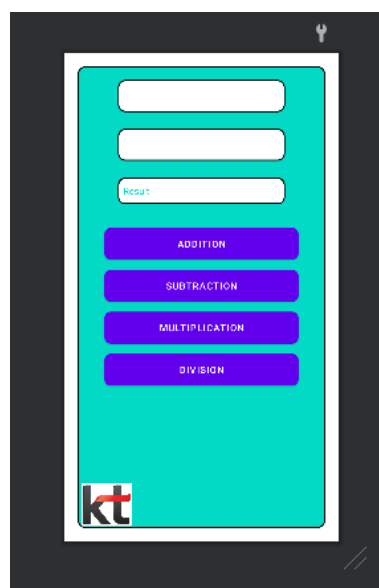
Figure 8: "After saving data Page will move to Calculator"

Figure 9: "Calculator"

8 Create an APK file with page of fragment in an application.

8.1 activitymain.xml

```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"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@android:color/holo_blue_dark"
    android:orientation="vertical"
    tools:context=".MainActivity">


    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <Button
            android:id="@+id/red"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_margin="10dp"
            android:layout_weight="1"
            android:background="@android:color/holo_purple"
            android:text="Red" />

        <Button
            android:id="@+id/yellow"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_margin="10dp"
            android:layout_weight="1"
            android:background="@android:color/holo_purple"
            android:text="Yellow" />
    </LinearLayout>

    <LinearLayout
        android:id="@+id/layout"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="horizontal"></LinearLayout>
</LinearLayout>
```
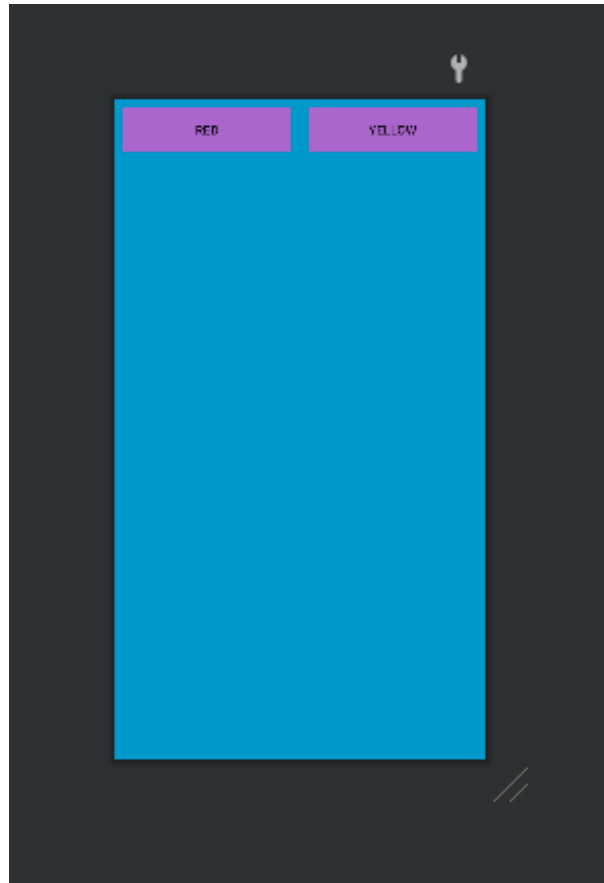
Figure 10: "E.g Fragment Apk"

8.2 MainActivity.xml

```
package com.t.fragment;

import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentTransaction;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.LinearLayout;

public class MainActivity extends AppCompatActivity {
    Button btnred,btnyellow;
    LinearLayout layout1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```java
        setContentView(R.layout.activity_main);

        btnred=findViewById(R.id.red);
        btnyellow=findViewById(R.id.yellow);
        layout1=findViewById(R.id.layout);

        btnred.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Red_fragment firstfragment=new Red_fragment();
                FragmentTransaction transaction
                =getSupportFragmentManager()
                .beginTransaction();
                transaction.replace(R.
                id.layout,firstfragment);
                transaction.commit();
            }
        });

        btnyellow.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Yellow_fragment Secondfragment=
                new Yellow_fragment();
                FragmentTransaction transaction
                =getSupportFragmentManager().
                beginTransaction();
                transaction.
                replace(R.id.layout,Secondfragment);
                transaction.commit();
            }
        });



    }
}
```

8.3   redfragment.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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=".Red_fragment">

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="@android:color/holo_red_light"
        android:gravity="center"
        android:text="Red Fragment"
        android:textSize="50dp" />
</FrameLayout>
```


Figure 11: "Red Fragment"

8.4   redfragment.xml

```
    <?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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=".Yellow_fragment">

<!-- TODO: Update blank fragment layout -->
<TextView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@android:color/holo_orange_light"
    android:gravity="center"
    android:text="Yellow Fragment"
    android:textSize="50dp" />
```

```
</FrameLayout>
```


Figure 12: "Yellow Fragment"

8.5   redfragment.java

```
package com.t.fragment;
```

```
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

public class Red_fragment extends Fragment {

    @Override
    public View onCreateView(LayoutInflater
    inflater, ViewGroup container,
                            Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.
        fragment_red_fragment, container, false);
    }
}
```

8.6   yellowfragment.java

```
    package com.t.fragment;

import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

public class Yellow_fragment extends Fragment {

    @Override
    public View onCreateView(LayoutInflater
    inflater, ViewGroup container,
                            Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.
        fragment_yellow_fragment, container, false);
    }
}
```

9 Create a form of Sign-In and feed data using SQLite Content Provider Database

9.1 MainActivity.java

```java
package com.t.signinform;

import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Button;
import android.widget.Toast;


public class MainActivity extends AppCompatActivity {
    EditText username;
    EditText email;
    EditText password;
    Button regis;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        database g=new database(this);

        //SQLiteDatabase db=g.getReadableDatabase();

        username = findViewById(R.id.username);
        password = findViewById(R.id.password);
        email=findViewById(R.id.email);
        regis = findViewById(R.id.regis);

        regis.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String n=username.getText().toString();
                String m=email.getText().toString();
                String p=password.getText().toString();
                if(n.equals("") || m.equals("") || p.equals(""))
                {
                    Toast.makeText(MainActivity.this,"Enter All The Fiel
```

```
                }
                else if(n.equals("kishan") && p.equals("tiwari")){
                    Intent i=new Intent(MainActivity.this,
                    Kcalculator.class);
                    startActivity(i);
                }
                else
                {
                        Boolean i=g.insert_data(n,m,p);
                        if(i==true)
                        {
                            Toast.makeText(MainActivity.this,
                            "Successful",Toast.
                            LENGTH_SHORT).show();
                             }
                        else{
                            Toast.makeText(MainActivity
                            .this,"Not Successful",Toast.
                            LENGTH_SHORT).show();
                            }
                }

            }
        });
            }
    }
```

9.2 database.java

```
    package com.t.signinform;

import android.content.ContentValues;
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import androidx.annotation.Nullable;

public class database extends SQLiteOpenHelper {
    private static final String dbname="signup.db";
    public database(Context context){
        super(context, dbname, null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase sqLiteDatabase) {
```

51

```java
        String q="create table users (_id
        integer primary key AUTOINCREMENT ,username
        text,email text, password text )";
        sqLiteDatabase.execSQL(q);
    }


    @Override
    public void onUpgrade(SQLiteDatabase
    sqLiteDatabase, int i, int i1) {
        sqLiteDatabase.execSQL("drop table if exists users");
        onCreate(sqLiteDatabase);


    }
    public boolean insert_data(String
    username, String email, String password){
        SQLiteDatabase db=this.getWritableDatabase();
        ContentValues c=new ContentValues();
        c.put("username",username);
        c.put("email",email);
        c.put("password",password);
        long r=db.insert("users",null,c);
        if (r==-1) {
            return false;
        }
        else {
            return true;
        }
    }
}
```

9.3   activityMain.xml

```xml
    <?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:id="@+id/register"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginLeft="5dp"
    android:layout_marginTop="5dp"
    android:layout_marginRight="5dp"
    android:layout_marginBottom="5dp"
    android:background="@drawable/bg"
    tools:context=".MainActivity">
```

```xml
<TextView
    android:id="@+id/signuptitle"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="25dp"
    android:gravity="center"
    android:text="Sign Up"
    android:textColor="@color/white"
    android:textSize="35dp"
    android:textStyle="bold" />

<EditText
    android:id="@+id/username"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/signuptitle"
    android:layout_marginStart="10dp"
    android:layout_marginTop="10dp"
    android:layout_marginEnd="10dp"
    android:layout_marginBottom="10dp"
    android:background="#30ffffff"
    android:drawableLeft="@drawable/
    ic_baseline_person_outline_24"
    android:drawablePadding="20dp"
    android:hint="Username"
    android:padding="20dp"
    android:textColor="@color/white"
    android:textColorHint="@color/white" />

<EditText
    android:id="@+id/email"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/username"
    android:layout_marginStart="10dp"
    android:layout_marginTop="10dp"
    android:layout_marginEnd="10dp"
    android:layout_marginBottom="10dp"
    android:background="#30ffffff"
    android:drawableLeft="@drawable/
    ic_baseline_email_24"
    android:drawablePadding="20dp"
    android:hint="Email"
    android:padding="20dp"
    android:textColor="@color/white"
```

```xml
        android:textColorHint="@color/white" />


    <EditText
        android:id="@+id/password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/email"
        android:layout_marginStart="10dp"
        android:layout_marginTop="10dp"
        android:layout_marginEnd="10dp"
        android:layout_marginBottom="10dp"
        android:background="#30ffffff"
        android:drawableLeft="@drawable/
        ic_baseline_password_24"
        android:drawablePadding="20dp"
        android:hint="Password"
        android:inputType="textPassword"
        android:padding="20dp"
        android:textColor="@color/white"
        android:textColorHint="@color/white" />

    <EditText
        android:id="@+id/rpassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/password"
        android:layout_marginStart="10dp"
        android:layout_marginTop="10dp"
        android:layout_marginEnd="10dp"
        android:layout_marginBottom="10dp"
        android:background="#30ffffff"
        android:drawableLeft="@drawable/
        ic_baseline_password_24"
        android:drawablePadding="20dp"
        android:hint="Confirm Password "
        android:inputType="textPassword"
        android:padding="20dp"
        android:textColor="@color/white"
        android:textColorHint="@color/white" />

    <Button
        android:id="@+id/regis"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/rpassword"
```

```xml
        android:layout_marginLeft="80dp"
        android:layout_marginTop="40dp"
        android:layout_marginRight="60dp"
        android:gravity="center"
        android:text="Register" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/info"
        android:text="or Sign Up with"
        android:layout_centerHorizontal="true"
        android:layout_above="@id/socialicons"
        android:textColor="@color/white"
        android:textSize="15dp"
        android:textStyle="bold"/>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/socialicons"
        android:layout_alignParentBottom="true"
        android:gravity="center">

        <ImageView
            android:layout_width="48dp"
            android:layout_height="48dp"
            android:src="@drawable/google"
            android:layout_margin="20dp"/>

        <ImageView
            android:layout_width="48dp"
            android:layout_height="48dp"
            android:src="@drawable/fb"
            android:layout_margin="20dp"/>

        <ImageView
            android:layout_width="48dp"
            android:layout_height="48dp"
            android:src="@drawable/twitter"
            android:layout_margin="20dp"/>

    </LinearLayout>
</RelativeLayout>
```
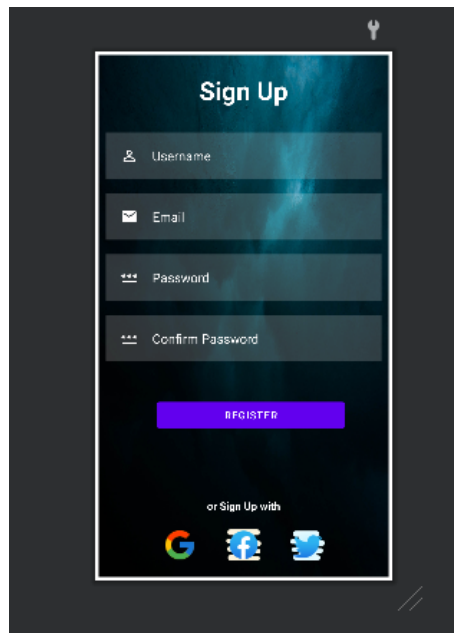
Figure 13: "Sign-In form"

9.4 activitykcalculator.xml

```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"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="20dp"
    android:layout_marginRight="20dp"
    android:layout_marginBottom="20dp"
    android:background="@drawable/background"
    android:orientation="vertical"
    tools:context=".Kcalculator">


    <EditText
        android:id="@+id/id1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="60dp"
        android:layout_marginTop="20dp"
        android:layout_marginRight="60dp"
        android:layout_marginBottom="20dp"
        android:background="@drawable/edit_text_border"
```

```xml
        android:ems="10"
        android:inputType="text"
        android:minHeight="48dp"
        android:paddingLeft="8dp"
        android:textColor="@color/black"
        tools:ignore="SpeakableTextPresentCheck" />

    <EditText
        android:id="@+id/id2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="60dp"
        android:layout_marginTop="5dp"
        android:layout_marginRight="60dp"
        android:layout_marginBottom="20dp"
        android:background="@drawable/edit_text_border"
        android:ems="10"
        android:inputType="text"
        android:minHeight="48dp"
        android:paddingLeft="8dp"
        android:textColor="@color/black"
        tools:ignore="SpeakableTextPresentCheck" />

    <TextView
        android:id="@+id/result"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="60dp"
        android:layout_marginTop="5dp"
        android:layout_marginRight="60dp"
        android:layout_marginBottom="25dp"
        android:background="@drawable/edit_text_border"
        android:hint="Result"
        android:maxHeight="40dp"
        android:minHeight="40dp"
        android:paddingLeft="8dp"
        android:paddingTop="10dp"
        android:textColor="@color/black"
        android:textColorHint="@color/teal_200" />

    <Button
        android:id="@+id/add"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="40dp"
        android:layout_marginTop="10dp"
```

```xml
        android:layout_marginRight="40dp"
        android:layout_marginBottom="5dp"
        android:background="@drawable/button_border"
        android:text="Addition" />

    <Button
        android:id="@+id/sub"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="40dp"
        android:layout_marginTop="10dp"
        android:layout_marginRight="40dp"
        android:layout_marginBottom="5dp"
        android:background="@drawable/button_border"
        android:text="Subtraction" />

    <Button
        android:id="@+id/multi"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="40dp"
        android:layout_marginTop="10dp"
        android:layout_marginRight="40dp"
        android:layout_marginBottom="5dp"
        android:background="@drawable/button_border"
        android:text="Multiplication" />

    <Button
        android:id="@+id/div"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="40dp"
        android:layout_marginTop="10dp"
        android:layout_marginRight="40dp"
        android:layout_marginBottom="5dp"
        android:background="@drawable/button_border"
        android:text="Division" />

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="77dp"
        android:layout_height="347dp"
        android:layout_gravity="fill_vertical"
        android:layout_marginLeft="5dp"
        android:foregroundGravity="bottom"
        app:srcCompat="@drawable/kt" />
```
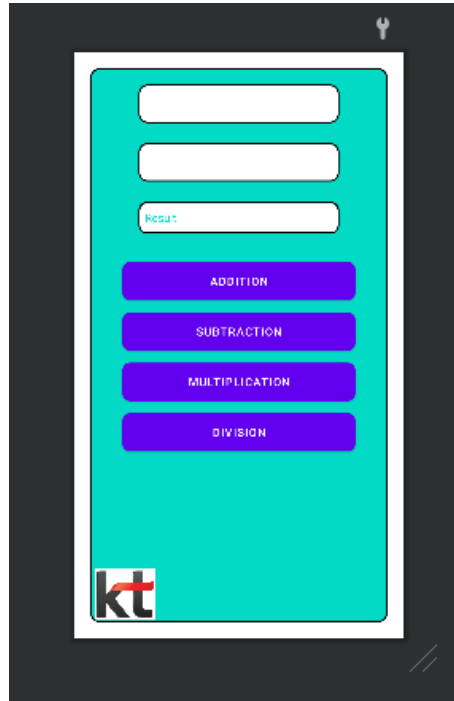
```
</LinearLayout>
```


Figure 14: "Calculator APk opened after login"

9.5 kcalculator.java

```java
package com.t.signinform;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class Kcalculator extends AppCompatActivity {
    EditText id1,id2;
    Button add,sub,multi,div;
```

```java
TextView result;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_kcalculator);
    id1=findViewById(R.id.id1);
    id2=findViewById(R.id.id2);
    add=findViewById(R.id.add);
    sub=findViewById(R.id.sub);
    multi=findViewById(R.id.multi);
    div=findViewById(R.id.div);
    result=findViewById(R.id.result);

    add.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {

            String p=id1.getText().toString();
            String q=id2.getText().toString();


            if(p.equals("24") && q.equals("@"))
            {
                Intent intent =new Intent();
                intent.setAction(Intent.ACTION_VIEW);
                intent.setData(Uri.parse("https://www.google.com
                startActivity(intent);
            }
            else
            {
                int f,s,a;
                f=Integer.parseInt(id1.getText().toString());
                s=Integer.parseInt(id2.getText().toString());
                a=f+s;
                result.setText("Result ="+a);
            }

        }
    });

    sub.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            int f,s,a;
            f=Integer.parseInt(id1.getText().toString());
```

```java
            s=Integer.parseInt(id2.getText().toString());
            a=f-s;
            result.setText("Result ="+a);
        }
    });

    multi.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            int f,s,a;
            f=Integer.parseInt(id1.getText().toString());
            s=Integer.parseInt(id2.getText().toString());
            a=f*s;
            result.setText("Result ="+a);
        }
    });

    div.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            int f,s,a;
            f=Integer.parseInt(id1.getText().toString());
            s=Integer.parseInt(id2.getText().toString());
            a=f/s;
            result.setText("Result ="+a);
        }
    });
    }
}
```

10 Demonsatration On Services

10.1 activityMain.xml

```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"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@android:color/holo_blue_light"
    android:gravity="center"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/btnstart"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="20dp"
        android:background="@color/purple_500"
        android:gravity="center"
        android:padding="20dp"
        android:text="Start service"
        android:textSize="20dp" />


    <Button
        android:id="@+id/btnstop"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="20dp"
        android:background="@color/purple_500"
        android:gravity="center"
        android:padding="20dp"
        android:text="Stop service"
        android:textSize="20dp" />
</LinearLayout>
```

10.2 MainActivity.java

```java
package com.t.service;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
```

```java
import android.view.View;
import android.widget.Button;

public class MainActivity extends
AppCompatActivity implements View.OnClickListener {
    Button btnstart,btnstop;



    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        btnstart=findViewById(R.id.btnstart);
        btnstop=findViewById(R.id.btnstop);

        btnstart.setOnClickListener(this);
        btnstop.setOnClickListener(this);

    }

    @Override
    public void onClick(View view) {
    if(view==btnstart){
        startService(new Intent(this,Musicservice.class));
    }else if(view==btnstop){
        stopService(new Intent(this,Musicservice.class));
    }
    }
}
```

10.3   Musicservice.java

```java
    package com.t.service;

import android.app.Service;
import android.content.Intent;
import android.media.MediaPlayer;
import android.os.IBinder;
import android.provider.Settings;
import android.support.annotation.Nullable;

public class Musicservice extends Service {
    private MediaPlayer mp;
```

```java
    @Nullable
    @Override
    public IBinder onBind(Intent intent) {
        return null;
    }

    @Override
    public int onStartCommand(Intent intent,
    int flags, int startId) {
        mp=MediaPlayer.create(this, Settings
        .System.DEFAULT_RINGTONE_URI);

        mp.setLooping(true);

        mp.start();

        return START_STICKY;
    }

    @Override
    public void onDestroy() {

        mp.stop();
        super.onDestroy();
    }
}
```

11 Demonstration on Android Security and Debugging

11.1 activitymain.xml

```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"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@android:color/holo_blue_light"
    android:gravity="center"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <Button
```
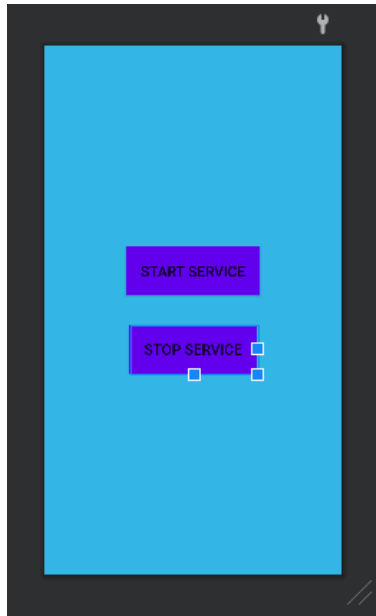
Figure 15: "Service Example"

```
    android:id="@+id/btnstart"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="20dp"
    android:background="@color/purple_500"
    android:gravity="center"
    android:padding="20dp"
    android:text="Start service"
    android:textSize="20dp" />


<TextView
    android:id="@+id/textView"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:padding="10dp"
    android:text="Debbuging"
    android:textSize="40dp" />

<Button
    android:id="@+id/btnstop"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="20dp"
    android:background="@color/purple_500"
    android:gravity="center"
```

```
        android:padding="20dp"
        android:text="Stop service"
        android:textSize="20dp" />
</LinearLayout>
```

11.2 Musicservice.java

```java
package com.t.service;

import android.app.Service;
import android.content.Intent;
import android.media.MediaPlayer;
import android.os.IBinder;
import android.provider.Settings;
import android.support.annotation.Nullable;
import android.widget.Toast;

public class Musicservice extends Service {
    private MediaPlayer mp;

    @Nullable
    @Override
    public IBinder onBind(Intent intent) {
        return null;
    }

    @Override
    public void onCreate(){
        super.onCreate();
        Toast.makeText(this, "Service Started"
        , Toast.LENGTH_SHORT).show();
    }

    @Override
    public int onStartCommand(Intent intent,
    int flags, int startId) {
        mp=MediaPlayer.create(this,
        Settings.System.DEFAULT_RINGTONE_URI);

        mp.setLooping(true);

        mp.start();

        return START_STICKY;
    }
```

```
@Override
public void onDestroy() {

    mp.stop();
    super.onDestroy();
    Toast.makeText(this, "Service End",
    Toast.LENGTH_SHORT).show();
}
}
```
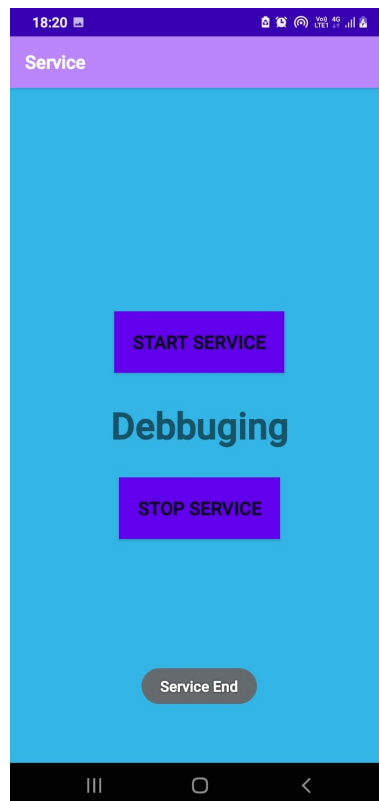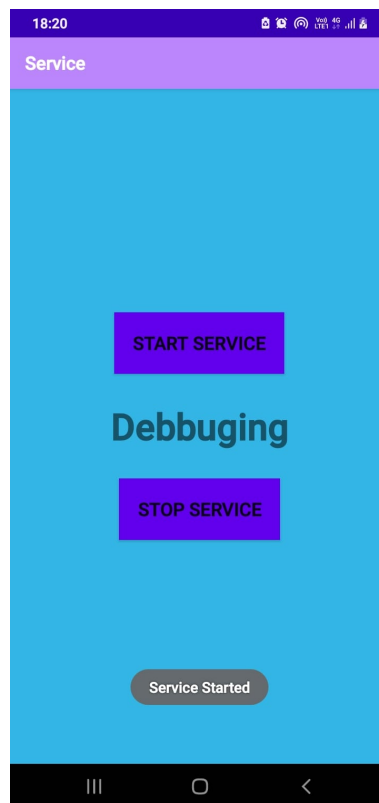

Figure 16: "Toast Message Service started"

Figure 17: "Toast message Service End"