||Jai Sri Gurudev||
Sri AdichunchanagiriShikshana Trust(R)

SJB Institute of Technology

No. 67, BGS Health & Education City, Dr. Vishnuvardhan Road
Kengeri, Bangalore – 560 060

Department of Information Science and Engineering

Mobile Application Development

[18CSMP68]

VI SEMESTER – B. E

Staff Name:	**JEEVARAJ R and RANJITH J**		
Section:	**A & B**	**Batch:**	**A1, A2, A3, B1, B2 and B3**

PREFACE

As smartphone's computing power continues to grow and as mobile applications (apps) continue to dominate digital engagement, apps have become a new frontier for advancing field experiment methodology. Using apps may help researchers to scale up the reach, precisely control randomization and experiment materials, collect a variety of objective and self-reported data over time, and more conveniently replicate and adapt an experiment. We performed a systematic review on field experiments involving apps published between 2007 and 2017.

SJB INTITUTE OF TECHNOLOGY

Institution's Vision

To become a recognized technical education centre with a global perspective

Institution's Mission

To provide learning opportunities that foster students ethical values, intelligent development in science & technology and social responsibility so that they become sensible and contributing members of the society.

Department of Information Science and Engineering

Department Vision

We envision our department as a catalyst for developing educated, engaged and employable individuals whose collective energy will be the driving force for prosperity and the quality of life in our diverse world.

Department Mission

Our mission is to provide quality technical education in the field of information technology and to strive for excellence in the education by developing and sharpening the intellectual and human potential for good industry and community.

PROGRAM EDUCATIONAL OBJECTIVES (PEO'S)

Graduates will -

- Possess expertise in problem solving, design and analysis, technical skills for a fruitful career accomplishing professional and social ethics with exposure to modern designing tools and technologies in Information Science and Engineering.
- Excel in communication, teamwork and multipledomains related to engineering issues accomplishing social responsibilities and management skills.
- Outclass in competitive environment through certification courses, gaining leadership qualities and progressive research to become successful entrepreneurs.

PROGRAM SPECIFIC OUTCOMES (PSO'S)

Graduates will be able to –

1. **PSO1:** Apply the Knowledge of Information Science to develop software solutions in current research trends and technology.
2. **PSO2:** Create Social awareness & environmental wisdom along with ethical responsibility to lead a successful career and sustain passion using optimal resources to become an Entrepreneur.

PROGRAM OUTCOMES-PO's

Engineering graduates will be able to:

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern

engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change

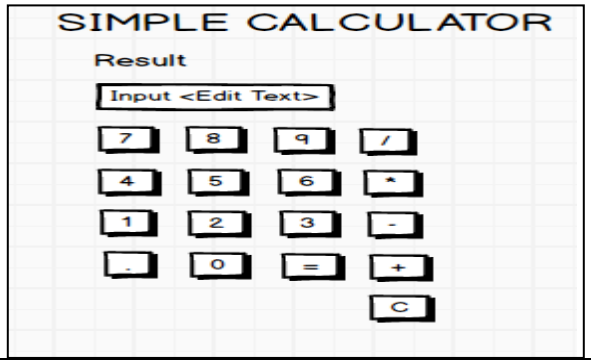
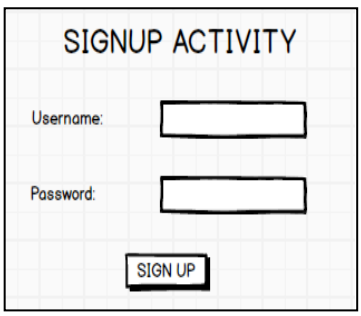
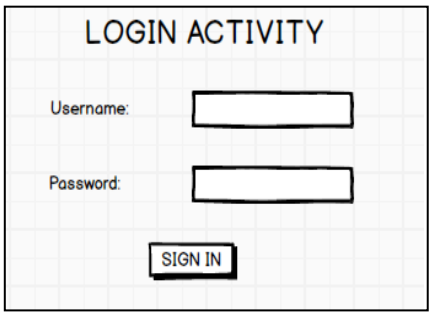
University Syllabus

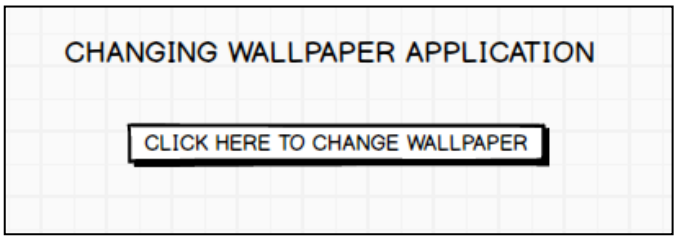
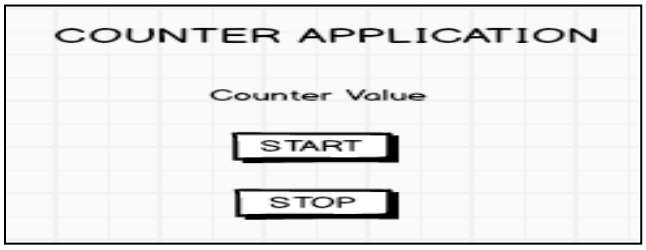
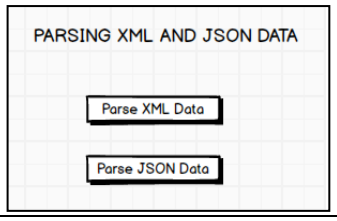
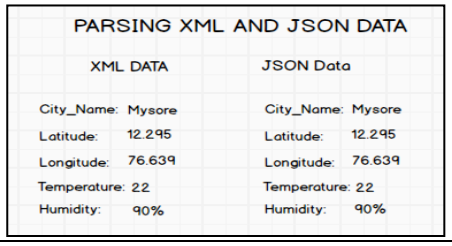
Course Title: MOBILE APPLICATION DEVELOPMENT (Effective from the academic year 2018 -2019) SEMESTER:VI			
Subject code	**18CSMP68**	IA Marks	**40**
Number of lecture hours/week	**3 Hours/Week**	Exam Marks	**60**
Total Number of Lecture Hours	**40**	Exam Hours	**03**
Credits -02		Total Marks-100	

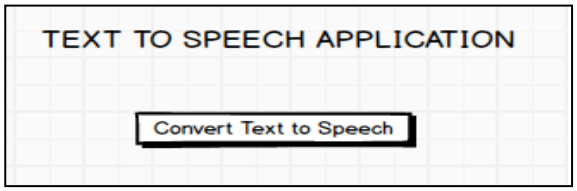
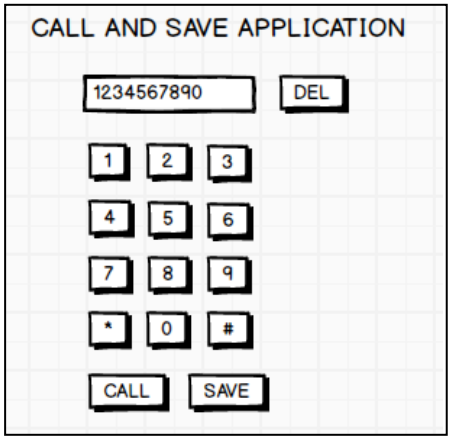
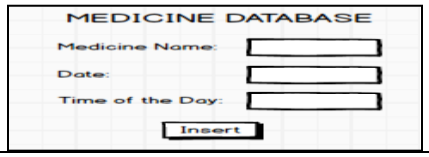
Course objectives: This laboratory (18CSMP68) will enable students to

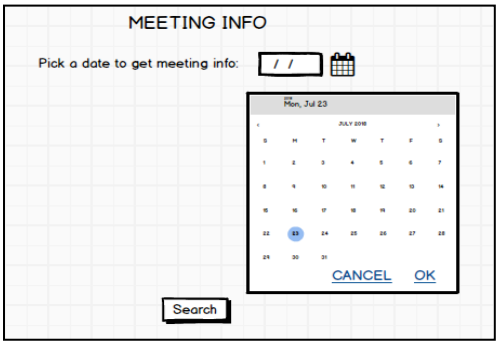
- Learn and acquire the art of Android Programming.
- ConfigureAndroid studio to run the applications.
- Understand and implement Android's User interface functions.
- Create, modify and query on SQlite database.
- Inspect different methods of sharing data using services.

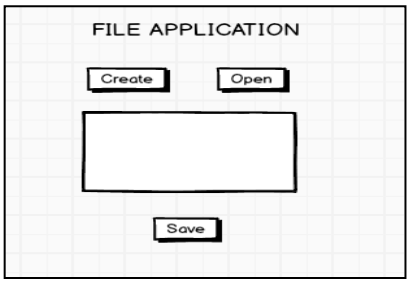
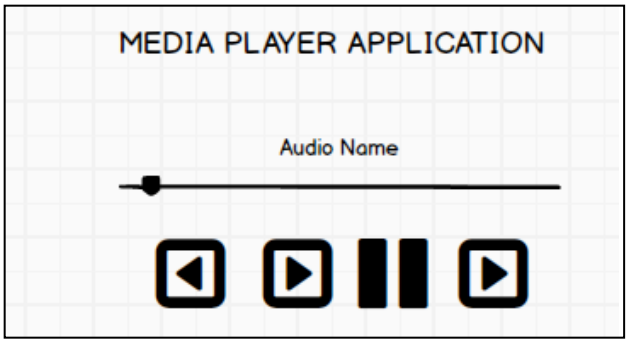
Sl.No.	Experiment	RBT
1	Create an application to design a Visiting Card. The Visiting card should have a company logo at the top right corner. The company name should be displayed in Capital letters, aligned to the center. Information like the name of the employee, job title, phone number, address, email, fax and the website address is to be displayed. Insert a horizontal line between the job title and the phone number. 	L3, L6

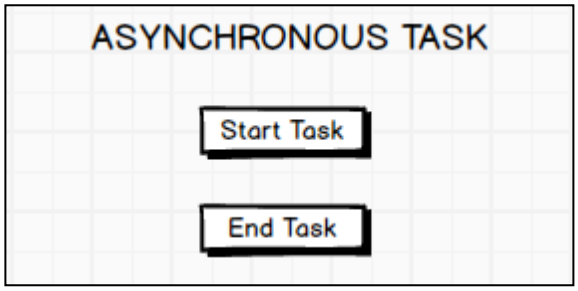
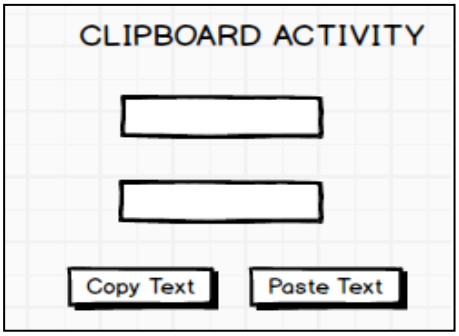
2	Develop an Android application using controls like Button, TextView, EditText for designing a Calculator having basic functionality like Addition, Subtraction, Multiplication, and Division 	L6
3	Create a SIGN Up activity with Username and Password. Validation of password should happen based on the following rules: • Password should contain uppercase and lowercase letter • Password should contain letters and numbers. • Password should contain special characters. • Minimum length of the password (the default value is 8). On successful **SIGN UP** proceed to the next Login activity. Here the user should **SIGN IN** using the Username and Password created during signup activity. If the Username and Password are matched then navigate to the next activity which displays a message saying "Successful Login" or else display a toast message saying "Login Failed". The user is given only two attempts and after that display a toast message saying "Failed Login Attempts" and disable the SIGN IN button. Use Bundle to transfer information from one activity to another.  	L6

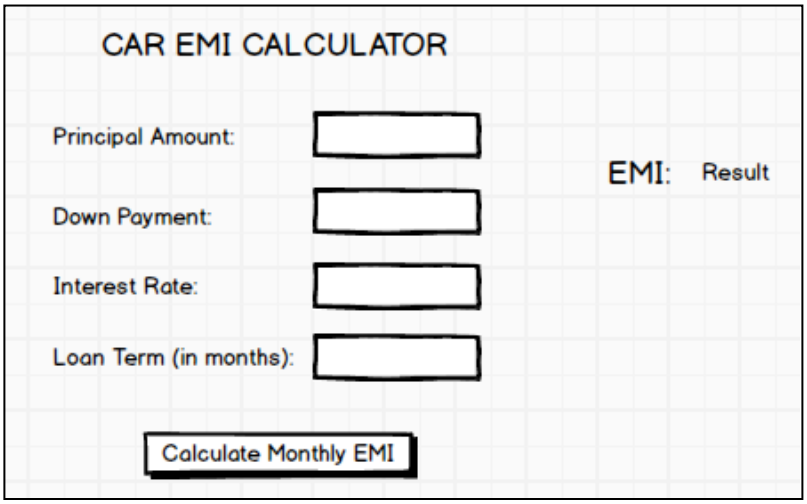
4	Develop an application to set an image as wallpaper. On click of a button, the wallpaper image should start to change randomly every 30 seconds. 	L3, L6
5	Write a program to create an activity with two buttons START and STOP. On Pressing of the START button, the activity must start the counter by displaying the numbers from One and the counter must keep on counting until the STOP button is pressed. Display the counter value in a TextView Control. 	L6
6	Create two files of XML and JSON type with values for City_Name, Latitude, Longitude, Temperature, and Humidity. Develop an application to create an activity with two buttons to parse the XML and JSON files which when clicked should display the data in their respective layouts side by  	L6

7	Develop a simple application with one Edit Text so that the user can write some text in it. Create a button called "Convert Text to Speech" that converts the user input text into voice. 	L3, L6
8	Create an activity like a phone dialer with CALL and SAVE buttons. On pressing the CALL button, it must call the phone number and on pressing the SAVE button it must save the number to the phone contacts. 	L3, L6
	Part B	
1	Write a program to enter Medicine Name, Date and Time of the Day as input from the user and store it in the SQLite database. Input for Time of the Day should be either Morning or Afternoon or Evening or Night. Trigger an alarm based on the Date and Time of the Day and display the Medicine Name. 	L3,L6

2	Develop a content provider application with an activity called "Meeting Schedule" which takes Date, Time and Meeting Agenda as input from the user and store this information into the SQLite database. Create another application with an activity called "Meeting Info" having DatePicker control, which on the selection of a date should display the Meeting Agenda information for that particular date, else it should display a toast message saying "No Meeting on this Date".  	L3,L6
3	Create an application to receive an incoming SMS which is notified to the user. On clicking this SMS notification, the message content and the number should be displayed on the screen. Use appropriate emulator control to send the SMS message to your application. 	L3,L6

4	Write a program to create an activity having a Text box, and also Save, Open and Create buttons. The user has to write some text in the Text box. On pressing the Create button the text should be saved as a text file in MkSDcard. On subsequent changes to the text, the Save button should be pressed to store the latest content to the same file. On pressing the Open button, it should display the contents from the previously stored files in the Text box. If the user tries to save the contents in the Textbox to a file without creating it, then a toast message has to be displayed saying "FirstCreate a File". 	L3,L6
5	Create an application to demonstrate a basic media player that allows the user to Forward, Backward, Play and Pause an audio. Also, make use of the indicator in the seek bar to move the audio forward or backward as required. 	L3,L6

6	Develop an application to demonstrate the use of Asynchronous tasks in android. The asynchronous task should implement the functionality of a simple moving banner. On pressing the **Start Task** button, the banner message should scroll from right to left. On pressing the **Stop Task** button, the banner message should stop. Let the banner message be "Demonstration of Asynchronous Task". 	L3,L6
7	Develop an application that makes use of the clipboard framework for copying and pasting of the text. The activity consists of two EditText controls and two Buttons to trigger the copy and paste functionality. 	L3,L6
8	Create an AIDL service that calculates Car Loan EMI. The formula to calculate EMI is **$E = P * (r(1+r)^n)/((1+r)^n-1)$** where E = The EMI payable on the car loan amount P = The car Loan Priciple amount r = The interest rate value computed on a monthly basis n = The loan tenure in the form of months	L3,L6

	The down payment amount has to be deducted from the principal amount paid towards buying the Car. Develop an application that makes use of this AIDL service to calculate the EMI. This application should have four EditText to read the Principal Amount, Down Payment, Interest Rate, Loan Term (in months) and a button named as "Calculate Monthly EMI". On click of this button, the result should be shown in a TextView. Also, calculate the EMI by varying the Loan Term and Interest Rate values.  	

COURSE OUTCOMES

On successful completion of this course students will be able to,

CO1	Create, test and debug Android application by setting up Android development environment.
CO2	Implement adaptive, responsive user interfaces that work across a wide range of devices.
CO3	Demonstrate methods in storing, sharing and retrieving data in Android applications.
CO4	Infer the role of permissions and security for Android applications.

CO-PO Mapping

CO No.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2	-	2	1	-	-	1	1	-	1
CO2	3	2	2	-	2	1	-	-	1	1	-	1
CO3	3	2	2	-	2	1	-	-	1	1	-	1
CO4	3	2	2	-	-	-	-	2	1	1	-	1

General Instructions for the Laboratory

Do's

- It is mandatory for all the students to attend all practical classes & complete the experiments as per syllabus.
- Students should strictly follow the lab timings, dress code with Apron & ID cards.
- Should maintain a neat observation book.
- Study the theory and logic before executing the program.
- Submit the completed lab records of executed programs and update the index book in every lab session.
- Should prepare for viva questions regularly.
- Handle the computer systems carefully.
- Maintain discipline and silence in the lab.

Don'ts

- Should not take Bags and Mobile phones into the Laboratory.
- Do not wear footwear inside the Laboratory
- Systems & Components should be handled carefully failing to which penalty will be imposed.
- Do not switch off the system abruptly.
- Should not chew gum or eat in the lab.

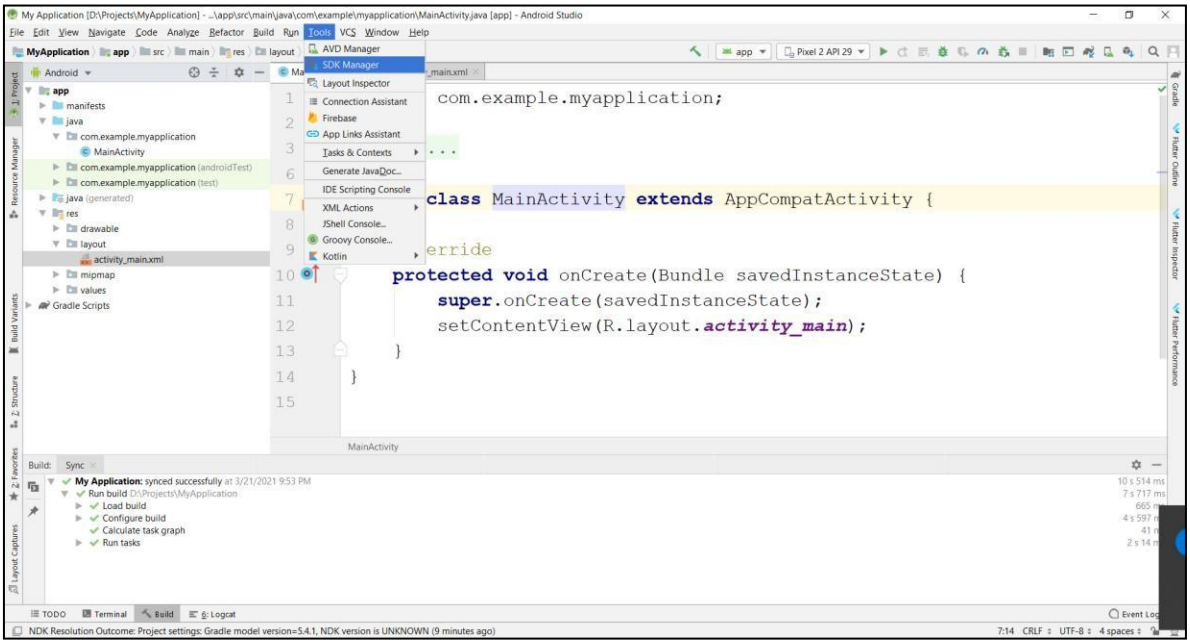
1. Android Studio Tutorials

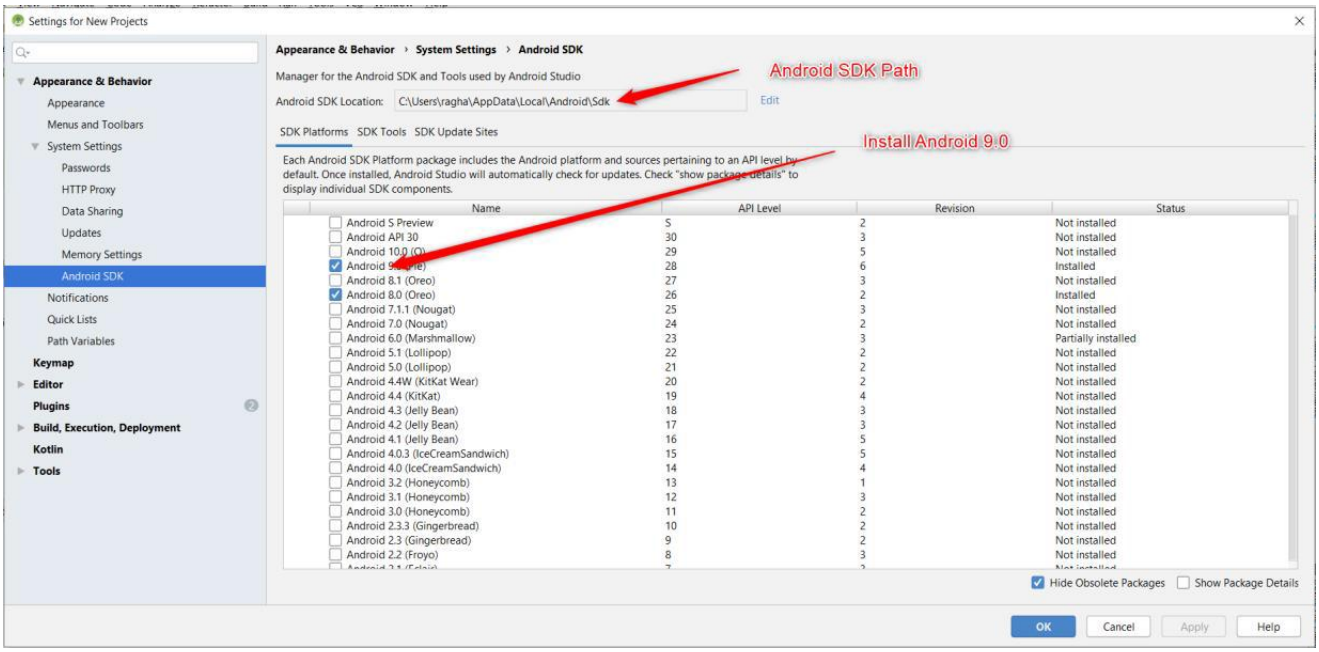
1.1 Install Android Studio and Packages:

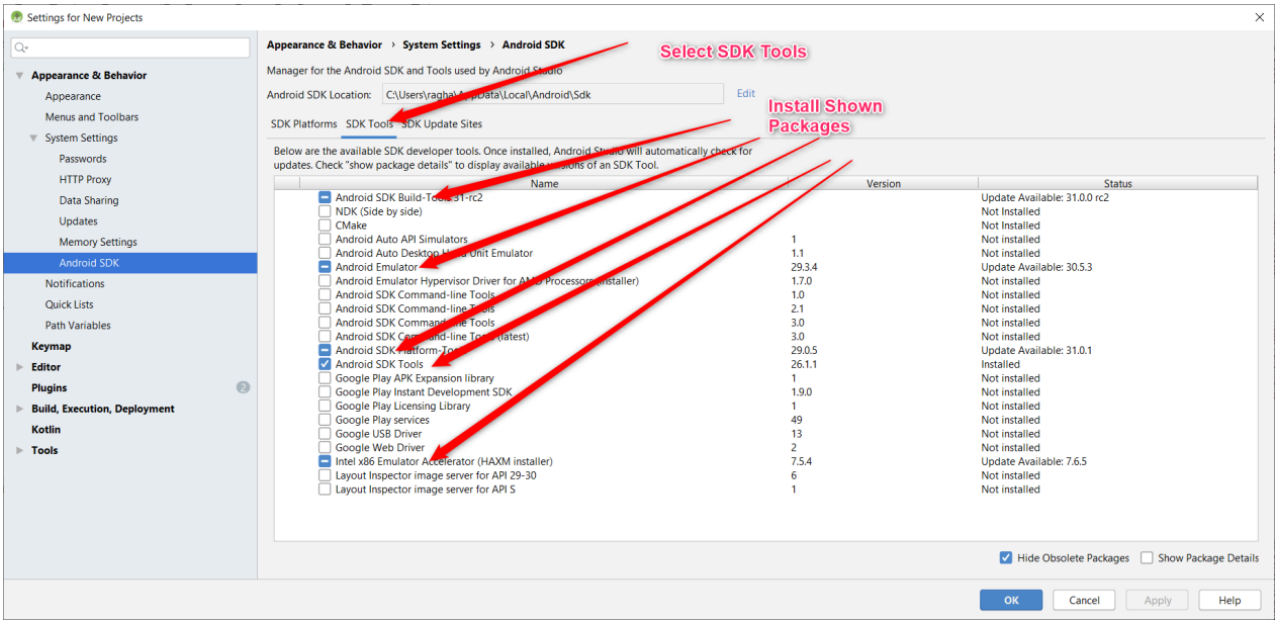
Download Android Version 4.0.2 from the below link
https://redirector.gvt1.com/edgedl/android/studio/install/4.0.2.0/android-studio-ide-193.6821437-windows.exe

1.2 Configure Android SDK packages:

Go to Tools →□SDK Manager

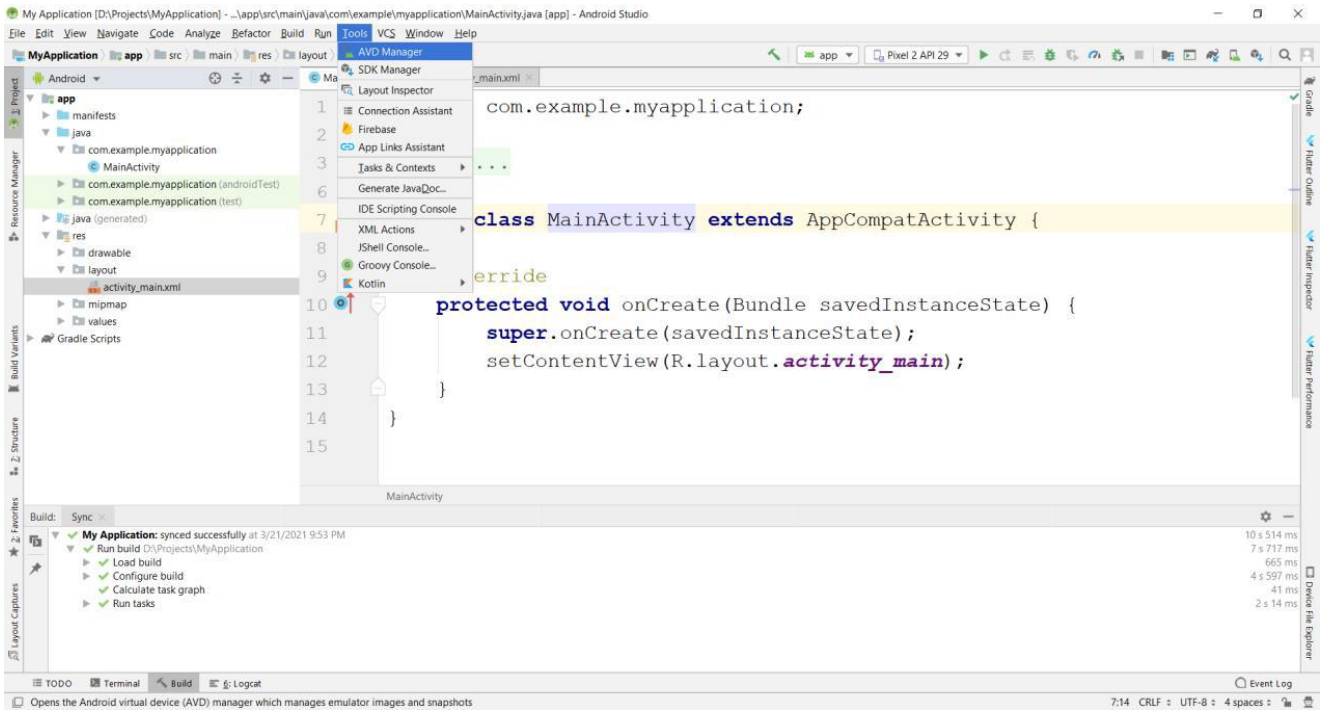


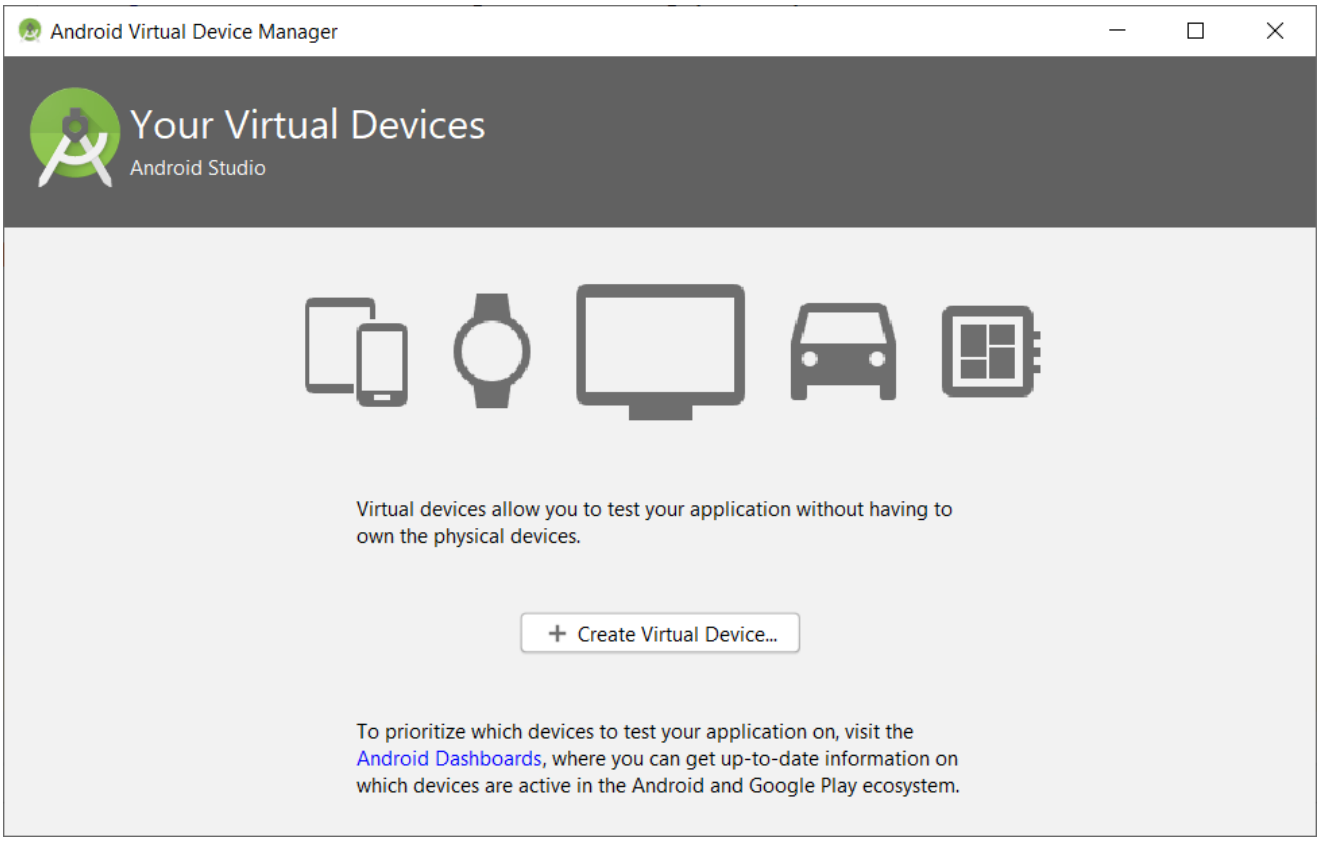




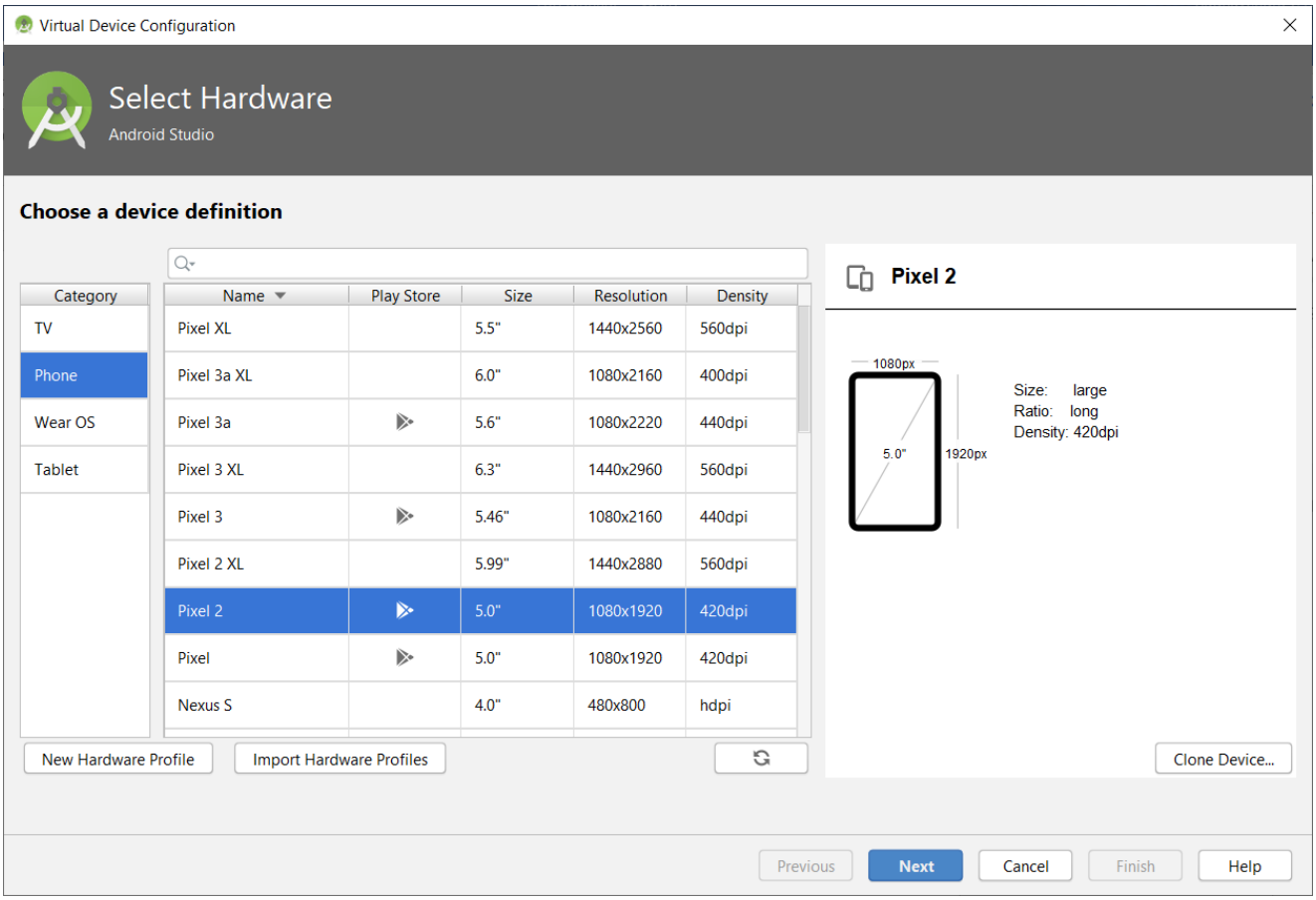
1.3 Creating Emulator

Go to Tools → Select AVD Manager

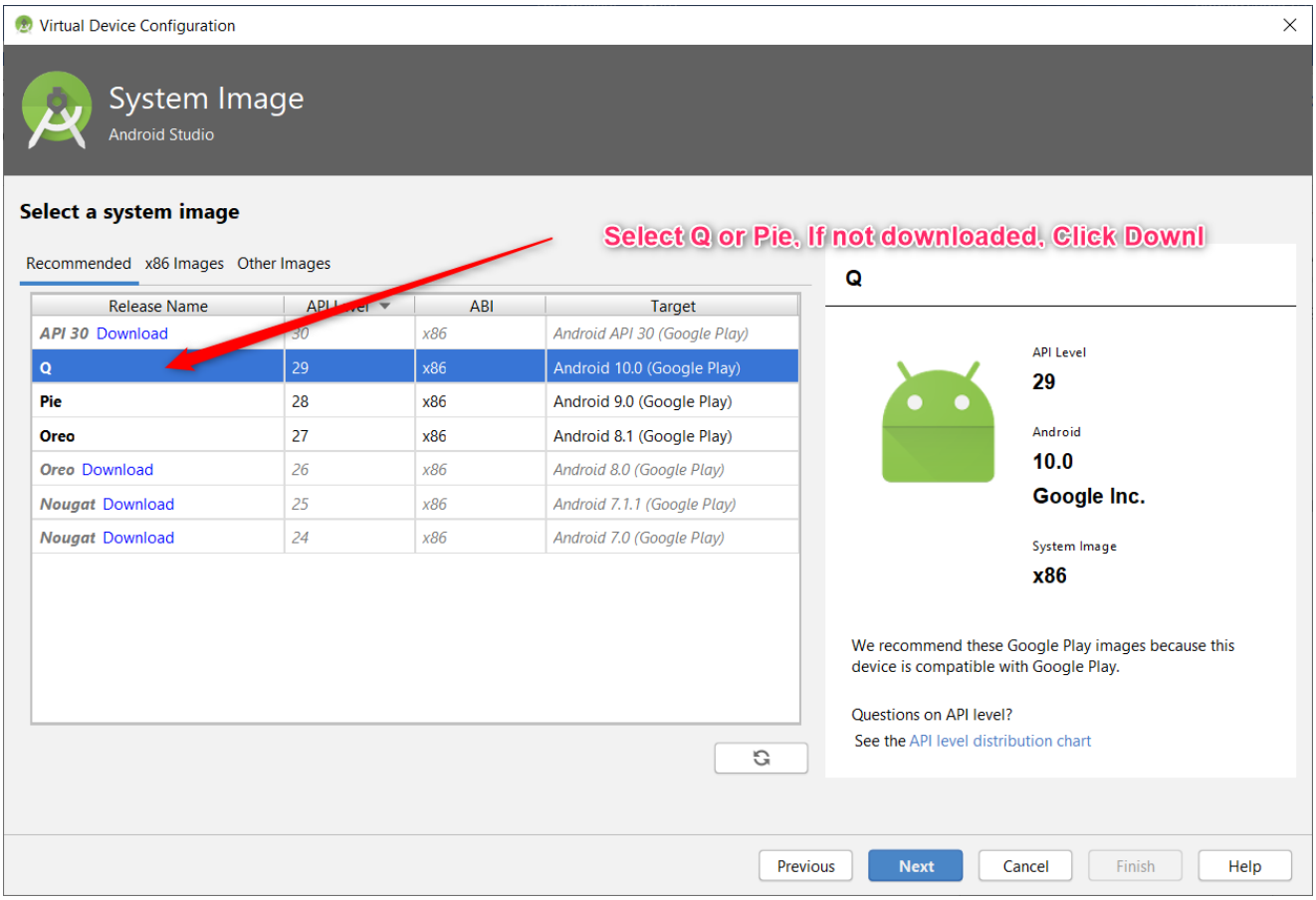




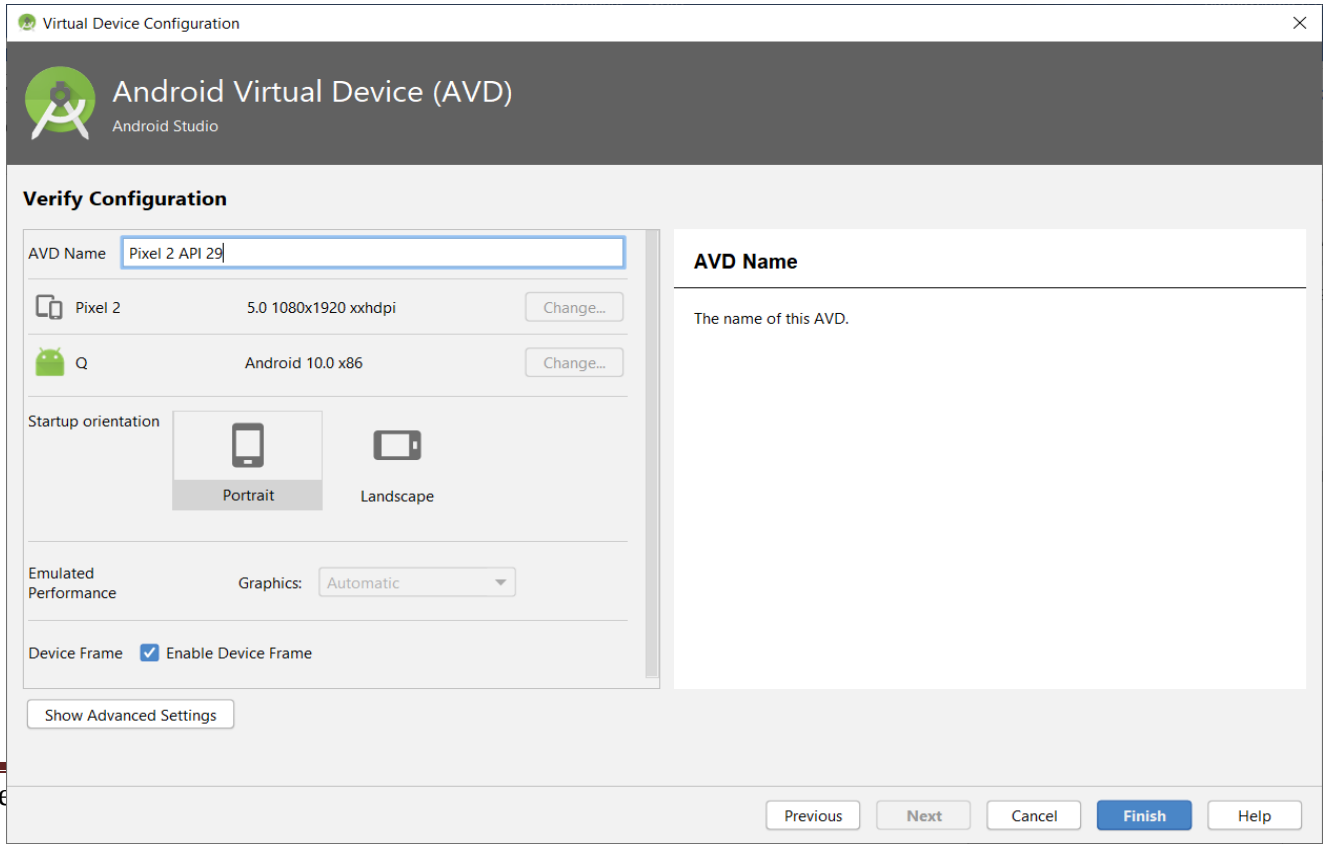
Select Create Virtual Device →Select Phone →Pixel 2 → Press Next



Select Android Q, if not already downloaded press download, After download completes Select Q and Press Next Button.

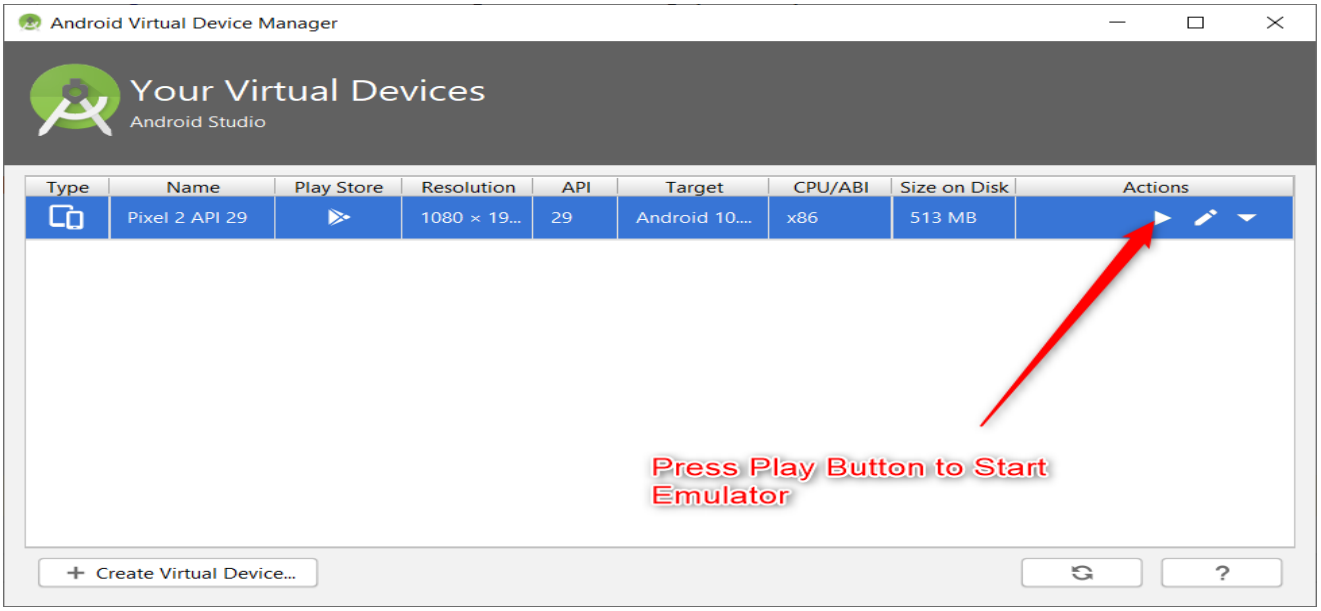


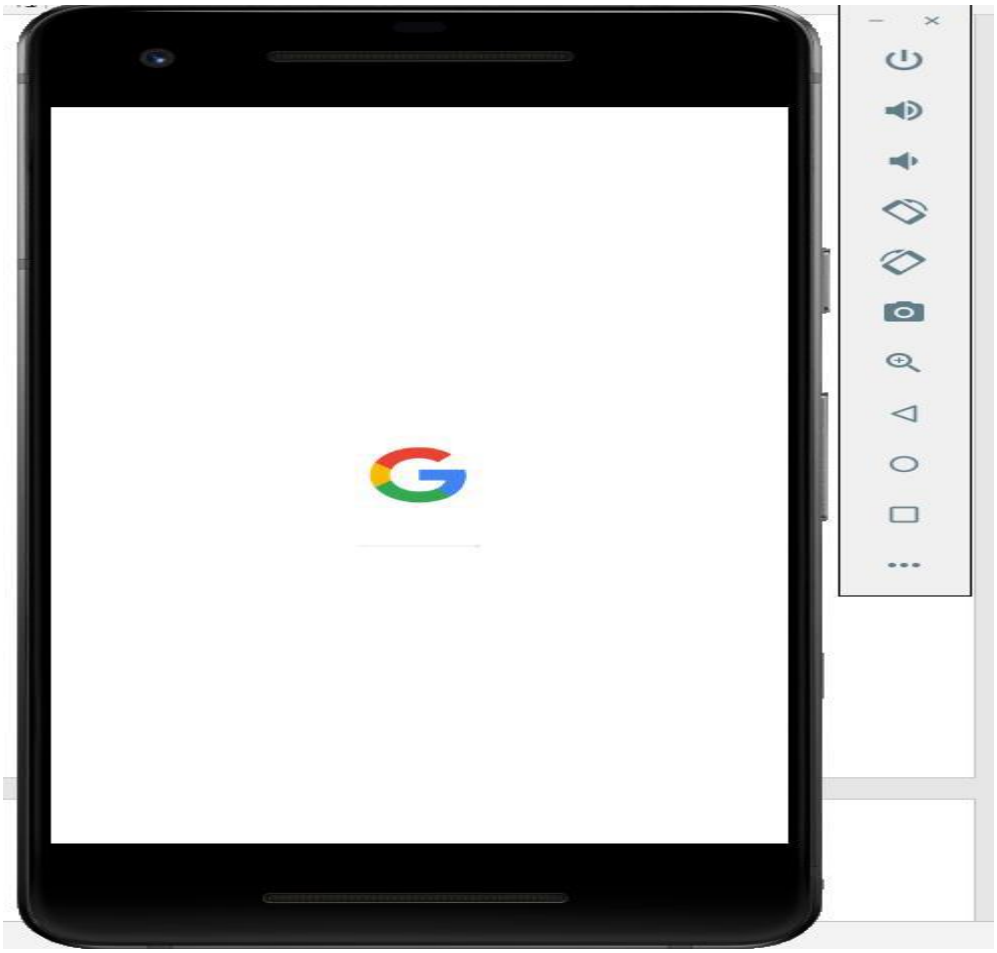
Enter AVD Name and Press Finish.



De

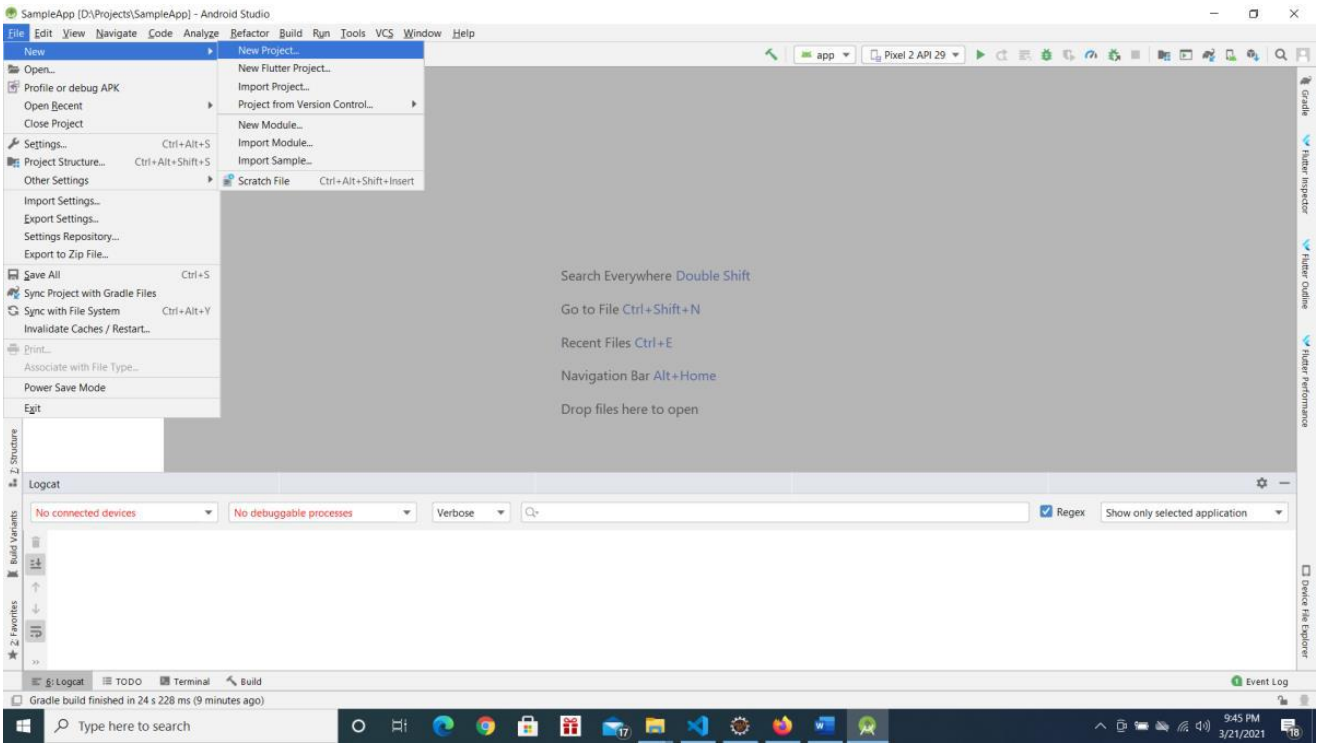
Press Play Button to Start Emulator



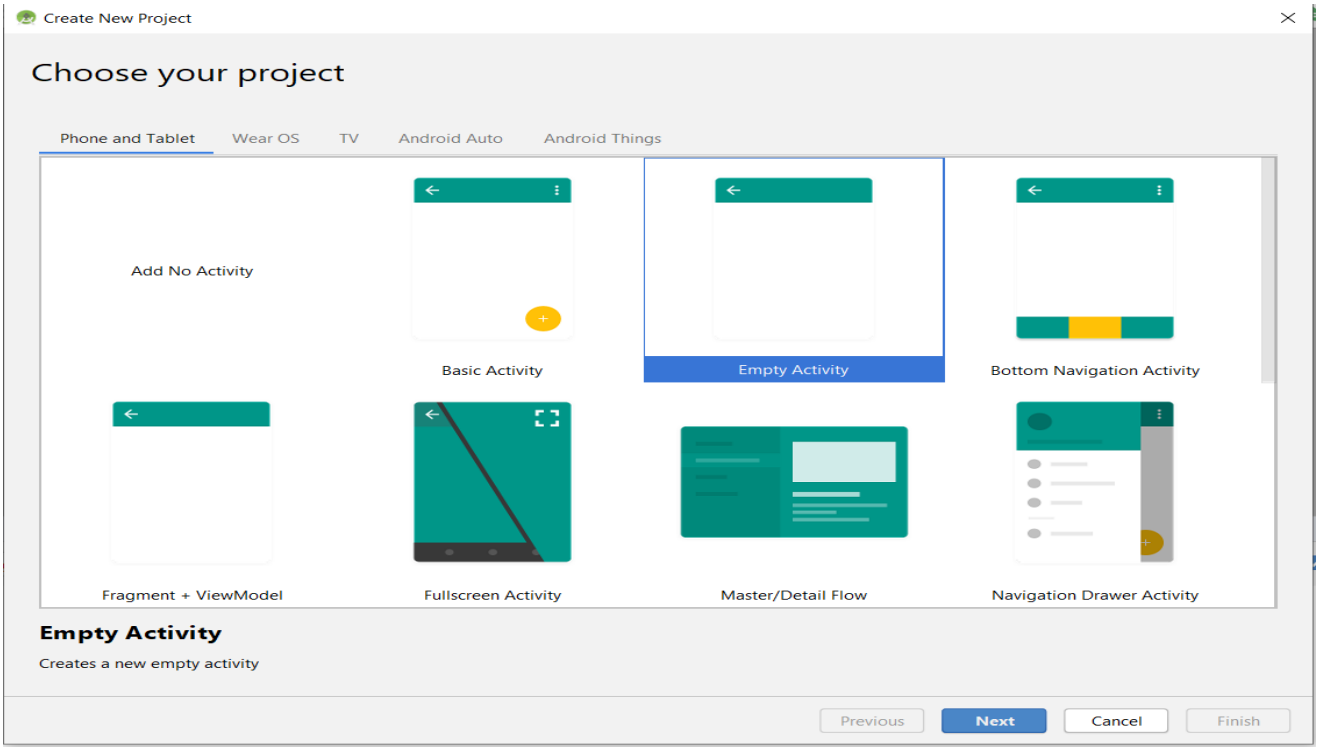


1.4 Creating a New Project in Android

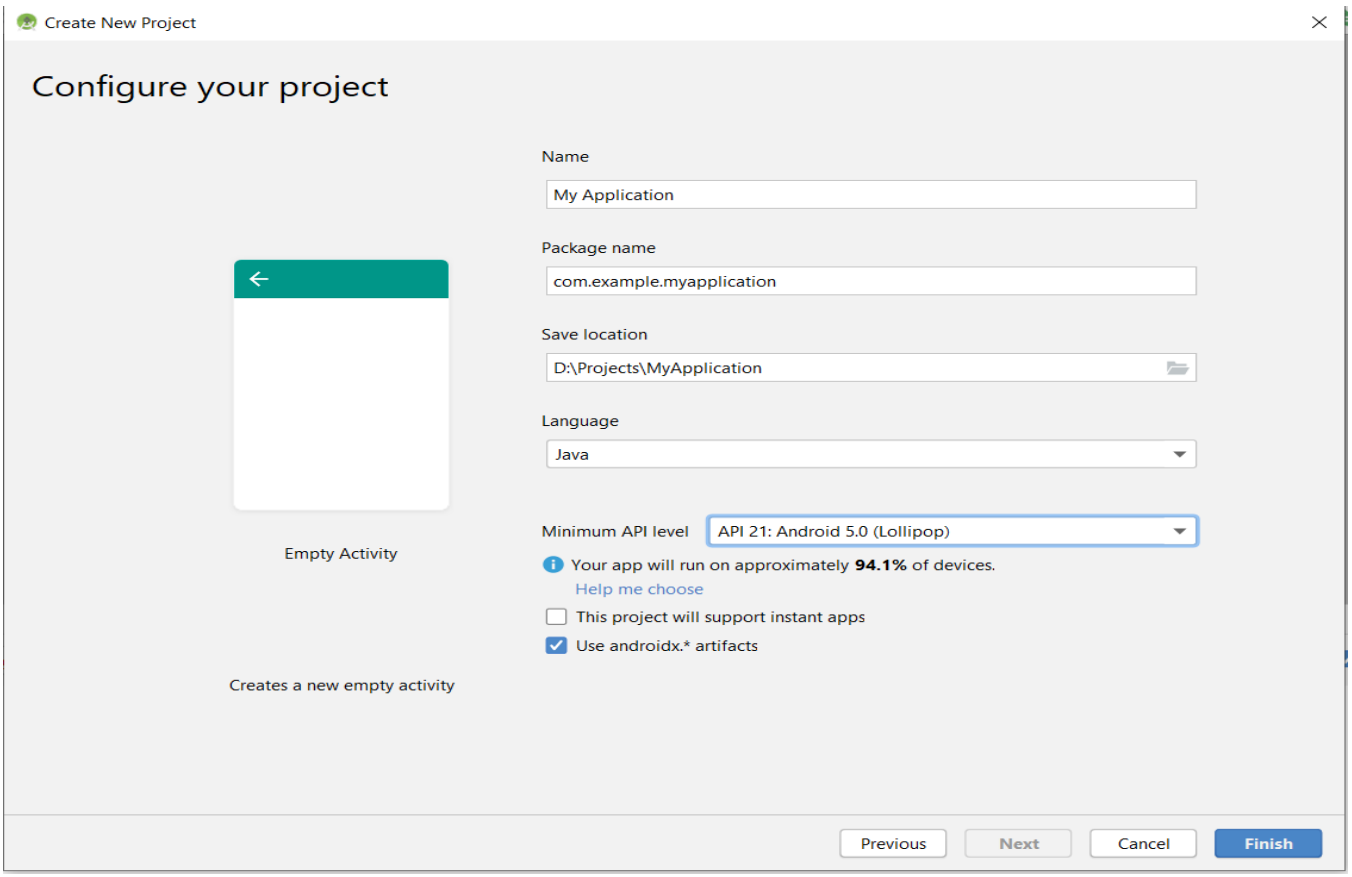
While creating a New Project for First Time, make sure Android Studio is connected to internet, It downloads the required packages from internet.
Go to File →New →New Project



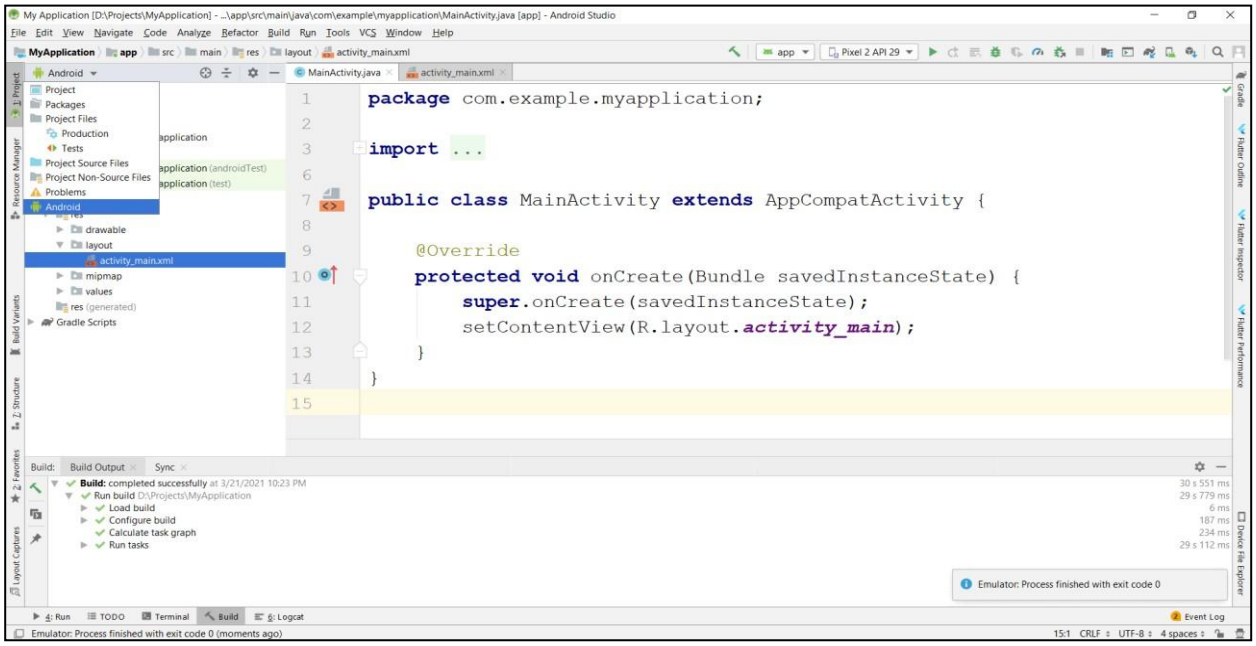
Choose Phone and Tablet → Empty Activity → Press Next



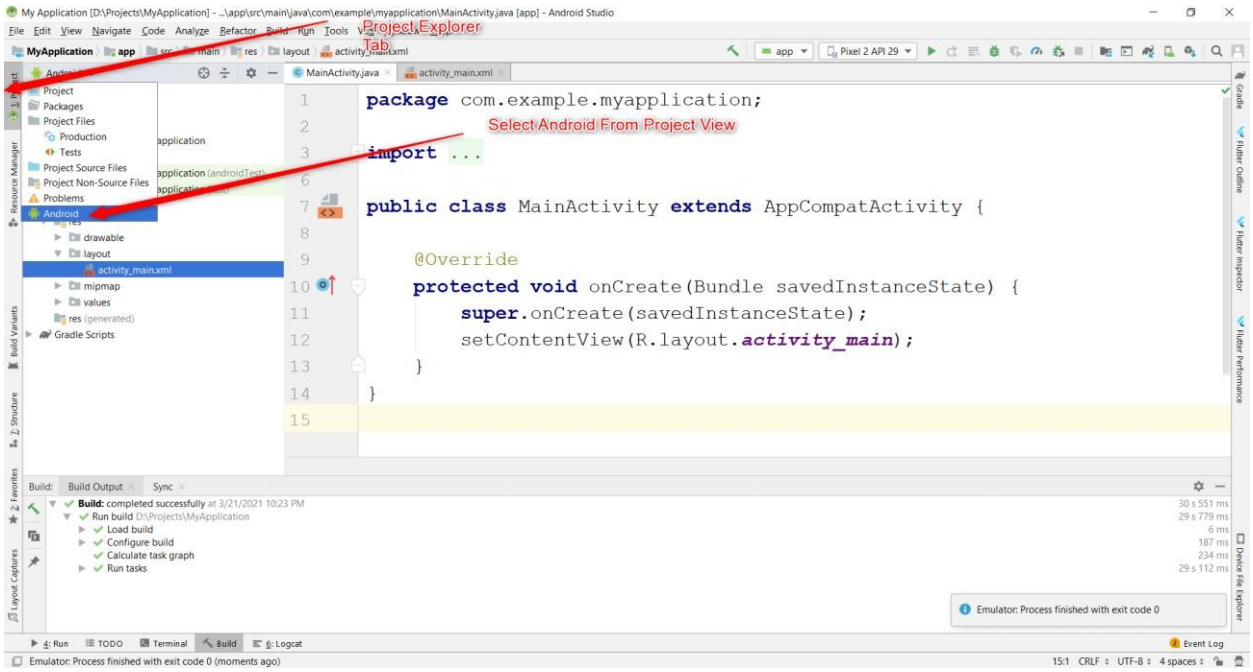
In Configure your Project Screen, Enter below details and Press Finish Button.
Enter Name of the Application → This will be application name this will be visible with Home Screen Icon.
Package Name → Enter package name atleast two identifier (Eg: com.example). Best Practice is 3 or more identifier (Eg: com.example.firstapp).
Save Location →Location where to save the Project
Language → Choose Java
Minimum API Level → Android 5.0
Select Checkbox Use androidx.artifacts folder as below screenshot.



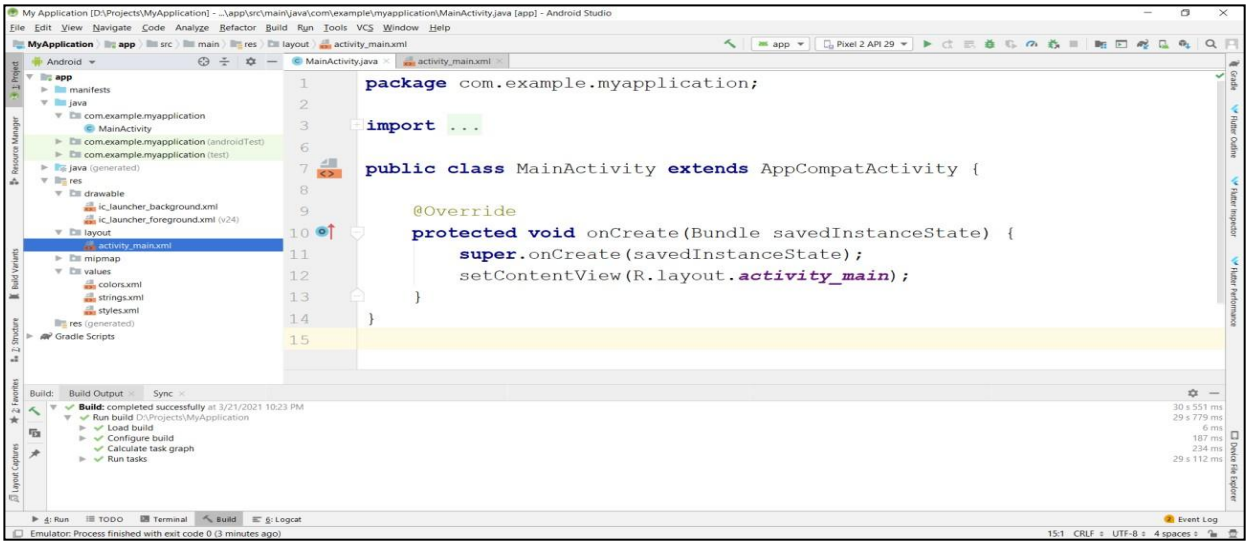
1.5 Android Project Structure:



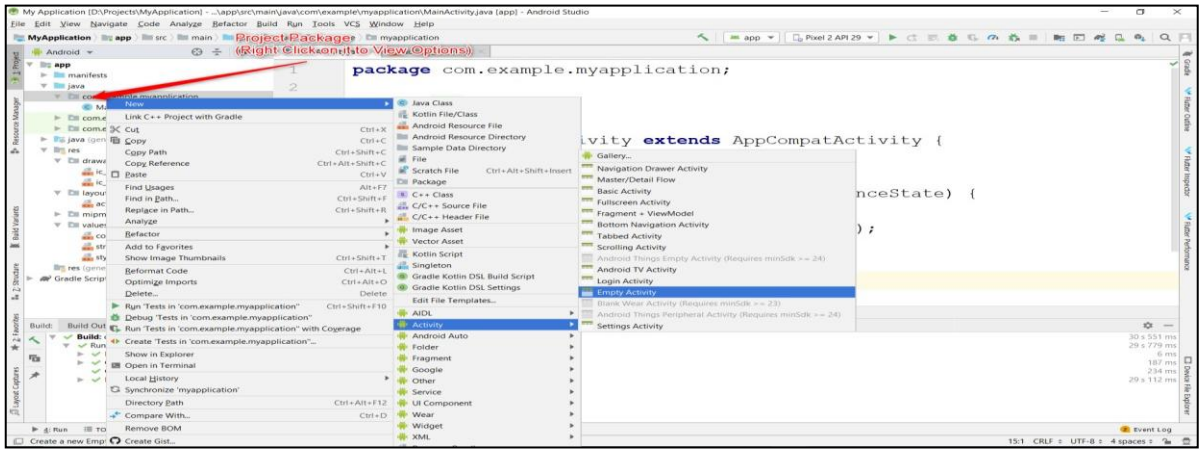
Select Project Explorer and Select Android from Project View

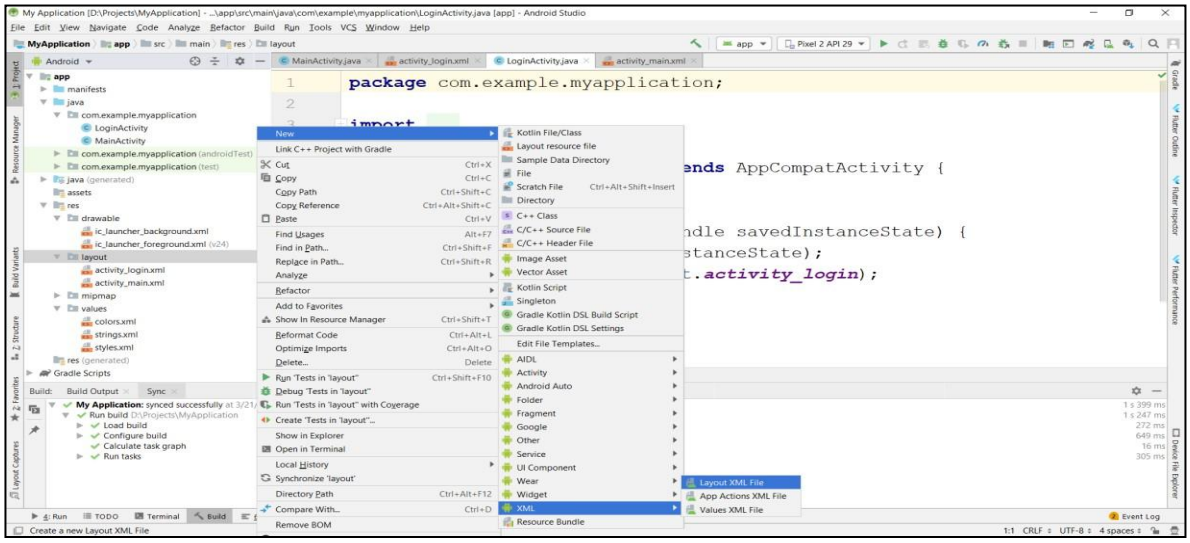


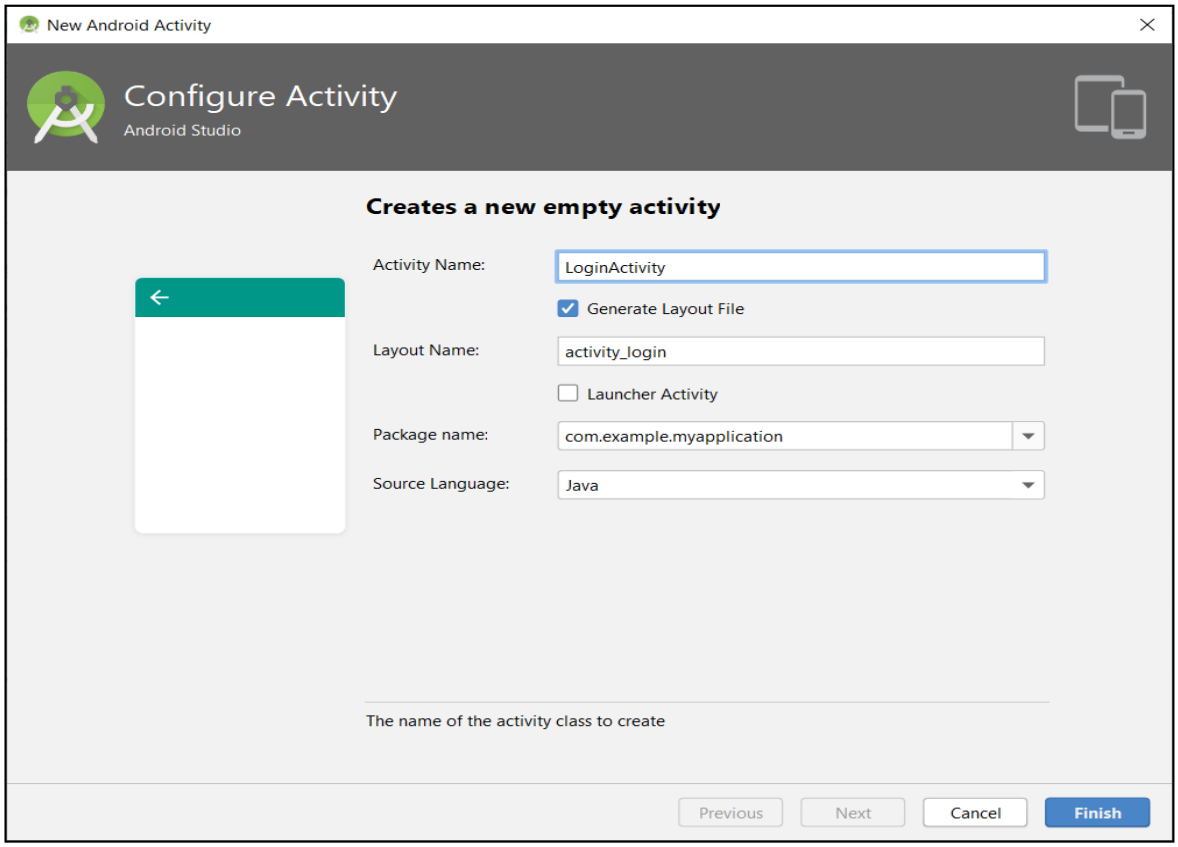
Basic View:

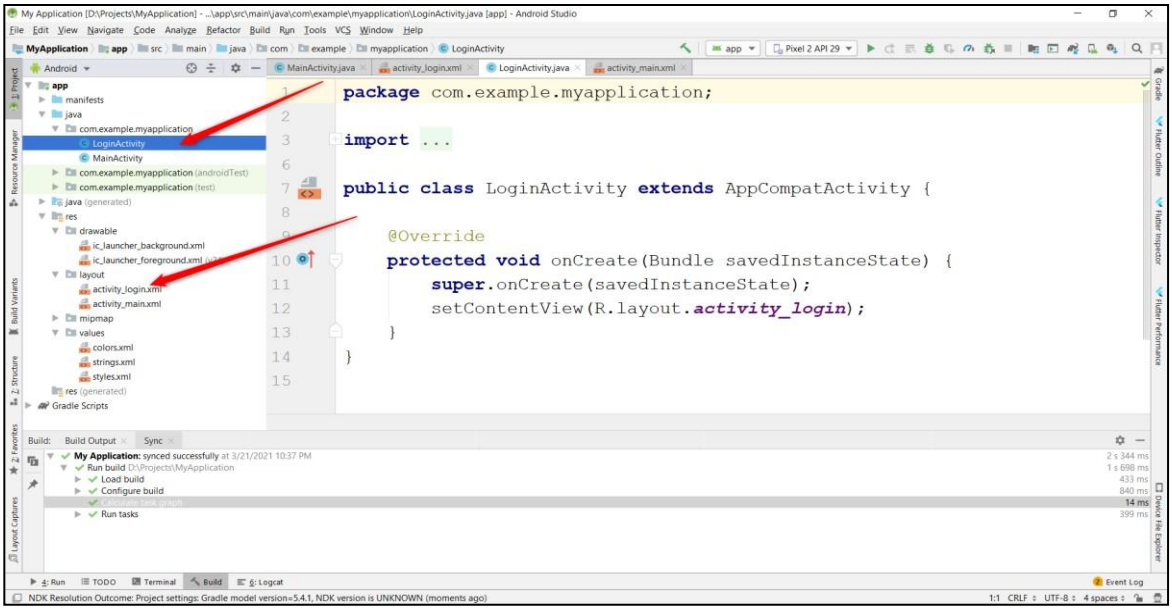


2. Importing an Existing Project in Android Studio
3. Creating an Activity in Android
Right Click on Package → New → Activity→ Empty Activity




package com.example.myapplication;
import
ends AppCompatActivity {
ndle savedInstanceState) {
stanceState);
t.activity_login);
New
Kotlin File/Class
Layout resource file
Sample Data Directory
File
Scratch File
Directory
C++ Class
C/C++ Source File
C/C++ Header File
Image Asset
Vector Asset
Kotlin Script
Singleton
Gradle Kotlin DSL Build Script
Gradle Kotlin DSL Settings
Edit File Templates...
AIDL
Activity
Android Auto
Folder
Fragment
Google
Other
Service
UI Component
Wear
Widget
XML
Resource Bundle
Layout XML File
App Actions XML File
Values XML File


New Android Activity
Configure Activity
Android Studio
Creates a new empty activity
Activity Name:
LoginActivity
Generate Layout File
Layout Name:
activity_login
Launcher Activity
Package name:
com.example.myapplication
Source Language:
Java
The name of the activity class to create
Previous
Next
Cancel
Finish

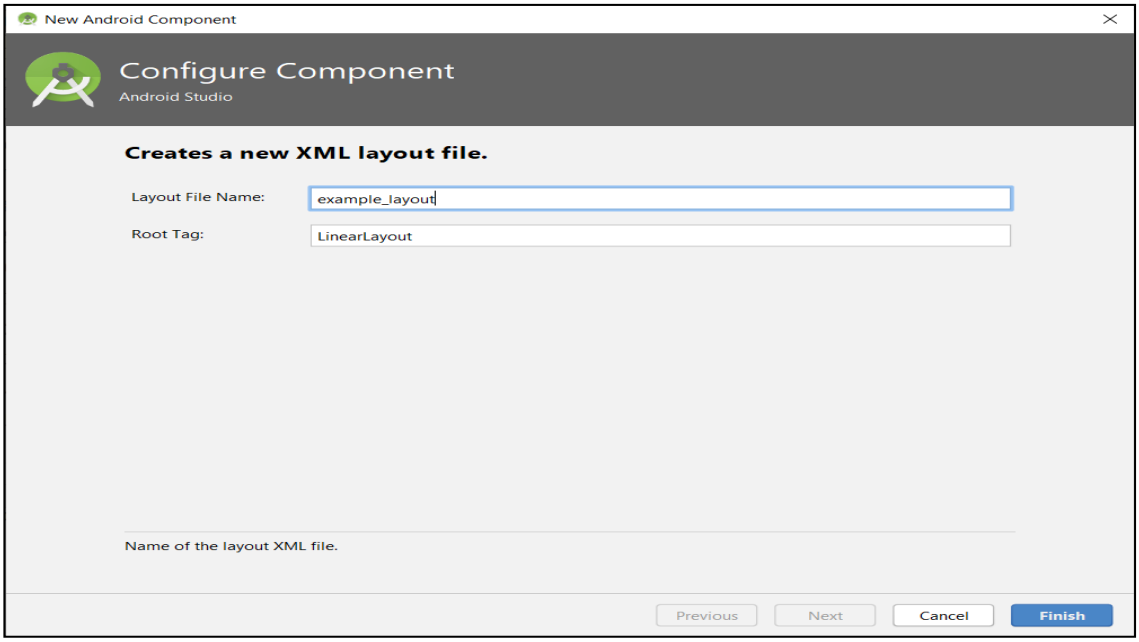


Enter Activity Name and Press Finish

1.6 Creating a Layout in Android

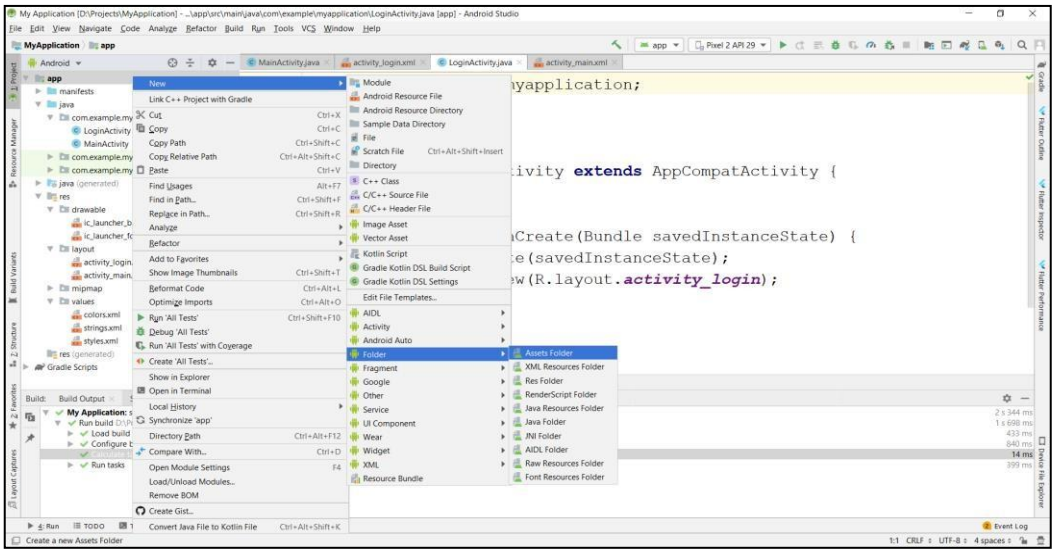
Right Click on Layout Folder → New → XML→Layout XML File

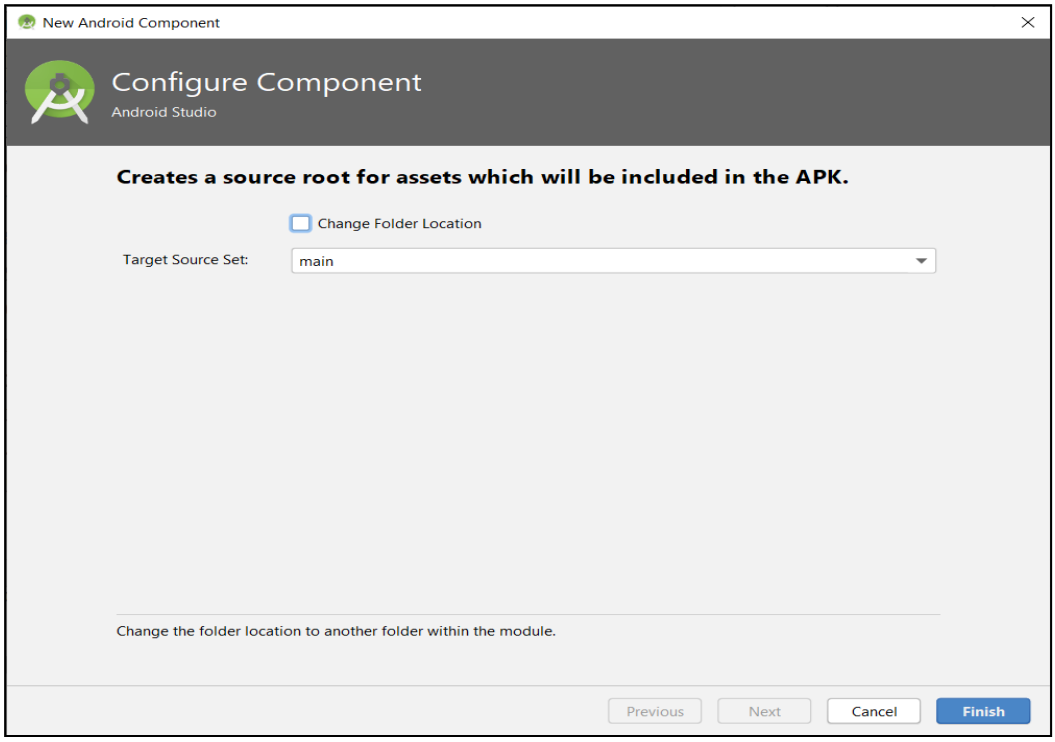
Enter xml file name and press Finish

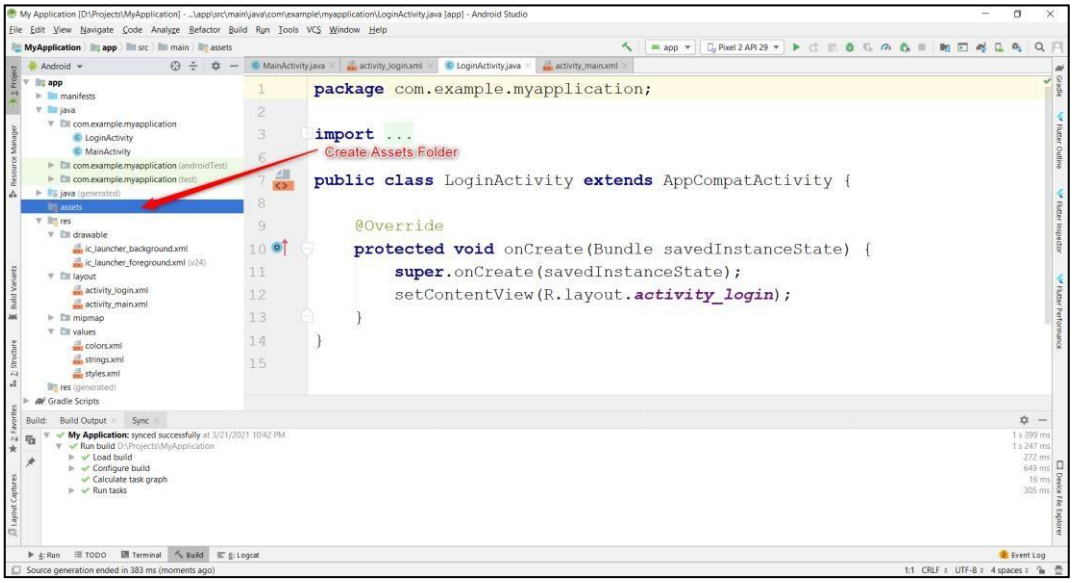


1.7 Creating Assets Folder in Android

Right Click on app folder → New → Folder → Assets Folder → Press Finish Button

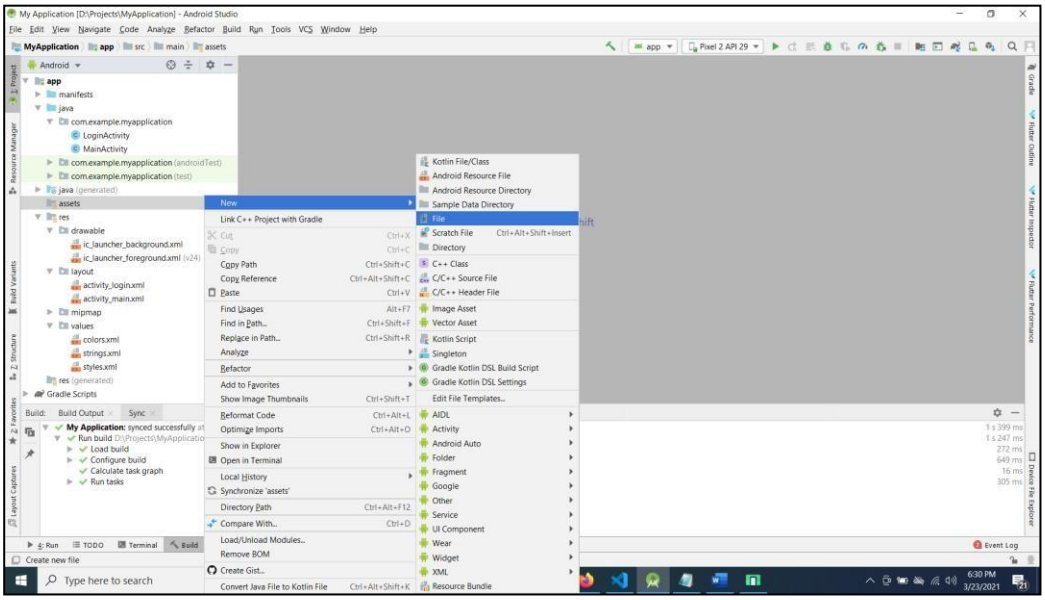




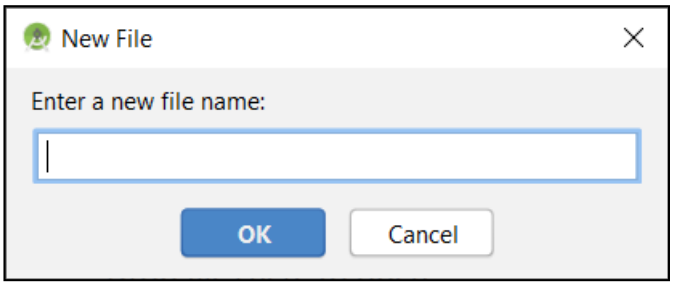


1.8 Creating File in assets Folder:

Right Click on assets folder → New → File



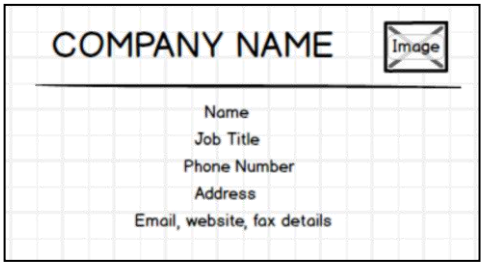
Enter filename with extension (Eg: abc.xml)



PART A

Program 1

Create an application to design a Visiting Card. The Visiting card should have a company logo at the top right corner. The company name should be displayed in Capital letters, aligned to the center. Information like the name of the employee, job title, phone number, address, email, fax and the website address is to be displayed. Insert a horizontal line between the job title and the phone number



1. Create a New Android Project with Empty Activity.
2. Open activity_main.xml file from res→ layout folder, check/add Linear Layout as the root view.
3. Create layout using nested Relative Layout and TextView.
4. Use View background property to draw the line
5. Add Image to drawable folder and reference the image in the layout using @drawable/<image_name>
6. Use android:layout_gravity/android:gravity properties to center the components.

MainActivity.java

```
package com.example.program1;

import androidx.appcompat.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);
```

```
        }
}
```

activity_main.xml:

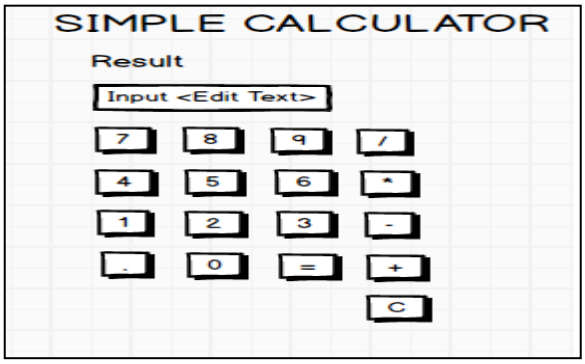
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout        xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity"
    android:paddingLeft="20dp"
    android:paddingTop="20dp"
    android:paddingRight="20dp">

    <RelativeLayout
        android:layout_width="wrap_content
        "
        android:layout_height="wrap_content"
        android:layout_gravity="center">

        <TextView
            android:layout_width="wrap_conten
            t"
            android:layout_height="wrap_content"
            android:text="COMPANY    NAME"
            android:textSize="16dp"
            android:textColor="#000000"
            android:id="@+id/lbl_company_na
            me"
            android:layout_marginRight="10dp"
            />

        <ImageView
            android:layout_width="50dp"
            android:layout_height="50dp"
            android:src="@drawable/email_icon"
            android:layout_toRightOf="@id/lbl_company_name"
            android:layout_marginLeft="10dp"/>

</RelativeLayout>


<View
```

```
        android:layout_width="match_parent"
        android:layout_height="2dp"
        android:background="#000000"/>

    <TextView
        android:layout_width="match_parent
        "
        android:layout_height="wrap_content"
        android:text="Name:"
        android:textSize="16dp"
        android:layout_marginBottom="10dp
        "   android:layout_marginTop="10dp"
        android:textColor="#000000"
        android:gravity="center"
        />


    <TextView
        android:layout_width="match_parent
        "
        android:layout_height="wrap_content"
        android:text="JobTitle:"
        android:textSize="16dp"
        android:layout_marginBottom="10dp
        "   android:layout_marginTop="10dp"
        android:textColor="#000000"
        android:gravity="center"
        />


    <TextView
        android:layout_width="match_parent
        "
        android:layout_height="wrap_content"
        android:text="PhoneNumber:"
        android:textSize="16dp"
        android:layout_marginBottom="10dp
        "   android:layout_marginTop="10dp"
        android:textColor="#000000"
        android:gravity="center"
        />


    <TextView
        android:layout_width="match_parent
        "
        android:layout_height="wrap_content"
        android:text="Address:"
```

```
            android:textSize="16dp"
            android:layout_marginBottom="10dp
            "   android:layout_marginTop="10dp"
            android:textColor="#000000"
            android:gravity="center"
            />


        <TextView
            android:layout_width="match_parent
            "
            android:layout_height="wrap_content"
            android:text="Email,  Website,  Fax"
            android:textSize="16dp"
            android:layout_marginBottom="10dp
            " android:layout_marginTop="10dp"
            android:textColor="#000000"
            android:gravity="center"
            />

</LinearLayout>
```

Program 2

Develop an Android application using controls like Button, TextView, EditText for designing a Calculator having basic functionality like Addition, Subtraction, Multiplication, and Division.



1. Create a New Android Project with Empty Activity.
2. Open activity_main.xml file from res→ layout folder, check/add Constraint Layout as the root view.
3. Create Layout using Drag and Drop framework.
4. Open MainActivty.java file, Override onCreate() method and bring activity_main.xml file on screen using setContentView() and bring the view references using findViewById() method.
5. Add Listeners to Button Click Event:
6. Create a class which implements OnClickListener interface.
7. Override onClick() method of OnClickListener Interface.
8. Register the button for click event by calling setOnClickListener() method of View class and pass the object of the class that implemented OnClickListener Interface.
9. Create a logic to Add/Subtract/Multiply/Divide to perform arithmetic operation on 2 operands (Eg: 10+20), If more than 2 operands or wrong input, display invalid input messages.

Design

MainActivity.java

```
package com.example.partaprogram2;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import java.util.regex.Pattern;
```

```
public class MainActivity extends AppCompatActivity
implements  View.OnClickListener {

 Button btnOne, btnTwo, btnThree, btnFour, btnFive,
btnSix;  Button btnSeven, btnEight, btnNine, btnZero;
 Button btnAdd,btnSub,btnMul,btnDiv;
 Button btnClear,btnEqual,btnDot;

 EditText txtResult;

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

 btnOne=(Button)findViewById(R.id.button_one);
 btnOne.setOnClickListener(this);

 btnTwo=(Button)findViewById(R.id.button_two);
 btnTwo.setOnClickListener(this);

 btnThree=(Button)findViewById(R.id.button_three);
 btnThree.setOnClickListener(this);


 btnFour=(Button)findViewById(R.id.button_four);
 btnFour.setOnClickListener(this);

 btnFive=(Button)findViewById(R.id.button_five);
 btnFive.setOnClickListener(this);

 btnSix=(Button)findViewById(R.id.button_six);
 btnSix.setOnClickListener(this);

 btnSeven=(Button)findViewById(R.id.button_seven);
 btnSeven.setOnClickListener(this);

 btnEight=(Button)findViewById(R.id.button_eight);
 btnEight.setOnClickListener(this);

 btnNine=(Button)findViewById(R.id.button_nine);
 btnNine.setOnClickListener(this);

 btnAdd=(Button)findViewById(R.id.button_add);
 btnAdd.setOnClickListener(this);

 btnSub=(Button)findViewById(R.id.button_sub);
 btnSub.setOnClickListener(this);

 btnMul=(Button)findViewById(R.id.button_mul);
 btnMul.setOnClickListener(this);
```

```
btnDiv=(Button)findViewById(R.id.button_div);
 btnDiv.setOnClickListener(this);


 btnClear=(Button)findViewById(R.id.button_clear);
 btnClear.setOnClickListener(this);

 btnEqual=(Button)findViewById(R.id.button_equal);
 btnEqual.setOnClickListener(this);

 btnDot=(Button)findViewById(R.id.button_dot);
 btnDot.setOnClickListener(this);

 txtResult=(EditText)findViewById(R.id.txt_result);

 txtResult.setText("");


 }

 public void onClick(View v)
 {
 if(v.equals(btnOne))
 {
 txtResult.append("1");
 }
 if(v.equals(btnTwo))
 {
 txtResult.append("2");
 }
 if(v.equals(btnThree))
 {
 txtResult.append("3");
 }
 if(v.equals(btnFour))
 {
 txtResult.append("4");
 }
 if(v.equals(btnFive))
 {
 txtResult.append("5");
 }
 if(v.equals(btnSix))
 {
 txtResult.append("6");
 }
 if(v.equals(btnSeven))
{
 txtResult.append("7");
 }
 if(v.equals(btnEight))
 {
```

```
 txtResult.append("8");
 }
 if(v.equals(btnNine))
 {
 txtResult.append("9");
 }
 if(v.equals(btnZero))
 {
 txtResult.append("0");
 }
 if(v.equals(btnClear))
 {
 txtResult.setText("");
 }
 if(v.equals(btnEqual))
 {
 try {
 String data = txtResult.getText().toString();  if
(data.contains("/")) {
         String[] operands = data.split("/");
              if(operands.length==2) {
 double operand1 =
Double.parseDouble(operands[0]);
 double operand2 =
Double.parseDouble(operands[1]);
 double result = operand1 / operand2;
 txtResult.setText(String.valueOf(result));  }
                    else
                     {
 Toast.makeText(getBaseContext(),  "Invalid
Input",
 Toast.LENGTH_LONG).show();  }

 } else if (data.contains("*")) {
 String[] operands =
data.split(Pattern.quote("*"));
 if(operands.length==2) {
 double operand1 =
Double.parseDouble(operands[0]);
 double operand2 =
Double.parseDouble(operands[1]);
 double result= operand1 * operand2;
txtResult.setText(String.valueOf(result));



}
                    else
                     {
 Toast.makeText(getBaseContext(),  "Invalid
Input",
 Toast.LENGTH_LONG).show();  }
```

```
 } else if (data.contains("+")) {
 String[] operands =
data.split(Pattern.quote("+"));
 if(operands.length==2) {
 double operand1 =
Double.parseDouble(operands[0]);
 double operand2 =
Double.parseDouble(operands[1]);
    double result= operand1 + operand2;
txtResult.setText(String.valueOf(result));
 }
                    else
                     {
 Toast.makeText(getBaseContext(),  "Invalid
Input",
                            Toast.LENGTH_LONG).show();
 }

 } else if (data.contains("-")) {
 String[] operands = data.split("-");
 if(operands.length==2) {
 double operand1 =
Double.parseDouble(operands[0]);
 double operand2 =
Double.parseDouble(operands[1]);
 double result = operand1 - operand2;
 txtResult.setText(String.valueOf(result));  }
                    else
                     {
 Toast.makeText(getBaseContext(),  "Invalid
Input",
                            Toast.LENGTH_LONG).show();
 }


 }
 }
 catch(Exception e)
 {
 Toast.makeText(getBaseContext(),
 "Invalid Input",
                        Toast.LENGTH_LONG).show();
 }

}
 if(v.equals(btnAdd))
 {
 txtResult.append("+");
 }
 if(v.equals(btnSub))
 {
 txtResult.append("-");
```

```
 }
 if(v.equals(btnMul))
 {
 txtResult.append("*");
 }
 if(v.equals(btnDiv))
 {
 txtResult.append("/");
 }
 }
}
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
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">

    <Button
        android:id="@+id/button_clear"
        android:layout_width="87dp"
        android:layout_height="53dp"
        android:layout_marginTop="30dp"
        android:text="C"
        app:layout_constraintStart_toStartOf="@+id/button_add"
        app:layout_constraintTop_toBottomOf="@+id/button_add"
        />

    <Button
        android:id="@+id/button_sub"
        android:layout_width="87dp"
        android:layout_height="53dp"
        android:layout_marginStart="20dp
        "
        android:layout_marginTop="30d
        p" android:text="-"
        app:layout_constraintStart_toEndOf="@+id/button_three"
        app:layout_constraintTop_toBottomOf="@+id/button_mul"
        />

    <Button
```

```
        android:id="@+id/button_add"
        android:layout_width="87dp"
        android:layout_height="53dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
        android:text="+"
        app:layout_constraintStart_toEndOf="@+id/button_equal"
        app:layout_constraintTop_toBottomOf="@+id/button_sub"
        />

    <Button
        android:id="@+id/button_mul"
        android:layout_width="87dp"
        android:layout_height="53dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
        android:text="*"
        app:layout_constraintStart_toEndOf="@+id/button_six"
        app:layout_constraintTop_toBottomOf="@+id/button_div" />

    <Button
        android:id="@+id/button_equal"
        android:layout_width="62dp"
        android:layout_height="53dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
        android:text="="
        app:layout_constraintStart_toEndOf="@+id/button_zero"
        app:layout_constraintTop_toBottomOf="@+id/button_three" />

    <Button
        android:id="@+id/button_zero"
        android:layout_width="62dp"
        android:layout_height="53dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
        android:text="0"
        app:layout_constraintStart_toEndOf="@+id/button_dot"
        app:layout_constraintTop_toBottomOf="@+id/button_two"
        />

    <Button
        android:id="@+id/button_dot"
        android:layout_width="62dp"
        android:layout_height="53dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
```

```
        android:text="."
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/button_one" />

    <Button
        android:id="@+id/button_three"
        android:layout_width="62dp"
        android:layout_height="53dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
        android:text="3"
        app:layout_constraintStart_toEndOf="@+id/button_two"
        app:layout_constraintTop_toBottomOf="@+id/button_six" />

    <Button
        android:id="@+id/button_two"
        android:layout_width="62dp"
        android:layout_height="53dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
        android:text="2"
        app:layout_constraintStart_toEndOf="@+id/button_one"
        app:layout_constraintTop_toBottomOf="@+id/button_five" />

    <Button
        android:id="@+id/button_one"
        android:layout_width="62dp"
        android:layout_height="53dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
        android:text="1"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/button_four" />

    <Button
        android:id="@+id/button_six"
        android:layout_width="62dp"
        android:layout_height="53dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
        android:text="6"
        app:layout_constraintStart_toEndOf="@+id/button_five"
        app:layout_constraintTop_toBottomOf="@+id/button_nine" />

    <Button
        android:id="@+id/button_seven"
        android:layout_width="62dp"
```

```
            android:layout_height="53dp"
            android:layout_marginStart="20dp"
            android:layout_marginTop="20dp"
            android:text="7"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/txt_result" />

        <Button
            android:id="@+id/button_eight"
            android:layout_width="62dp"
            android:layout_height="53dp"
            android:layout_marginStart="20dp"
            android:layout_marginTop="20dp"
            android:text="8"
            app:layout_constraintStart_toEndOf="@+id/button_seven"
            app:layout_constraintTop_toBottomOf="@+id/txt_result" />

        <Button
            android:id="@+id/button_nine"
            android:layout_width="62dp"
            android:layout_height="53dp"
            android:layout_marginStart="20dp"
            android:layout_marginTop="20dp"
            android:text="9"
            app:layout_constraintStart_toEndOf="@+id/button_eight"
            app:layout_constraintTop_toBottomOf="@+id/txt_result" />

        <Button
            android:id="@+id/button_four"
            android:layout_width="62dp"
            android:layout_height="53dp"
            android:layout_marginStart="20dp"
            android:layout_marginTop="30dp"
            android:text="4"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/button_seven" />


<TextView
            android:id="@+id/textView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="30dp"
            android:text="SIMPLE                CALCULATOR"
            android:textSize="26dp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent" />
```

```
    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content
        "
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="20dp"
        android:text="Result"
        android:textSize="18dp"
        android:textStyle="bold"
        app:layout_constraintEnd_toStartOf="@+id/textView"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView"
        />

    <EditText
        android:id="@+id/txt_result"
        android:layout_width="310dp"
        android:layout_height="46dp"
        android:layout_marginTop="20dp"
        android:ems="10"
        android:inputType="textPersonName"
        app:layout_constraintStart_toStartOf="@+id/textView2"
        app:layout_constraintTop_toBottomOf="@+id/textView2"
        />

    <Button
        android:id="@+id/button_div"
        android:layout_width="87dp"
        android:layout_height="53dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="20dp"
        android:text="/"
        app:layout_constraintStart_toEndOf="@+id/button_nine"
        app:layout_constraintTop_toBottomOf="@+id/txt_result" />

    <Button
        android:id="@+id/button_five"
        android:layout_width="62dp"
        android:layout_height="53dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
        android:text="5"
        app:layout_constraintStart_toEndOf="@+id/button_four"
        app:layout_constraintTop_toBottomOf="@+id/button_eight" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

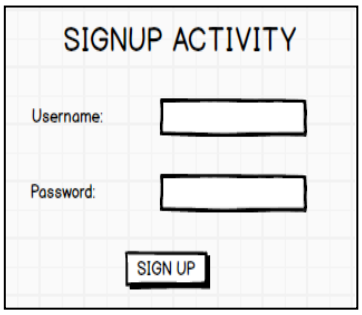
SAMPLE OUTPUT

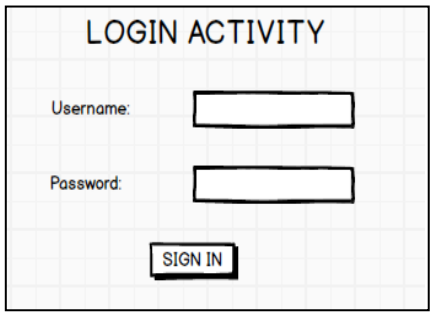
Program 3

Create a SIGN Up activity with Username and Password. Validation of password should happen based on the following rules:

- Password should contain uppercase and lowercase letters.
- Password should contain letters and numbers.
- Password should contain special characters.
- Minimum length of the password (the default value is 8).

On successful **SIGN UP** proceed to the next Login activity. Here the user should **SIGN IN** using the Username and Password created during signup activity. If the Username and Password are matched then navigate to the next activity which displays a message saying "Successful Login" or else display a toast message saying "Login Failed". The user is given only two attempts and after that display a toast message saying "Failed Login Attempts" and disable the SIGN IN button. Use Bundle to transfer information from one activity to another.





MainActivity.java

```
package com.example.parta.program3;

import androidx.appcompat.app.AppCompatActivity;

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

import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class MainActivity extends AppCompatActivity implements
 View.OnClickListener {

 EditText txtUsername;
 EditText txtPassword;
```

```
 Button btnSignup;

 String regularExpression
 ="^(?=.*[A-Z])(?=.*[a-z])(?=.*\\d)(?=.*[@$!])[A-Za
z\\d@$!]{8,}$";

 @Override
 protected void onCreate(Bundle savedInstanceState) {
 super.onCreate(savedInstanceState);
 setContentView(R.layout.activity_main);
 txtUsername=(EditText)findViewById(R.id.txt_username);
 txtPassword=(EditText)findViewById(R.id.txt_password);

 btnSignup=(Button)findViewById(R.id.btn_signup);
 btnSignup.setOnClickListener(this);

 }

 public void onClick(View v)
 {
 String username=txtUsername.getText().toString();  String
password=txtPassword.getText().toString();

 if(validatePassword(password)) {
 Bundle bundle = new Bundle();
 bundle.putString("user", username);
 bundle.putString("pass", password);

 Intent it = new Intent(this, LoginActivity.class);
 it.putExtra("data", bundle);

 startActivity(it);
 }
 else
 {
 Toast.makeText(getBaseContext(),
 "Invalid Password",
 Toast.LENGTH_LONG).show();
 }

}

 public boolean validatePassword(String password)  {
 Pattern pattern= Pattern.compile(regularExpression);
 Matcher matcher=pattern.matcher(password);  return
matcher.matches();

 }
}
```

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
 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">

 <TextView
 android:id="@+id/textView2"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginTop="100dp"
 android:text="SIGN UP"
 android:textColor="@android:color/background_dark"
 android:textSize="22dp"
 app:layout_constraintEnd_toEndOf="parent"
 app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintTop_toTopOf="parent" />

 <TextView
 android:id="@+id/textView3"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginStart="30dp"
 android:layout_marginTop="50dp"
 android:text="USERNAME"
 app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintTop_toBottomOf="@+id/textView2" />

 <TextView
 android:id="@+id/textView4"
 android:layout_width="68dp"
 android:layout_height="0dp"
 android:layout_marginTop="50dp"
 android:text="PASSWORD"
 app:layout_constraintStart_toStartOf="@+id/textView3"
app:layout_constraintTop_toBottomOf="@+id/textView3" />

 <EditText
 android:id="@+id/txt_username"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginStart="40dp"
 android:layout_marginEnd="10dp"
 android:ems="10"
 android:inputType="textPersonName"
 app:layout_constraintBottom_toBottomOf="@+id/textView3"
 app:layout_constraintEnd_toEndOf="parent"
 app:layout_constraintStart_toEndOf="@+id/textView3"
 app:layout_constraintTop_toTopOf="@+id/textView3" />
```

```
 <EditText
 android:id="@+id/txt_password"
 android:layout_width="0dp"
 android:layout_height="40dp"
 android:layout_marginTop="26dp"
 android:ems="10"
 android:inputType="textPassword"
 app:layout_constraintEnd_toEndOf="@+id/txt_username"
 app:layout_constraintStart_toStartOf="@+id/txt_username"
 app:layout_constraintTop_toBottomOf="@+id/txt_username" />

 <Button
 android:id="@+id/btn_signup"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginTop="30dp"
 android:text="Sign Up"
 app:layout_constraintEnd_toEndOf="parent"
 app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintTop_toBottomOf="@+id/txt_password" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

LoginActivity.java:

```
package com.example.parta.program3;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class LoginActivity extends AppCompatActivity implements
 View.OnClickListener {

EditText txtLoginUsername;
 EditText txtLoginPassword;
 Button btnLogin;

 String user,pass;

 int count=0;

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

 txtLoginUsername=(EditText)
 findViewById(R.id.txt_login_username);
 txtLoginPassword=(EditText)
```

```
        findViewById(R.id.txt_login_password);

        btnLogin=(Button)findViewById(R.id.btn_login_signin);
        btnLogin.setOnClickListener(this);

        Bundle bundle=getIntent().getBundleExtra("data");
        user=bundle.getString("user");
        pass=bundle.getString("pass");



        }

        public void onClick(View v)
        {
        String user1=txtLoginUsername.getText().toString();
        String pass1=txtLoginPassword.getText().toString();

        if(user.equals(user1)&&pass.equals(pass1))  {
        Toast.makeText(this,"Login Successful"
        ,Toast.LENGTH_LONG).show();  }
        else
        {
        count++;
        if(count==3)
        {
        btnLogin.setEnabled(false);
        Toast.makeText(this,  "Failed Login Attempts"
,Toast.LENGTH_LONG).show();
        }
        else
        {
        Toast.makeText(this,"Login Failed "+count
        ,Toast.LENGTH_LONG).show();  }
        }

       }


        }
```

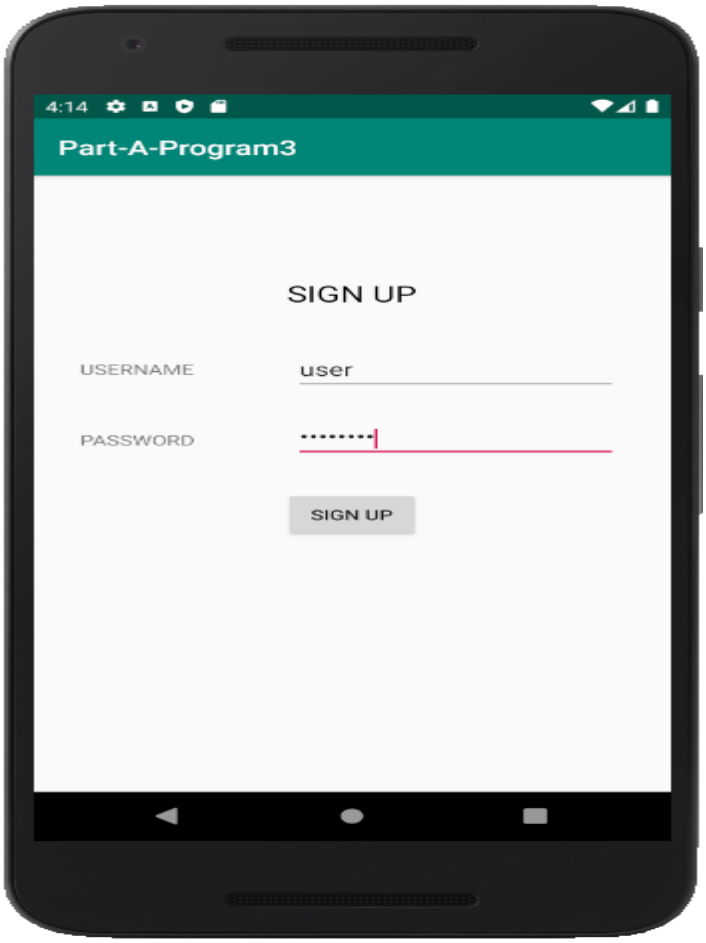
activity_login.xml:

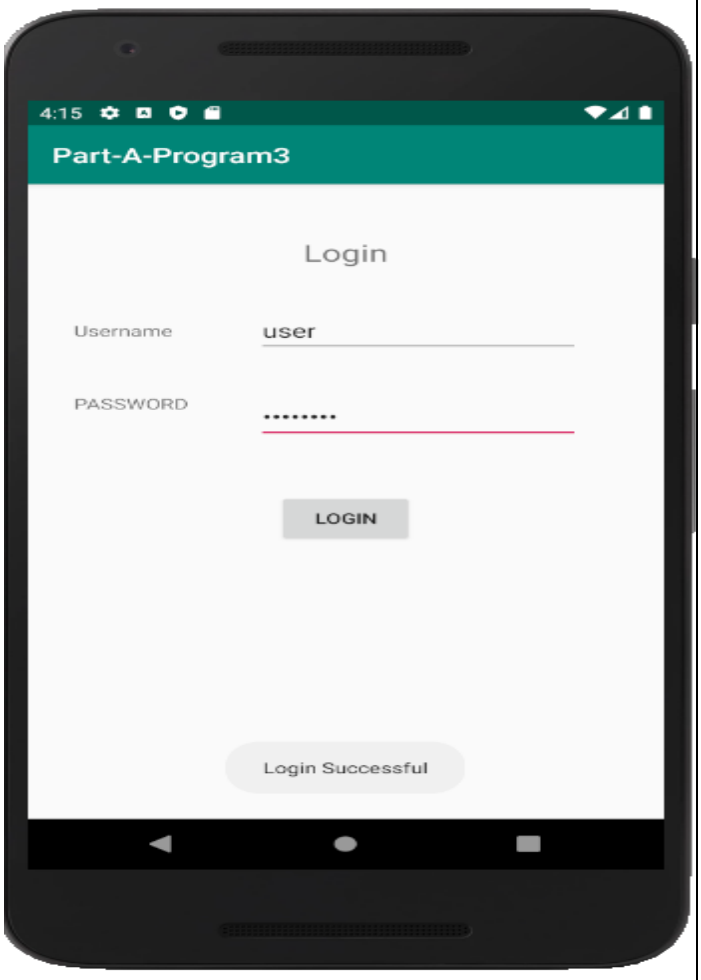
```
        <?xml version="1.0" encoding="utf-8"?>
        <androidx.constraintlayout.widget.ConstraintLayout
         xmlns:android="http://schemas.android.com/apk/res/android"
         xmlns:app="http://schemas.android.com/apk/res-auto"
         xmlns:tools="http://schemas.android.com/tools"
         android:layout_width="match_parent"
         android:layout_height="match_parent"
         tools:context=".LoginActivity">
```

```
 <TextView
 android:id="@+id/textView7"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginTop="50dp"
 android:text="Login"
 android:textSize="22dp"
  app:layout_constraintEnd_toEndOf="parent"
 app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintTop_toTopOf="parent" />

 <TextView
 android:id="@+id/textView9"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginStart="30dp"
 android:layout_marginTop="50dp"
 android:text="Username"
 app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintTop_toBottomOf="@+id/textView7" />

 <EditText
 android:id="@+id/txt_login_username"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginStart="20dp"
 android:layout_marginEnd="20dp"
 android:ems="10"
 android:inputType="textPersonName"
 app:layout_constraintBottom_toBottomOf="@+id/textView9"
 app:layout_constraintEnd_toEndOf="parent"
 app:layout_constraintStart_toEndOf="@+id/textView9"
 app:layout_constraintTop_toTopOf="@+id/textView9" />

<TextView
 android:id="@+id/textView10"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginStart="30dp"
 android:layout_marginTop="50dp"
 android:text="PASSWORD"
 app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintTop_toBottomOf="@+id/textView9" />

 <EditText
 android:id="@+id/txt_login_password"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:ems="10"
 android:inputType="textPassword"
 app:layout_constraintEnd_toEndOf="@+id/txt_login_username"
 app:layout_constraintStart_toStartOf="@+id/txt_login_username"
 app:layout_constraintTop_toTopOf="@+id/textView10" />
```

```
 <Button
 android:id="@+id/btn_login_signin"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginTop="50dp"
 android:text="Login"
 app:layout_constraintEnd_toEndOf="parent"
 app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintTop_toBottomOf="@+id/txt_login_password"                    />
</androidx.constraintlayout.widget.ConstraintLayout>
```

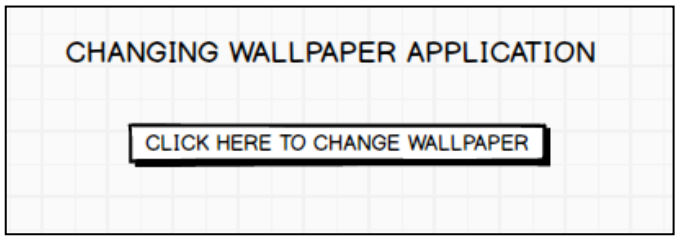
Sample Output





Program 4

Develop an application to set an image as wallpaper. On click of a button, the wallpaper image should start to change randomly every 30 seconds.



MainActivity.java:

```
package com.example.program4;

import androidx.appcompat.app.AppCompatActivity;

import android.app.WallpaperManager;
import android.graphics.BitmapFactory;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

import java.util.Timer;
import java.util.TimerTask;

public class MainActivity extends AppCompatActivity implements
 View.OnClickListener {

 Button btnChangeWallpaper;

 boolean running;

 int[] imagesArray=new int[]{R.drawable.image1,
 R.drawable.image2,
 R.drawable.image3,R.drawable.image4};

 int i=0;

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

 btnChangeWallpaper=(Button)
 findViewById(R.id.btn_start_change_wallpaper);
 btnChangeWallpaper.setOnClickListener(this);

 }
```

```
 public void onClick(View v)
 {
 if(!running)
 {
 new Timer().schedule
 (new MyTimer(),0,30000);
 running=true;
 }

 }

 class MyTimer extends TimerTask
 {
 public void run()
 {

 try {
 WallpaperManager wallpaperManager =
 WallpaperManager.getInstance(getBaseContext());  if(i==4)
 {
 i=1;
 }
 if(i==2)
 {
 i=3;
 }
if(i==3)
 {
 i=2;
 }
 if(i==1)
 {
 i=3;
 }

 wallpaperManager.setBitmap
 (BitmapFactory.decodeResource(getResources()  ,imagesArray[i]));
 i++;
 }
 catch(Exception e)
 {

 }

 }
 }

}
```

activity main.xml:

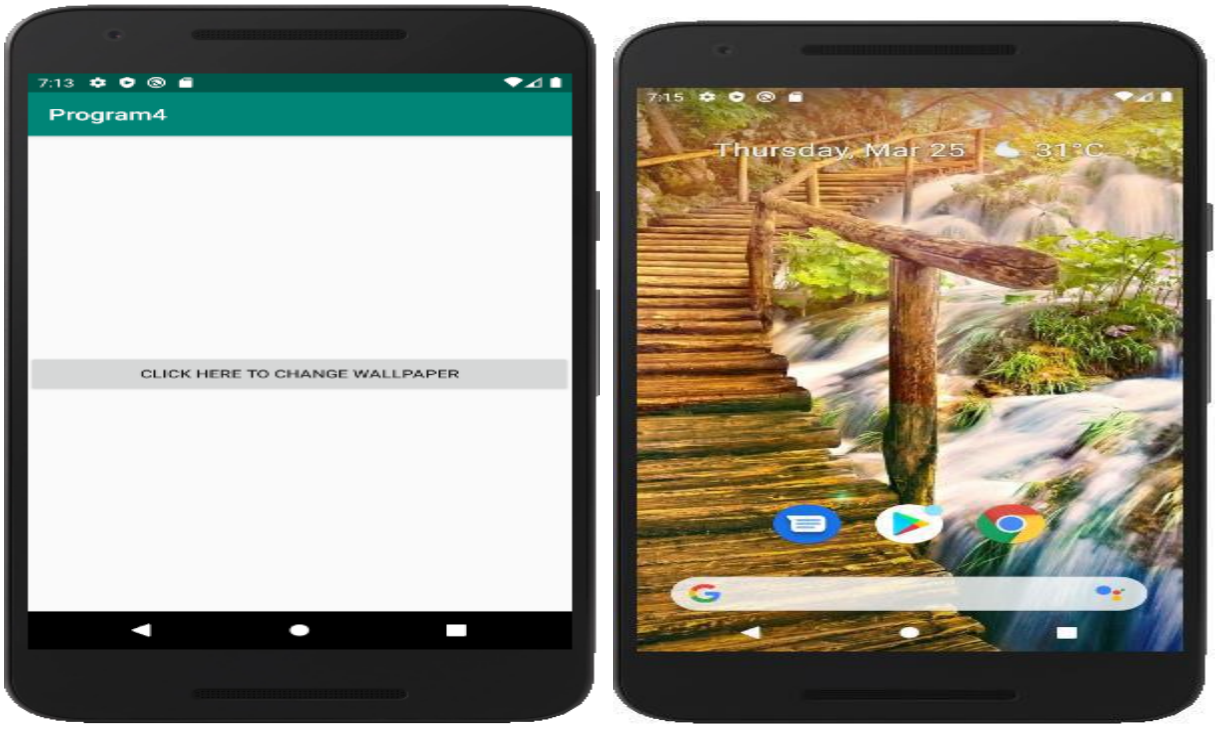
```
<?xml version="1.0" encoding="utf-8"?>
```

```
<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:orientation="vertical"
 android:gravity="center"
 tools:context=".MainActivity">

 <Button
 android:layout_width="match_parent"
 android:layout_height="wrap_content"
 android:text="Click here to Change Wallpaper"
 android:id="@+id/btn_start_change_wallpaper"/>



</LinearLayout>
```

AndroidManifest.xml:

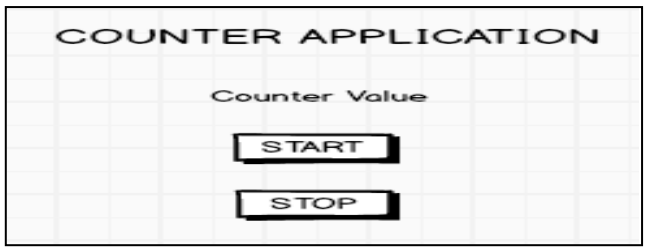
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
 package="com.example.program4">

 <uses-permission android:name="android.permission.SET_WALLPAPER"/>

 <application
android:allowBackup="true"
 android:icon="@mipmap/ic_launcher"
 android:label="@string/app_name"
 android:roundIcon="@mipmap/ic_launcher_round"
 android:supportsRtl="true"
 android:theme="@style/AppTheme">
 <activity android:name=".MainActivity">
 <intent-filter>
 <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

Sample Output



Program 5

Write a program to create an activity with two buttons START and STOP. On Pressing of the START button, the activity must start the counter by displaying the numbers from One and the counter must keep on counting until the STOP button is pressed. Display the counter value in a TextViewcontrol.



MainActivity.java:

```
package com.example.program5;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.os.Handler;
import android.os.Message;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;



public class MainActivity extends AppCompatActivity implements
 View.OnClickListener {

 TextView lblCounter;
 Button btnStart,btnStop;

 int counter=0;
 boolean running=false;

 @Override
 protected void onCreate(Bundle savedInstanceState) {
 super.onCreate(savedInstanceState);
 setContentView(R.layout.activity_main);
 lblCounter=(TextView)findViewById(R.id.lbl_text);
 btnStart=(Button)findViewById(R.id.btn_start);
 btnStop=(Button)findViewById(R.id.btn_stop);
 btnStop.setOnClickListener(this);
 btnStart.setOnClickListener(this);
 }

 public void onClick(View v)
 {
 if(v.equals(btnStart))
```

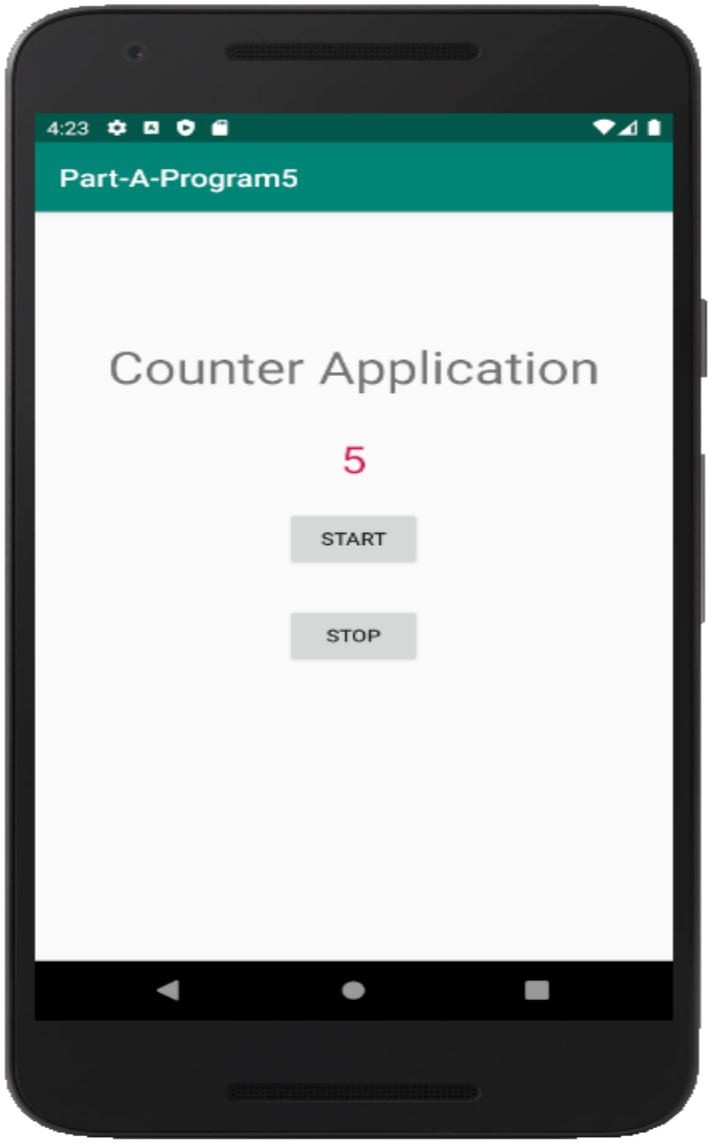
```
 {
 counter=0;
 running=true;
 new MyCounter().start();
 }
 else if(v.equals(btnStop))
 {
 running=false;
 }
 }

 Handler handler=new Handler()
 {
 public void handleMessage(Message m)  {
 lblCounter.setText(String.valueOf(m.what));  }

 };

 class MyCounter extends Thread
 {
 public void run()
 {
 while(running)
{
 counter++;
 handler.sendEmptyMessage(counter);  try {
 Thread.sleep(1000);
 }
 catch(Exception e)
 {

 }
 }

 }
 }
}
```

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
 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/lbl_counter"
 android:layout_width="match_parent"
 android:layout_height="match_parent"
 tools:context=".MainActivity">

 <TextView
```

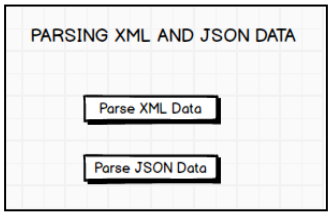
```
 android:id="@+id/textView"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginTop="100dp"
 android:text="Counter Application"
 android:textSize="30sp"
 app:layout_constraintEnd_toEndOf="parent"
 app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintTop_toTopOf="parent" />

 <TextView
 android:id="@+id/lbl_text"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginTop="30dp"
 android:text="Counter Value"
 android:textColor="@color/colorAccent"
 android:textSize="50dp"
 app:layout_constraintEnd_toEndOf="parent"
 app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintTop_toBottomOf="@+id/textView" />

 <Button
android:id="@+id/btn_start"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginTop="20dp"
 android:text="Start"
 app:layout_constraintEnd_toEndOf="parent"
 app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintTop_toBottomOf="@+id/lbl_text" />

 <Button
 android:id="@+id/btn_stop"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_marginTop="30dp"
 android:text="Stop"
 app:layout_constraintEnd_toEndOf="parent"
 app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintTop_toBottomOf="@+id/btn_start"                    />
</androidx.constraintlayout.widget.ConstraintLayout>
```

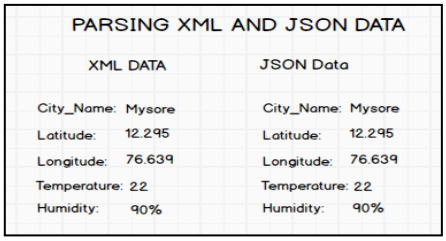
Sample Output



Program 6

Create two files of XML and JSON type with values for City_Name, Latitude, Longitude, Temperature, and Humidity. Develop an application to create an activity with two buttons to parse the XML and JSON files which when clicked should display the data in their respective layouts side by side.





1. Create a New Android Project with Empty Activity.
2. Open activity_main.xml file from res ⟶ layout folder, check/add ConstraintLayout as the root view.
3. Create the layout design using Drag and Drop framework.
4. Add Listeners to Button Click Event:
   - Create a class which implements OnClickListener interface.
   - Override onClick() method of OnClickListener Interface.
   - Register the button for click event by calling setOnClickListener() method of View class and pass the object of the class that implemented OnClickListener Interface.
5. Create assets folder (Refer Section Android Studio Tutorial)
6. Create a input.xml file inside assets folder and paste the below Xml Data

```
<?xml version="1.0"?>
<records>
    <employee>
        <city_name>Mysore</city_name>
        <Latitude>12.295</Latitude>
        <Longitude>76.639</Longitude>
        <Temperature>22</Temperature>
        <Humidity>90%</Humidity>
    </employee>
</records>
```

7. Create a input.json file inside assets folder and paste the below Json Data

```
{

   "employee":          {
   "city_name":
   "Mysore",
   "Latitude": "12.295",
   "Longitude": "76.639",
   "Temperature": 22,

   "Humidity": "90%"
}

}
```

8. Read the XML and Json Data in the files and display on screen

MainActivity.java

```
package com.example.parta_program6;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity implements
View.OnClickListener {
    Button
    btnParseXml,btnParseJson;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btnParseXml=(Button)findViewById(R.id.btn_parsexml);
        btnParseJson=(Button)findViewById(R.id.btn_parsejson);
        btnParseJson.setOnClickListener(this);
        btnParseXml.setOnClickListener(this);
    }

    @Override
    public void onClick(View v) {

        if(v.equals(btnParseJson))
        {
```

```
            Intent it=new Intent(this,ViewActivity.class);
            it.putExtra("mode",1);
            startActivity(it);


        }
        else if(v.equals(btnParseXml))
        {
            Intent it=new Intent(this,ViewActivity.class);
            it.putExtra("mode",2);
            startActivity(it);
        }

    }
}
```

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
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">

    <Button
        android:id="@+id/btn_parsexml"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="80dp"
        android:text="Parse XML Data"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView4" />

    <Button
        android:id="@+id/btn_parsejson"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="80dp"
        android:text="Parse Json Data"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
```

```
        app:layout_constraintTop_toBottomOf="@+id/btn_parsexml" />

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="40dp"
        android:text="PARSING XML AND JSON
        DATA"                android:textSize="20dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
          app:layout_constraintTop_toTopOf="parent" />
      </androidx.constraintlayout.widget.ConstraintLayout>
```

activity_view.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
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=".ViewActivity">

    <TextView
        android:id="@+id/lbl_xml_data"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="Xml Data"
        app:layout_constraintStart_toStartOf="@+id/textView2"
        app:layout_constraintTop_toBottomOf="@+id/textView2" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="40dp"
        android:text="PARSING XML AND JSON DATA"
        android:textSize="20dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
```

```
    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="40dp"
        android:layout_marginTop="20dp"
        android:text="XML                    DATA"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:layout_marginEnd="40dp"
        android:text="JSON              DATA"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView" />

    <TextView
        android:id="@+id/lbl_json_data"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="Json                          Data"
        app:layout_constraintEnd_toEndOf="@+id/textView3"
        app:layout_constraintTop_toBottomOf="@+id/textView3" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

ViewActivity.java:

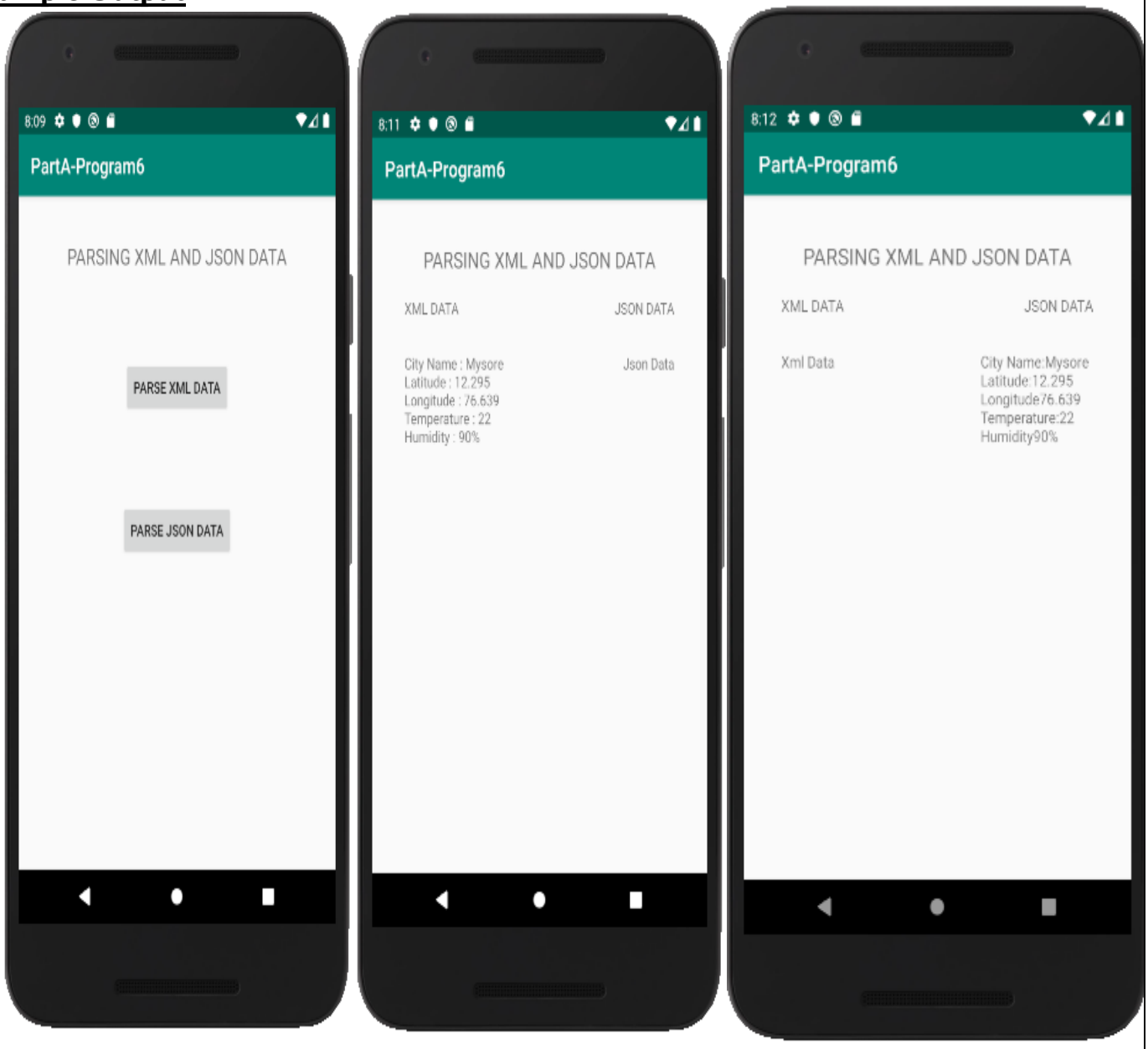
```
package com.example.parta_program6;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.TextView;

import
org.json.JSONObject;
import
org.w3c.dom.Document;
```

```
import
org.w3c.dom.Element;
import org.w3c.dom.Node;
import
org.w3c.dom.NodeList;

import java.io.InputStream;

import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;

public class ViewActivity extends AppCompatActivity {

    TextView lblXmlData,lblJsonData;

    int
    mode
    0;
    @Overi
    de
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_view);
        lblXmlData=(TextView)findViewById(R.id.lbl_xml_data);
        lblJsonData=(TextView)findViewById(R.id.lbl_json_data);
        mode=getIntent().getIntExtra("mode",0);

        if(mode==1)
        {
            parseJson();
        }
        else
        {
            parseXmlDocument();
        }
    }

    public String parseXmlDocument()
    {
        try {
```

```
                InputStream is = getAssets().open("input.xml");

                DocumentBuilderFactory dbFactory =
 DocumentBuilderFactory.newInstance();
                DocumentBuilder dBuilder = dbFactory.newDocumentBuilder();
                Document doc = dBuilder.parse(is);

                Element
                element=doc.getDocumentElement();
                element.normalize();

                NodeList nList = doc.getElementsByTagName("employee");

                for (int i=0; i<nList.getLength(); i++) {

                     Node node = nList.item(i);
                     if (node.getNodeType() ==
                          Node.ELEMENT_NODE) { Element
                          element2 = (Element) node;
                          lblXmlData.setText("City Name : " + getValue("city_name",
 element2)+"\n");

 element2)+"\n");

 element2)+"\n");
catch (Exception e) {

e.printStackTrace();

}

          return      null;
     }


private static String getValue(String tag, Element element) { NodeList
nodeList =
 element.getElementsByTagName(tag).item(0).getChildNodes(); Node
          node = nodeList.item(0);
          return node.getNodeValue();
     }

     public void parseJson()
```

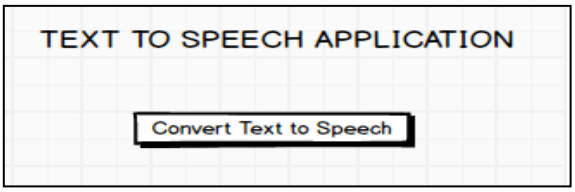
```
    {
        try {
            InputStream inputStream=getAssets().open("input.json");
            byte[] data=new byte[inputStream.available()];
            inputStream.read(data);

            String readData=new String(data);
            JSONObject jsonObject=new JSONObject(readData);
            JSONObject jsonObject1=jsonObject.getJSONObject("employee");
            lblJsonData.setText("City
Name:"+jsonObject1.getString("city_name")+"\n");

lblJsonData.append("Latitude:"+jsonObject1.getString("Latitude")+"\n");
lblJsonData.append("Longitude"+jsonObject1.getString("Longitude")+"\n");

lblJsonData.append("Temperature:"+jsonObject1.getInt("Temperature")+"\n");
lblJsonData.append("Humidity"+jsonObject1.getString("Humidity")+"\n");
        }

Catch (Exception e)

{
        e.printStackTrace();


}
}
}
```

Sample Output



Program 7

Develop a simple application with one EditText so that the user can write some text in it. Create a button called "Convert Text to Speech" that converts the user input text into voice.



MainActivity.java

```
package com.example.parta.parta_program7;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.speech.tts.TextToSpeech;
import android.view.View;
 import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import java.util.Locale;

public class MainActivity extends AppCompatActivity implements
View.OnClickListener
{

    EditText txtSpeak;
    Button btnSpeak;
    TextToSpeech textToSpeech;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        txtSpeak=(EditText)findViewById(R.id.editText);
        btnSpeak=(Button)findViewById(R.id.btn_speak);
        btnSpeak.setOnClickListener(this); textToSpeech=new
        TextToSpeech(getBaseContext(),
                new TextToSpeech.OnInitListener() {
            @Override
```

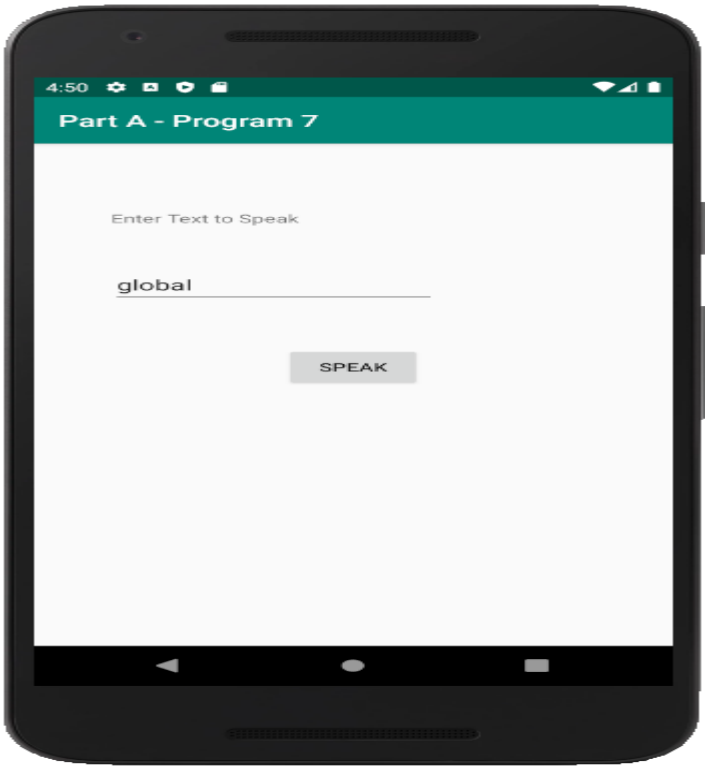
```
            public void onInit(int status) {
                if(status!=TextToSpeech.ERROR)
                {
                    Toast.makeText(getBaseContext(), "Success",
                            Toast.LENGTH_LONG).show();
                }
            }
        });
        textToSpeech.setLanguage(Locale.UK);
    }

    public void onClick(View v)
    {
        String text=txtSpeak.getText().toString();
        textToSpeech.speak(text,

                TextToSpeech.QUEUE_FLUSH,null);

    }
}
```

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
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/txt_texttospeak"    android:layout_width="match_parent"
    android:layout_height="match_parent" tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="50dp"
        android:layout_marginTop="80dp" android:text="Enter
        Text                    to                    Speak"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/editText"
```

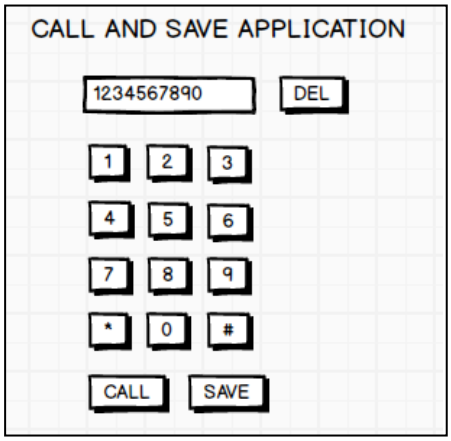
```
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="48dp"  android:ems="10"
            android:inputType="textPersonName"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.0"
            app:layout_constraintStart_toStartOf="@+id/textView"
            app:layout_constraintTop_toBottomOf="@+id/textView" />

        <Button
            android:id="@+id/btn_speak"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="52dp"
            android:text="Speak"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/editText" />
 </androidx.constraintlayout.widget.ConstraintLayout>
```

Sample Output



Program 8

Create an activity like a phone dialer with CALL and SAVE buttons. On pressing the CALL button, it must call the phone number and on pressing the SAVE button it must save the number to the phone contacts.



MainActivity.java:

```
package com.example.part_a_program_8;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.provider.ContactsContract;
import android.view.View; import
android.widget.Button; import
android.widget.EditText;

public class MainActivity extends AppCompatActivity implements
View.OnClickListener {

    Button btnOne,btnTwo,btnThree,btnFour,btnFive;
    Button btnSix,btnSeven,btnEight,btnNine,btnZero;
    Button btnDel,btnStar,btnHash,btnCall,btnSave;
    EditText txtPhonenumber;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
```

```
btnOne=(Button)findViewById(R.id.btn_one);
btnOne.setOnClickListener(this);

btnTwo=(Button)findViewById(R.id.btn_two);
btnTwo.setOnClickListener(this);

btnThree=(Button)findViewById(R.id.btn_three); btnThree.setOnClickListener(this);


btnFour=(Button)findViewById(R.id.btn_four);
btnFour.setOnClickListener(this);

btnFive=(Button)findViewById(R.id.btn_five);
btnFive.setOnClickListener(this);

btnSix=(Button)findViewById(R.id.btn_six); btnSix.setOnClickListener(this);

btnSeven=(Button)findViewById(R.id.btn_seven); btnSeven.setOnClickListener(this);

btnEight=(Button)findViewById(R.id.btn_eight); btnEight.setOnClickListener(this);

btnNine=(Button)findViewById(R.id.btn_nine);
btnNine.setOnClickListener(this);

btnZero=(Button)findViewById(R.id.btn_zero);
btnZero.setOnClickListener(this);

btnStar=(Button)findViewById(R.id.btn_start);
btnStar.setOnClickListener(this);

btnHash=(Button)findViewById(R.id.btn_hash);
btnHash.setOnClickListener(this);

btnCall=(Button)findViewById(R.id.btn_call);
btnCall.setOnClickListener(this);

btnSave=(Button)findViewById(R.id.btn_save);
btnSave.setOnClickListener(this);

btnDel=(Button)findViewById(R.id.btn_delete);
btnDel.setOnClickListener(this);

txtPhonenumber=(EditText)findViewById(R.id.txt_phonenumber);
txtPhonenumber.setText("");
```

```
}

public void onClick(View v)
{
    if(v.equals(btnOne))
    {
        txtPhonenumber.append("1");
    }
    else    if(v.equals(btnTwo))
    {
        txtPhonenumber.append("2");
    }
    else    if(v.equals(btnThree))
    {
        txtPhonenumber.append("3");
    }
    else    if(v.equals(btnFour))
    {
        txtPhonenumber.append("4");
    }
    else    if(v.equals(btnFive))
    {
        txtPhonenumber.append("5");
    }
    else    if(v.equals(btnSix))
    {
        txtPhonenumber.append("6");
    }
    else    if(v.equals(btnSeven))
    {
        txtPhonenumber.append("7");
    }
    else    if(v.equals(btnEight))
    {
        txtPhonenumber.append("8");
    }
    else    if(v.equals(btnNine))
    {
        txtPhonenumber.append("9");
    }
    else    if(v.equals(btnZero))
    {
        txtPhonenumber.append("0");
    }
```

```
else if(v.equals(btnStar))
{
    txtPhonenumber.append("*");
}
else    if(v.equals(btnHash))
{
    txtPhonenumber.append("#");
}
else    if(v.equals(btnSave))
{

    Intent contactIntent = new Intent
            (ContactsContract.Intents.Insert.ACTION);
    contactIntent.setType
            (ContactsContract.RawContacts.CONTENT_TYPE);

    contactIntent
            .putExtra(ContactsContract.Intents.Insert.NAME,
                    "Unknown");
    contactIntent.putExtra(ContactsContract.Intents.Insert.PHONE,
            txtPhonenumber.getText().toString());
    startActivity(contactIntent);
}
else    if(v.equals(btnCall))
{
    String data=txtPhonenumber.getText().toString(); Intent
    it=new Intent(Intent.ACTION_CALL);
    it.setData(Uri.parse("tel:"+data)); startActivity(it);

}
else    if(v.equals(btnDel))
{
    String data=txtPhonenumber.getText().toString();
    if(data.length()>0)
    {
        txtPhonenumber.setText
                (data.substring(0,data.length()-1));

    }
    else
    {
        txtPhonenumber.setText("");
    }
```

```
            }

        }
}
```

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
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">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="50dp"
        android:text="PHONE                    DAILER"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/txt_phonenumber"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"  android:ems="10"
        android:inputType="textPersonName"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView" />

    <Button
        android:id="@+id/btn_delete"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
        android:text="Delete"
        app:layout_constraintStart_toEndOf="@+id/txt_phonenumber"
        app:layout_constraintTop_toBottomOf="@+id/textView" />
```

```
<Button
    android:id="@+id/btn_one"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="20dp"
    android:layout_marginTop="30dp"    android:text="1"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/txt_phonenumber" />

<Button
    android:id="@+id/btn_two"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="30dp"
    android:text="2"
    app:layout_constraintEnd_toStartOf="@+id/btn_three"
    app:layout_constraintStart_toEndOf="@+id/btn_one"
    app:layout_constraintTop_toBottomOf="@+id/txt_phonenumber" />

<Button
    android:id="@+id/btn_three"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="30dp"
    android:layout_marginEnd="20dp"
    android:text="3"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/txt_phonenumber" />

<Button
    android:id="@+id/btn_four"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="20dp"
    android:layout_marginTop="30dp"    android:text="4"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/btn_one" />

<Button
    android:id="@+id/btn_five"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="30dp"
    android:text="5"
```

```
        app:layout_constraintEnd_toStartOf="@+id/btn_six"
        app:layout_constraintStart_toEndOf="@+id/btn_four"
        app:layout_constraintTop_toBottomOf="@+id/btn_two" />

    <Button
        android:id="@+id/btn_six"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:layout_marginEnd="20dp"
        android:text="6"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/btn_three" />

    <Button
        android:id="@+id/btn_seven"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"    android:text="7"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/btn_four" />
    <Button
        android:id="@+id/btn_eight"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="8"
        app:layout_constraintEnd_toStartOf="@+id/btn_nine"
        app:layout_constraintStart_toEndOf="@+id/btn_seven"
        app:layout_constraintTop_toBottomOf="@+id/btn_five" />

    <Button
        android:id="@+id/btn_nine"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:layout_marginEnd="20dp"
        android:text="9"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/btn_six" />

    <Button
        android:id="@+id/btn_zero"
        android:layout_width="wrap_content"
```

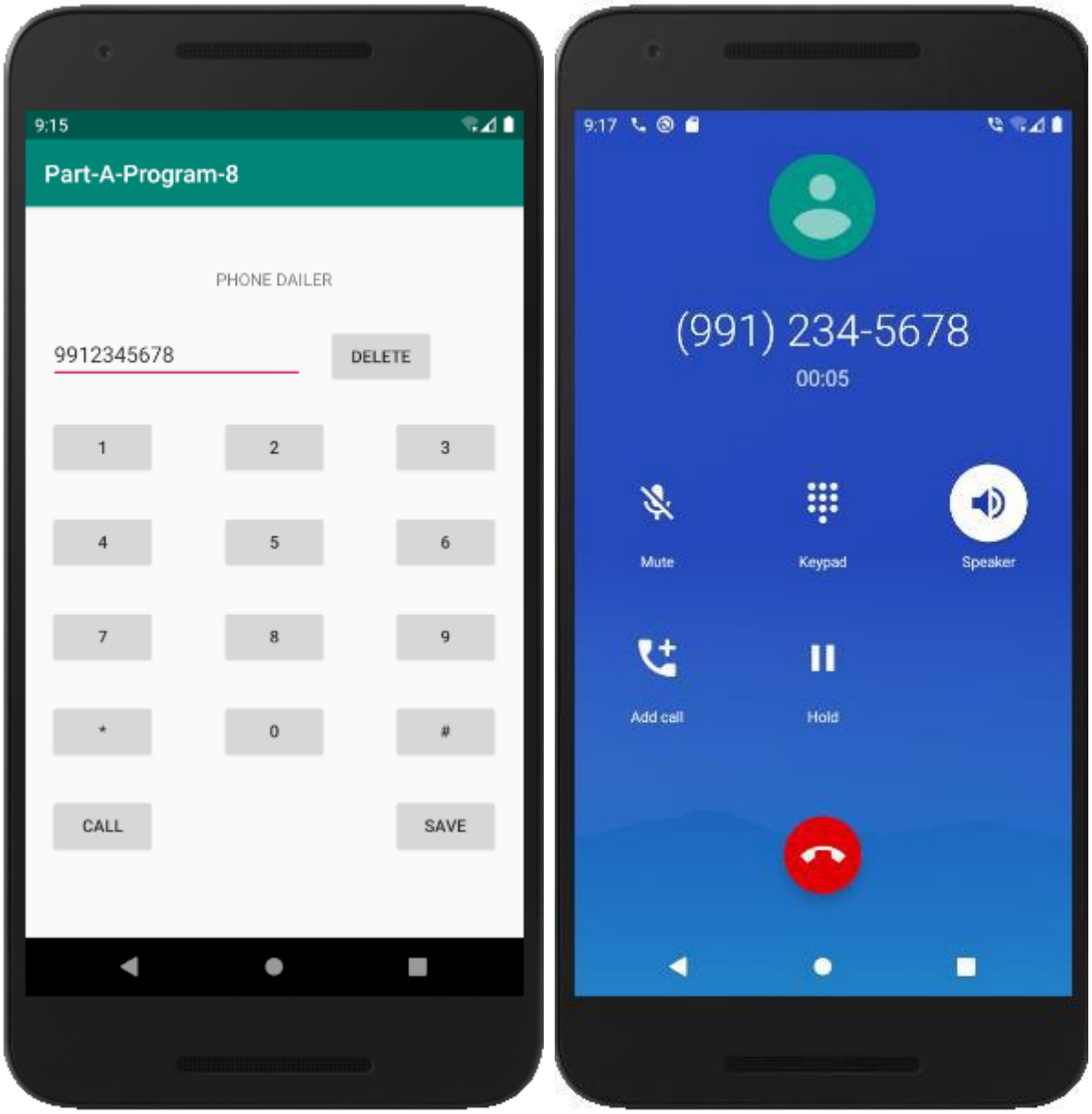
```
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"    android:text="0"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/btn_eight" />

    <Button
        android:id="@+id/btn_call"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
        android:text="Call"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/btn_zero" />

    <Button
        android:id="@+id/btn_save"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:layout_marginEnd="20dp"
        android:text="Save"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/btn_zero" />

    <Button
        android:id="@+id/btn_start"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="30dp"
        android:text="*"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/btn_seven" />

    <Button
        android:id="@+id/btn_hash"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:layout_marginEnd="20dp"
        android:text="#"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/btn_nine" />
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest                xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.part_a_program_8">

    <uses-permission android:name="android.permission.CALL_PHONE"/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

    </application>

    </manifest>
```

Sample Output



Sample Viva Questions

1. What is an Android?
2. Illustrate the Android lifecycle Activity.
3. What Is an Android SDK?
4. What is Android "compatibility"?
5. What are the key components Android Architecture?
6. Describe the Android Framework.
7. What are the data types supported by AIDL?
8. What is Gradle Framework ?
9. Why do we need AVD?
10. How do you add gradle dependencies ?
11. What is the difference between Mobile Application Testing and Mobile Testing?
12. What are the different data storage options available on the Android platform?
13. The list of data storage options on the Android platform.
14. Describe Activities.
15. What are Intents? What are the types of Intents?
16. What is Application class?
17. What is a View?
18. What is a view Group?
19. What are these UI components that we can use in our application?
20. Define Constraint Layout. 21. Why did we need Constraint Layout?
22. What are the different types of Android widgets?
23. Which are the files that demonstrate implementing and using of the custom widget?
24. Enumerate the three key loops while monitoring an activity?
25. What are the major steps involved in creating a bounded service through AIDL