

# **RNS INSTITUTE OF TECHNOLOGY**

**Dr. VISHNUVARDHAN ROAD, CHANNASANDRA, BENGALURU - 560 098**



## **Department of Information Science & Engineering**

### **MOBILE APPLICATION DEVELOPMENT LABORATORY MANUAL**

**VI Semester**

**18CSMP68**

**Faculty-in-charge**

**Mrs. Shwetha G N**

**Ms. Soumya S K**

**Ms. Priyanka M R**

# RNS INSTITUTE OF TECHNOLOGY

Dr. VISHNUVARDHAN ROAD, CHANNASANDRA, BENGALURU -560 098

## Department of Information Science and Engineering



### VISION of the College

Building RNSIT into a World - Class Institution

### MISSION of the College

**To impart high quality education** in Engineering, Technology and Management with a difference, enabling students to excel in their career by

1. Attracting quality Students and preparing them with a strong foundation in fundamentals so as *to achieve distinctions in various walks of life* leading to outstanding contributions.
2. Imparting value based, need based, and choice based and skill based professional education to the aspiring youth and *carving them into disciplined, World class Professionals with social responsibility.*
3. Promoting excellence in Teaching, Research and Consultancy that galvanizes academic consciousness among Faculty and Students.
4. Exposing Students to emerging frontiers of knowledge in various domains and make them suitable for Industry, Entrepreneurship, Higher studies, and Research & Development.
5. Providing freedom of action and choice for all the Stake holders with better visibility.

### VISION of the Department

Fostering winning professionals of Strong Informative Potentials.

### MISSION of the Department

Imparting high quality education in the area of Information Science so as to graduate the students with **good fundamentals**, "**Information System Integration**", "**Software Creation**" capability & suitably train them to thrive in **Industries, higher schools of learning** and **R & D centers** with a comprehensive perspective.

**PROGRAM EDUCATIONAL OBJECTIVES (PEOs)**

**ISE Graduates** within three-four years of graduation will have

- **PEO1:** Acquired the fundamentals of computers and applied knowledge of Information Science & Engineering and continue to develop their technical competencies by problem solving using programming.
- **PEO2:** Ability to formulate problems, attained the Proficiency to develop system/application software in a scalable and robust manner with various platforms, tools and frameworks to provide cost effective solutions.
- **PEO3:** Obtained the capacity to investigate the necessities of the software Product, adapt to technological advancement, promote collaboration and interdisciplinary activities, Protecting Environment and developing Comprehensive leadership.
- **PEO4:** Enabled to be employed and provide innovative solutions to real-world problems across different domains.
- **PEO5:** Possessed communication skills, ability to work in teams, professional ethics, social responsibility, entrepreneur and management, to achieve higher career goals, and pursue higher studies.

**PROGRAM OUTCOMES (POs)**

**Engineering Graduates** will be able to:

- **PO1: Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization for the solution of complex engineering problems
- **PO2: Problem analysis:** Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.
- **PO3: 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 public health and safety, and cultural, societal, and environmental considerations.
- **PO4: 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.
- **PO5: Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modeling to complex engineering activities, with an understanding of the limitations.
- **PO6: 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.

- **PO7: 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.
- **PO8: Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- **PO9: Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- **PO10: Communication:** Communicate effectively on complex engineering activities with the engineering community and with the 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.
- **PO11: 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.
- **PO12: 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.

#### **PROGRAM SPECIFIC OUTCOMES (PSOs)**

**ISE Graduates** will have

- **PSO1 – Problem Solving Abilities:** Ability to demonstrate the fundamental and theoretical concepts, analyze the real time problems and develop customized software solutions by applying the knowledge of mathematics and algorithmic techniques.
- **PSO2 – Applied Engineering Skills:** Enable creative thinking, Ability to apply standard practices and strategies, technical skills in software design, development, integration of systems and management for improving the security, reliability and survivability of the infrastructure.
- **PSO3 – General Expertise and Higher Learning:** Ability to exchange knowledge effectively demonstrate the ability of team work, documentation skills, professional ethics, entrepreneurial skills and continuing higher education in the field of Information technology.

## Course Details

**Course Name :Mobile Application Development**

**Course Code : 18CSMP68**

**Course prerequisite: Core Java**

## Course Objectives

**Upon completion of this course, students are expected to:**

1. Learn and acquire the art of Android Programming.
2. Configure Android studio to run the applications.
3. Understand and implement Android's User interface functions.
4. Create, modify and query on SQLite database.
5. Inspect different methods of sharing data using services.

## Course Outcomes

After successful completion of the Course, the participants will be able to

Course Outcomes	
<b>18CSMP68.1</b>	Learn and acquire the art of Android Programming.
<b>18CSMP68.2</b>	Configure Android studio to run the applications.
<b>18CSMP68.3</b>	Understand and implement Android's User interface functions.
<b>18CSMP68.4</b>	Create, modify and query on SQLite database.
<b>18CSMP68.5</b>	Inspect different methods of sharing data using services.
<b>18CSMP68.6</b>	Design and develop a mini project to demonstrate various android application.

Course Outcomes – Program Outcomes Correlation Matrix																
COURSE OUTCOMES		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
C18CSMP68.1	CO1	3	1	2		3				3				2	2	
C18CSMP68.2	CO2	3	1	2		3				2				1		
C18CSMP68.3	CO3	2	1	3		3				3				2	2	
C18CSMP68.4	CO4	3	2	3		3				3				1	2	
C18CSMP68.5	CO5	3	2	3		3				3				2	2	
C18CSMP68.6	CO6			3	3	2				2	2	1	2			2

### Procedure to Conduct Practical Examination

#### Experiment distribution

- For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
- For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

**Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.**

#### Marks Distribution (Coursed to change in accordance with university regulations)

- For laboratories having only one part –  
Procedure + Execution + Viva-Voce:  $15+70+15= 100$  Marks
- For laboratories having PART A and PART B  
Part A – Procedure + Execution + Viva =  $6 + 28 + 6 = 40$  Marks  
Part B – Procedure + Execution + Viva =  $9 + 42 + 9 = 60$  Marks

## **SYLLABUS**

### **MOBILE APPLICATION DEVELOPMENT**

**Subject Code: 18CSMP68**

**IA Marks: 40**

**No. of Practical Hrs. / Week: 0:0:2**

**Exam Marks: 60**

**Total No. of Practical Hrs: 3 Hours/Week**

**Exam Hours: 03**

**No. of Credits: 02**

---

#### **Descriptions (if any):**

1. The installation procedure of the Android Studio/Java software must be demonstrated and carried out in groups.
2. Students should use the latest version of Android Studio/Java/ Kotlin to execute these programs. Diagrams given are for representational purposes only, students are expected to improvise on them.
3. Part B programs should be developed as an application and are to be demonstrated as a mini project in a group by adding extra features or the students can also develop their application and demonstrate it as a mini-project. (Projects/programs are not limited to the list given in Part B).

## PART A

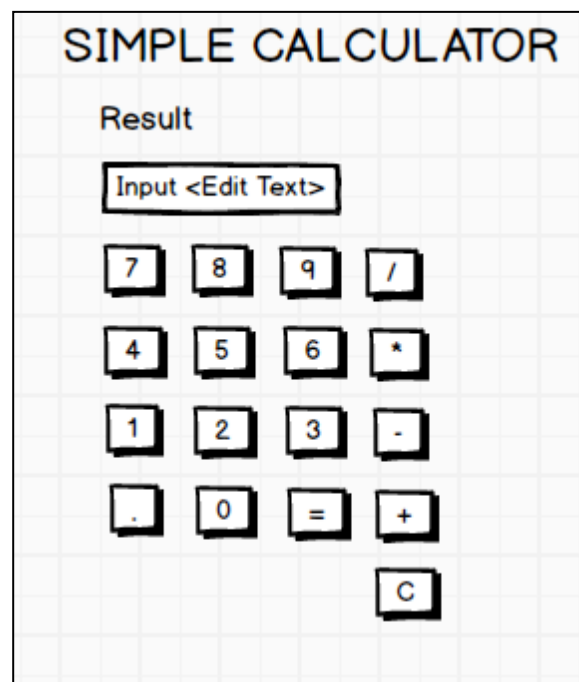
### Program 1

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.



### 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.





### 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.

SIGNUP ACTIVITY	
Username:	<input type="text"/>
Password:	<input type="password"/>
<input type="button" value="SIGN UP"/>	

LOGIN ACTIVITY	
Username:	<input type="text"/>
Password:	<input type="password"/>
<input type="button" value="SIGN IN"/>	

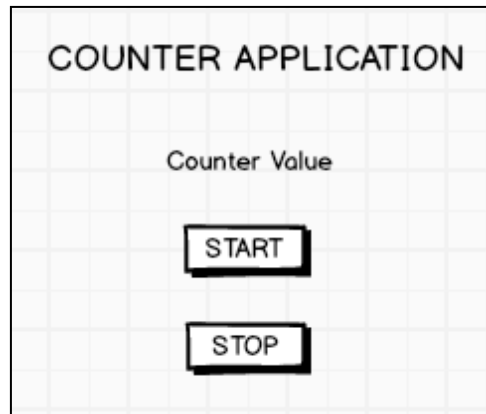
### Program 4

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.

CHANGING WALLPAPER APPLICATION
<input type="button" value="CLICK HERE TO CHANGE WALLPAPER"/>

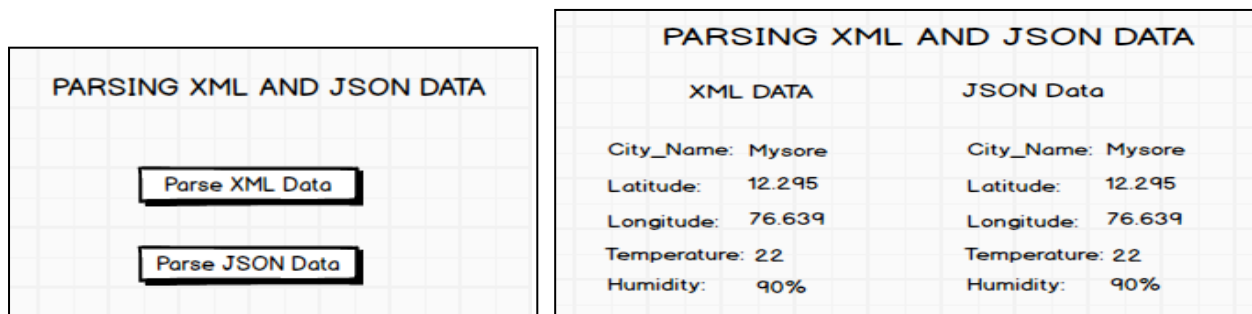
### Program 5

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 TextView control.



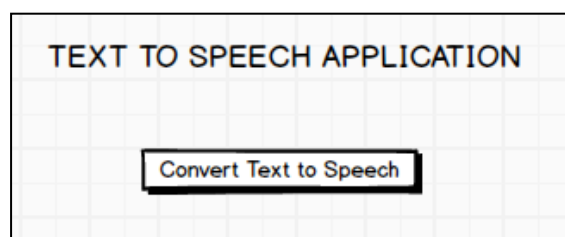
### Program 6

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.



### Program 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.



### 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.

# CALL AND SAVE APPLICATION

1234567890

DEL

1

2

3

4

5

6

7

8

9

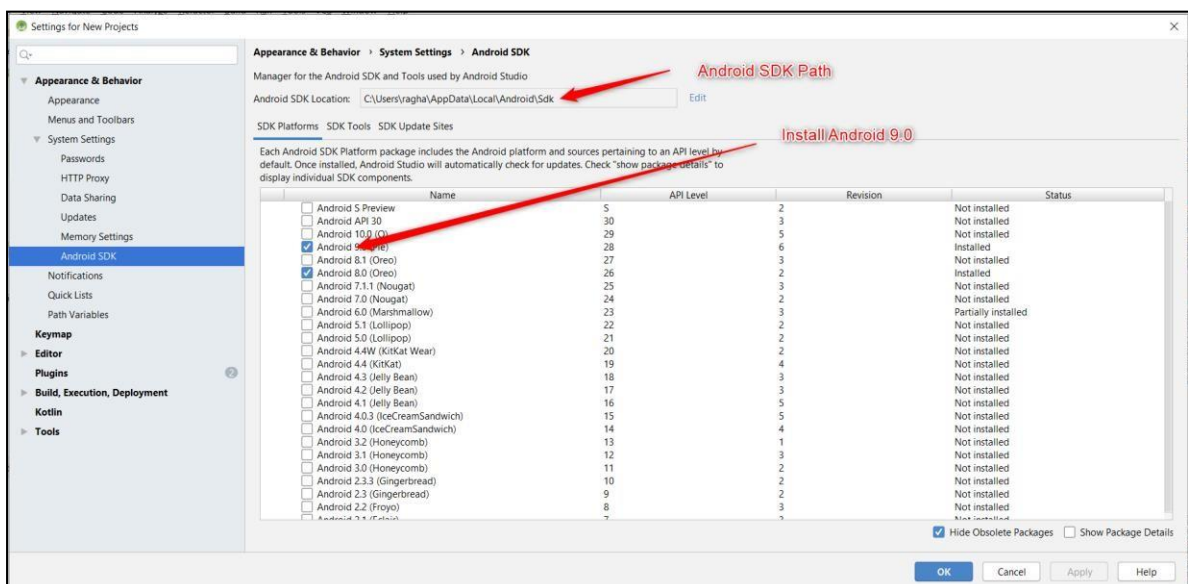
\*

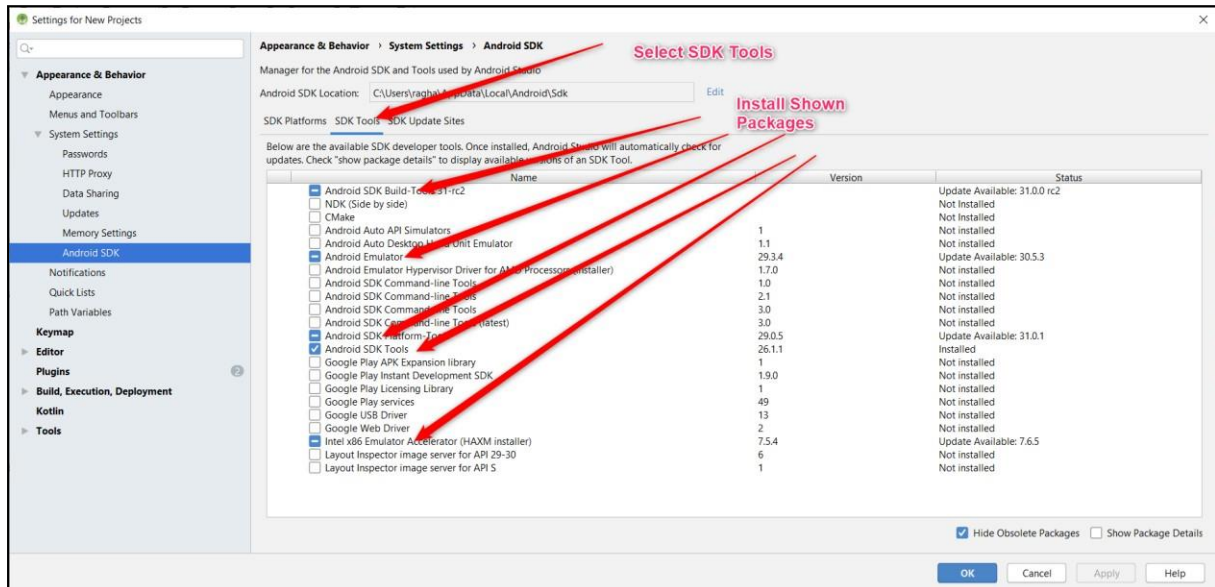
0

#

CALL

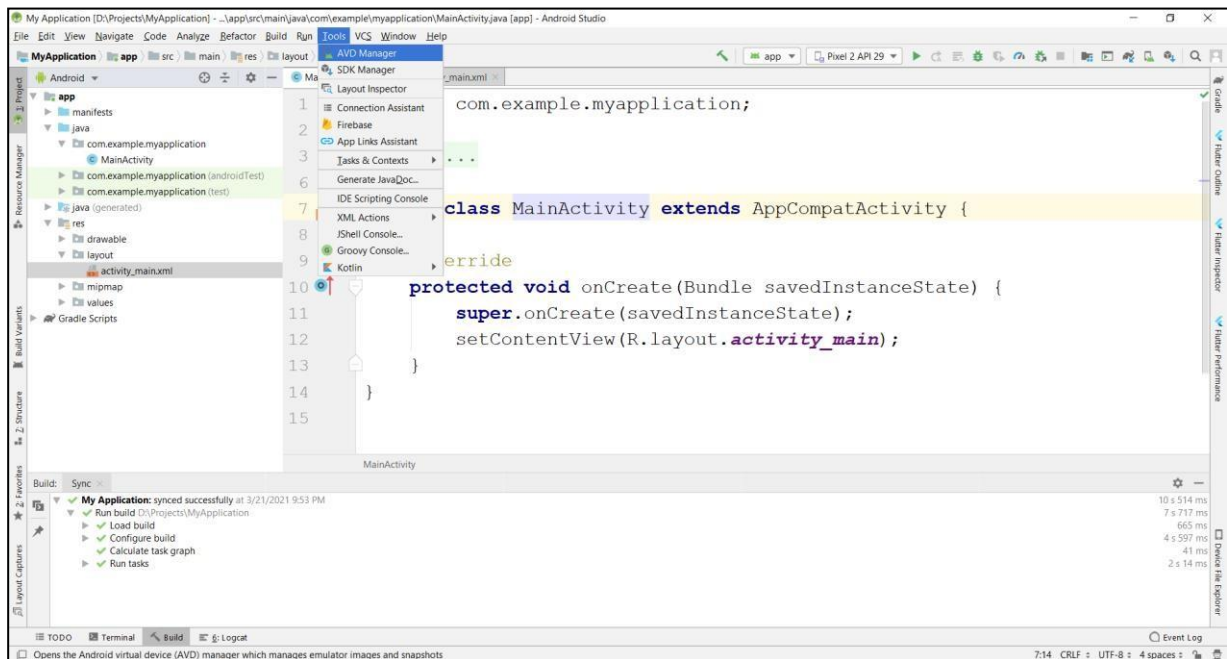
SAVE

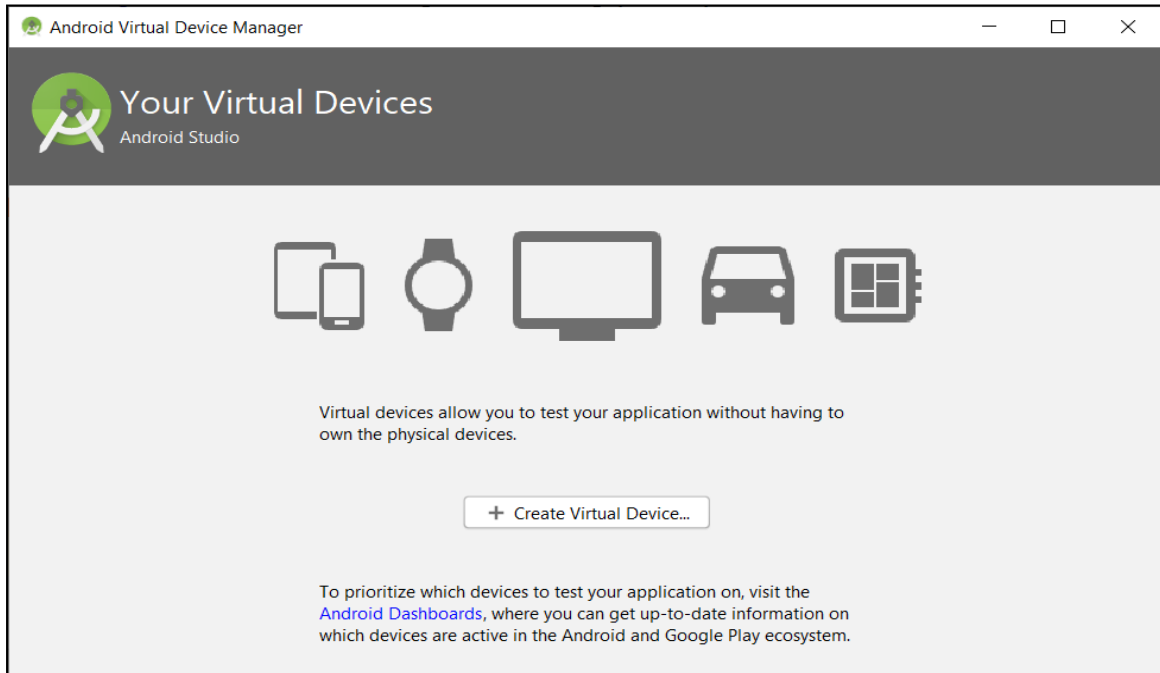




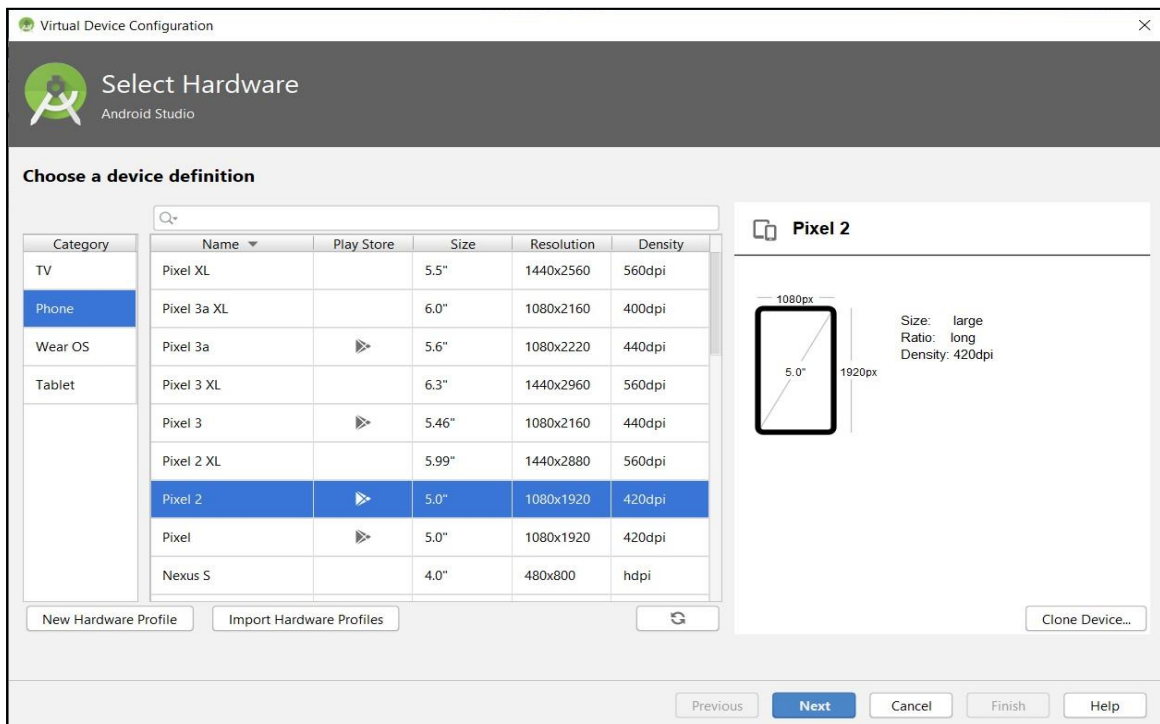
## 1.3 Creating Emulator

Go to Tools → Select AVD Manager

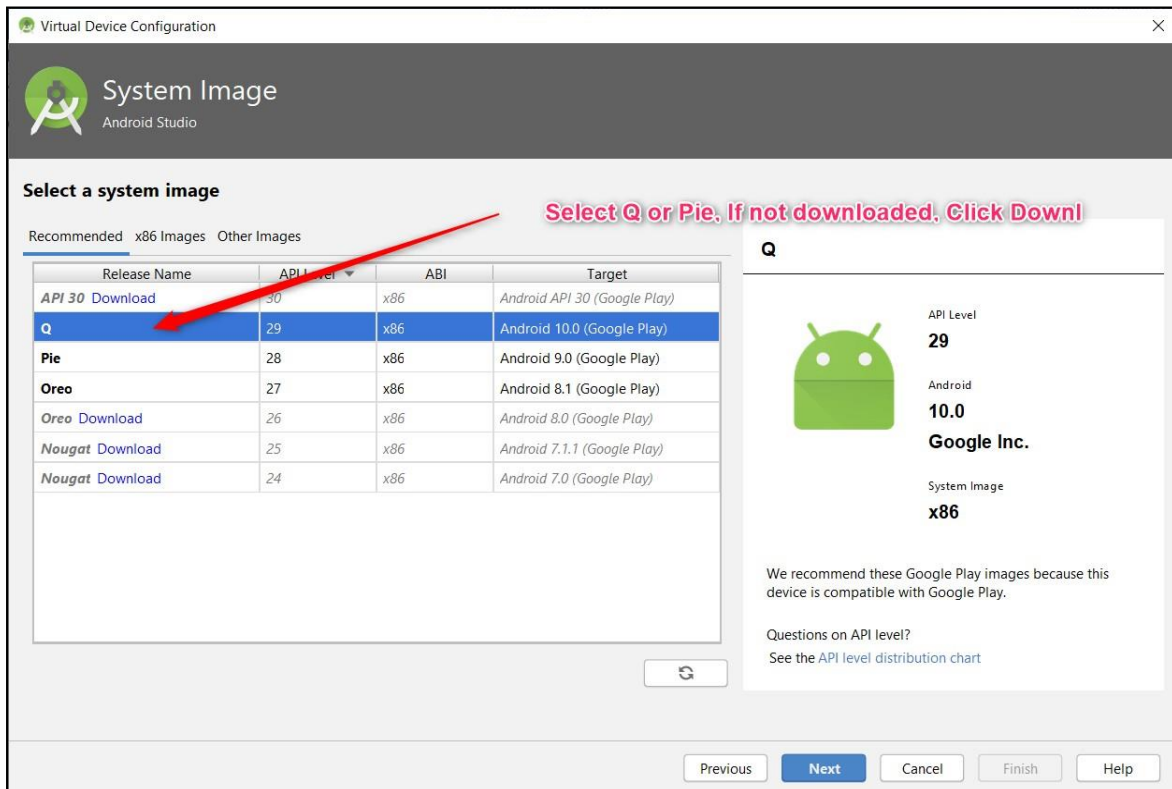




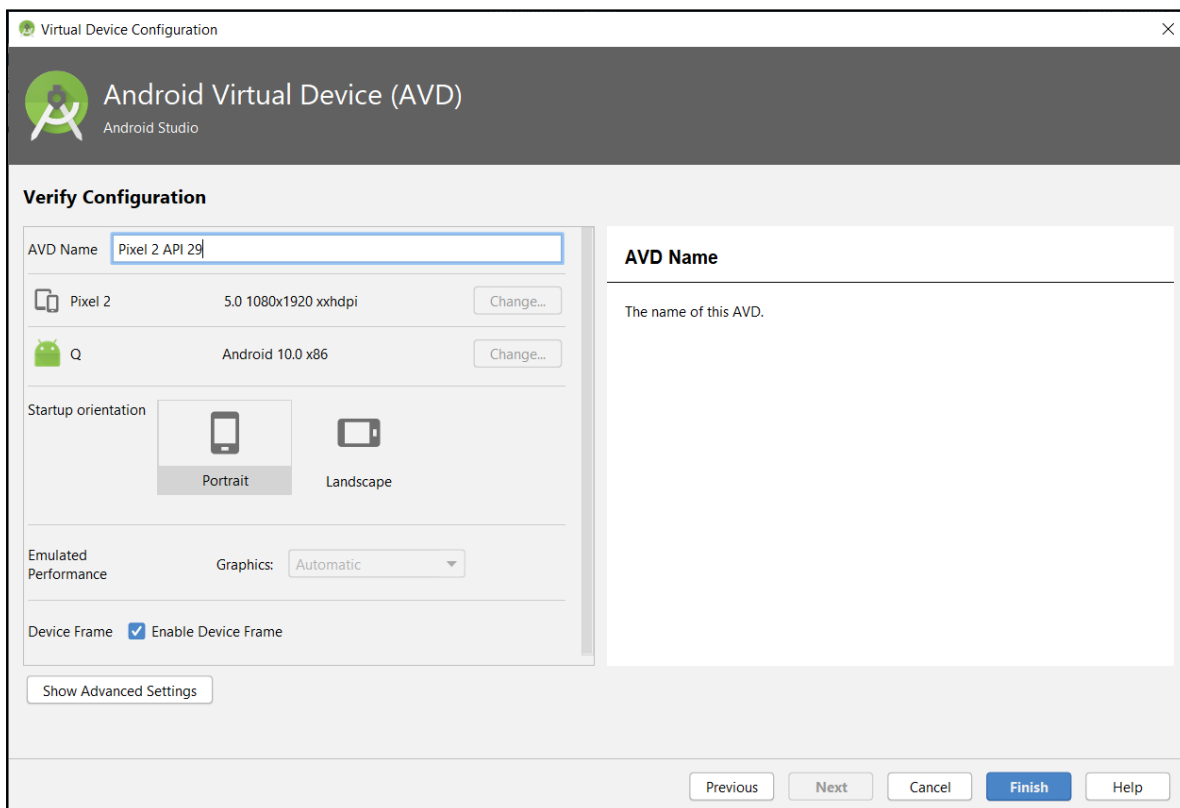
Select Create Virtual Device → Select Phone → Pixel 2 → Press Next



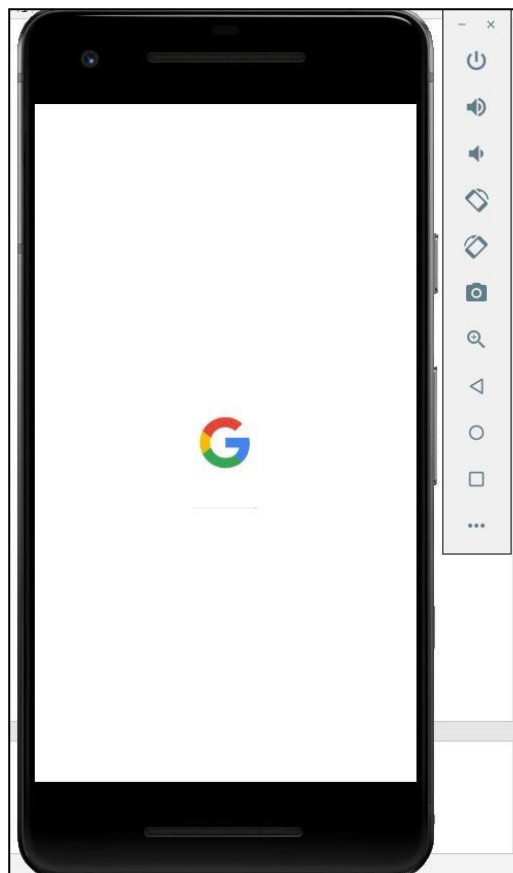
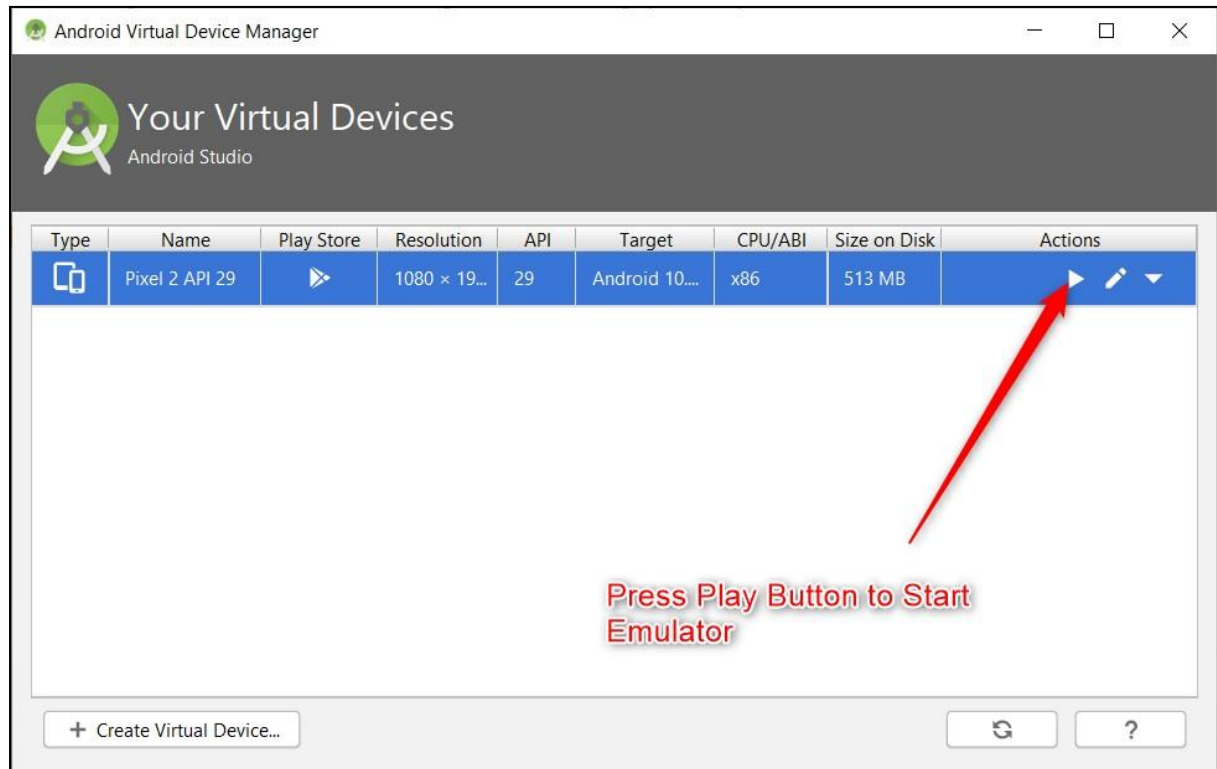
Select Android Q, if not already downloaded press download, After download completes Select Q and Press Next Button.



Enter AVD Name and Press Finish.



Press Play Button to Start Emulator

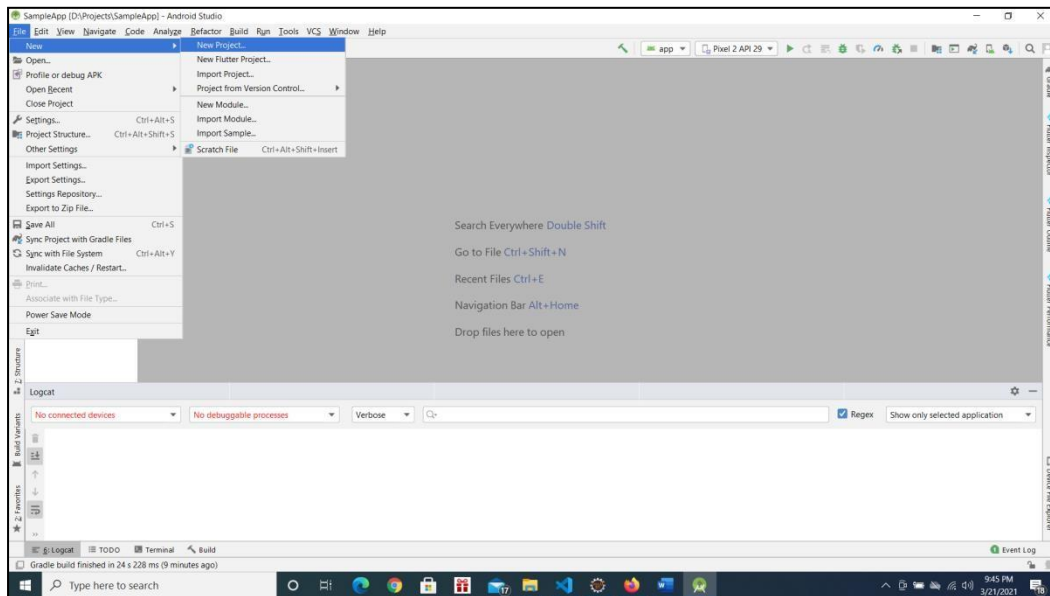




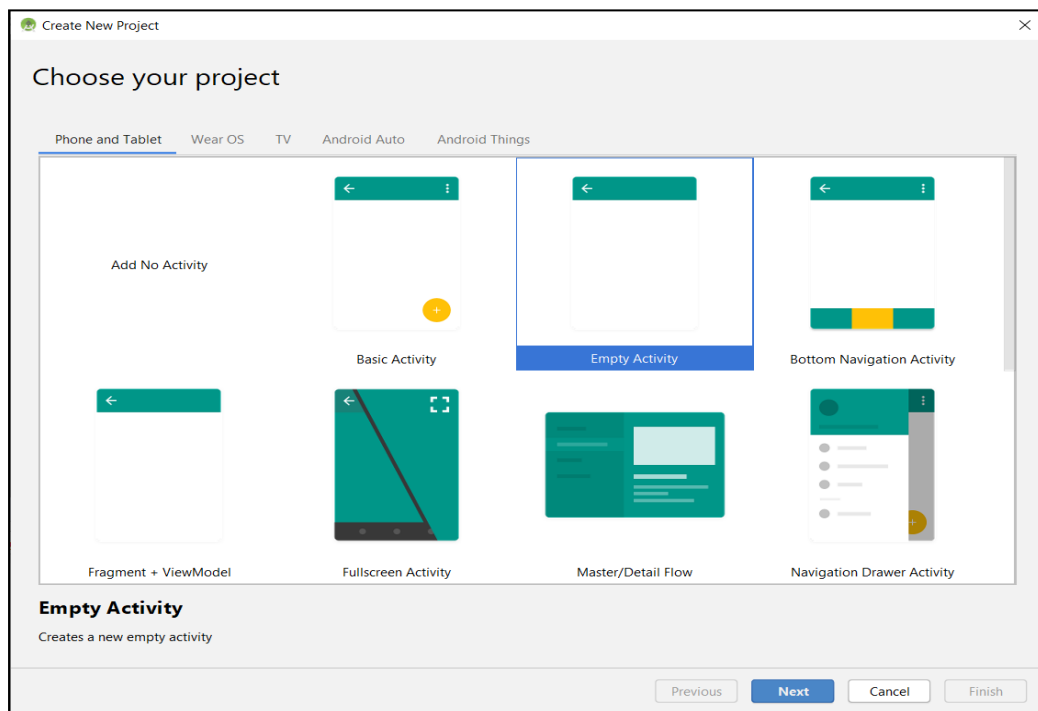
## 1.4 Creating a New Project in Android

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 HomeScreen Icon.

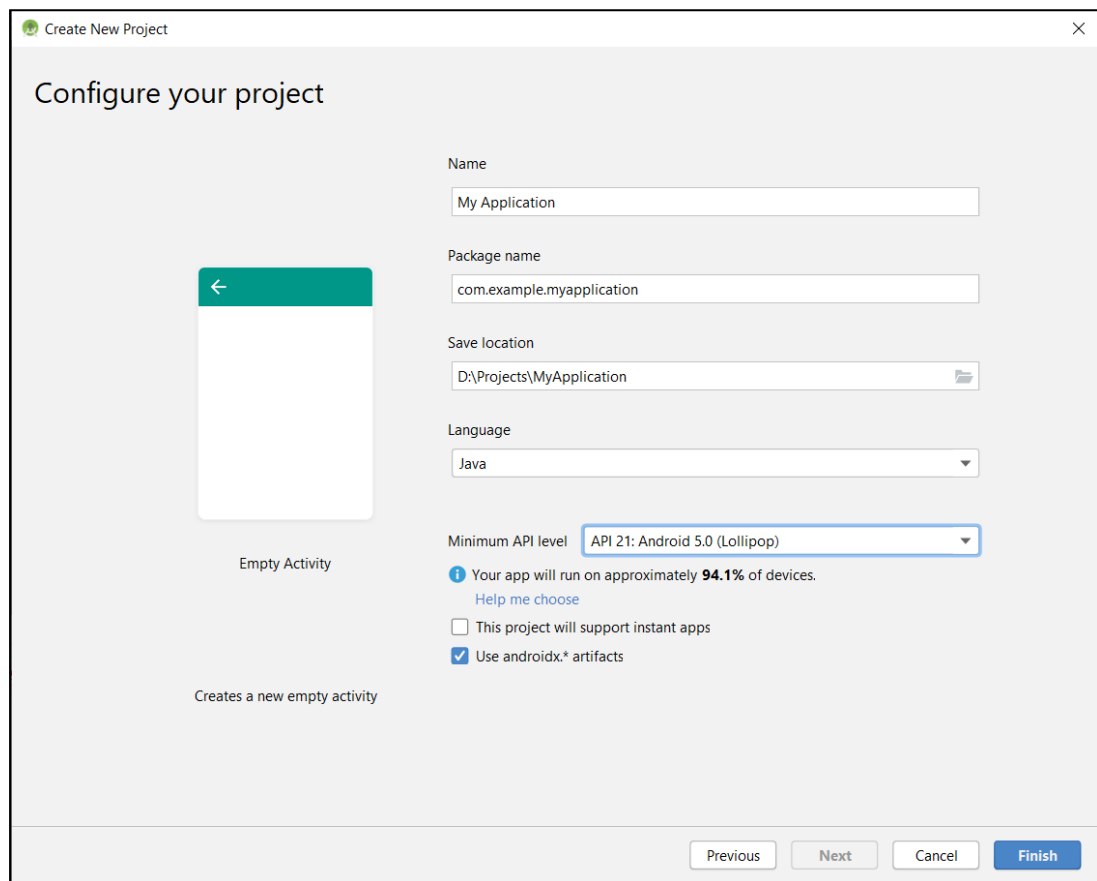
Package Name → Enter package name atleast two identifier (Eg: com.example). Best Practice is 3 or more identifier (Eg: com.example.firstapp).

Save Location → Location where to save the

ProjectLanguage → Choose Java

Minimum API Level → Android 5.0

Select Checkbox Use androidx.artifacts folder as below screenshot.

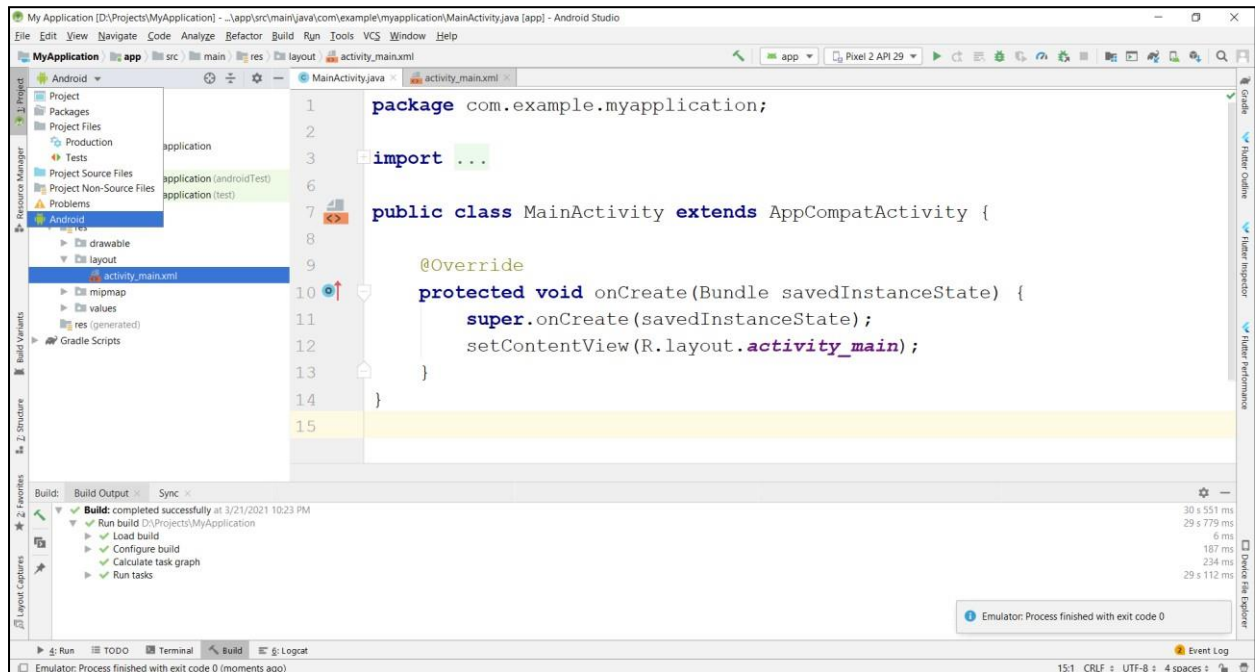


The screenshot shows the 'Create New Project' dialog in Android Studio. The title bar says 'Create New Project'. The main heading is 'Configure your project'. On the left, there is a preview of an 'Empty Activity' with a green back arrow and the text 'Creates a new empty activity'. On the right, there are several input fields and checkboxes:

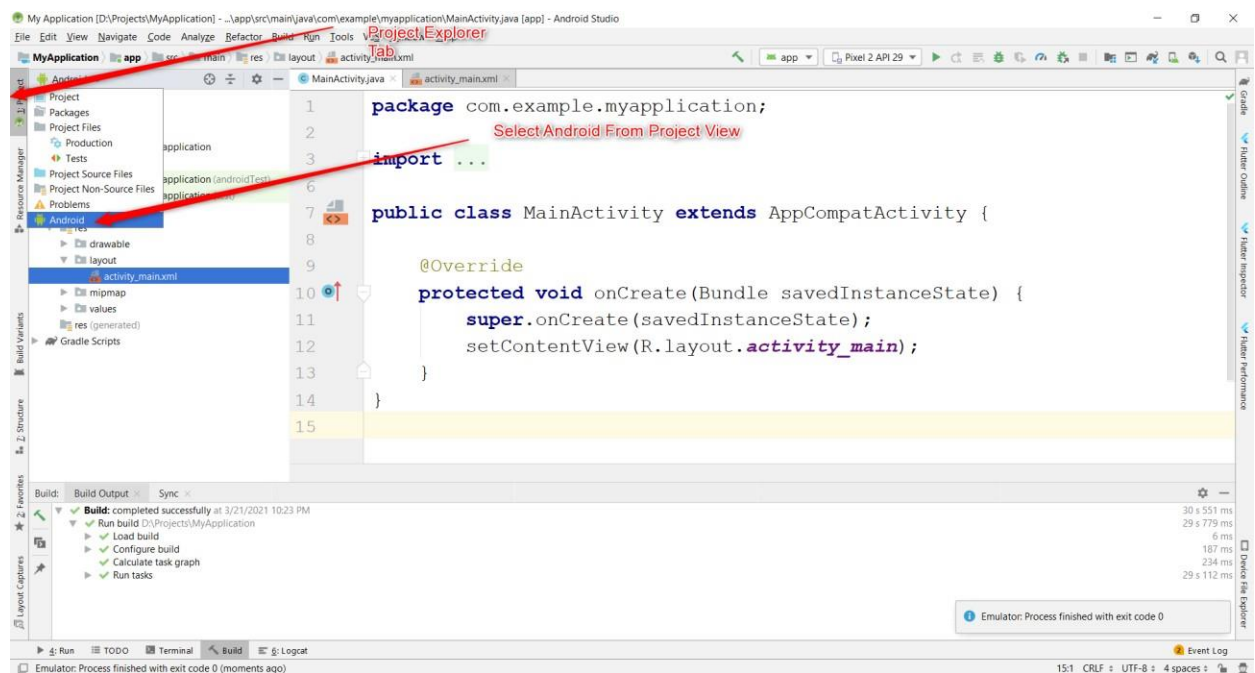
- Name:** My Application
- Package name:** com.example.myapplication
- Save location:** D:\Projects\MyApplication
- Language:** Java
- Minimum API level:** API 21: Android 5.0 (Lollipop)
- Information:** Your app will run on approximately **94.1%** of devices. [Help me choose](#)
- Checkboxes:**
  - ☐ This project will support instant apps
  - ☒ Use androidx.\* artifacts

At the bottom right, there are four buttons: 'Previous', 'Next', 'Cancel', and 'Finish'.

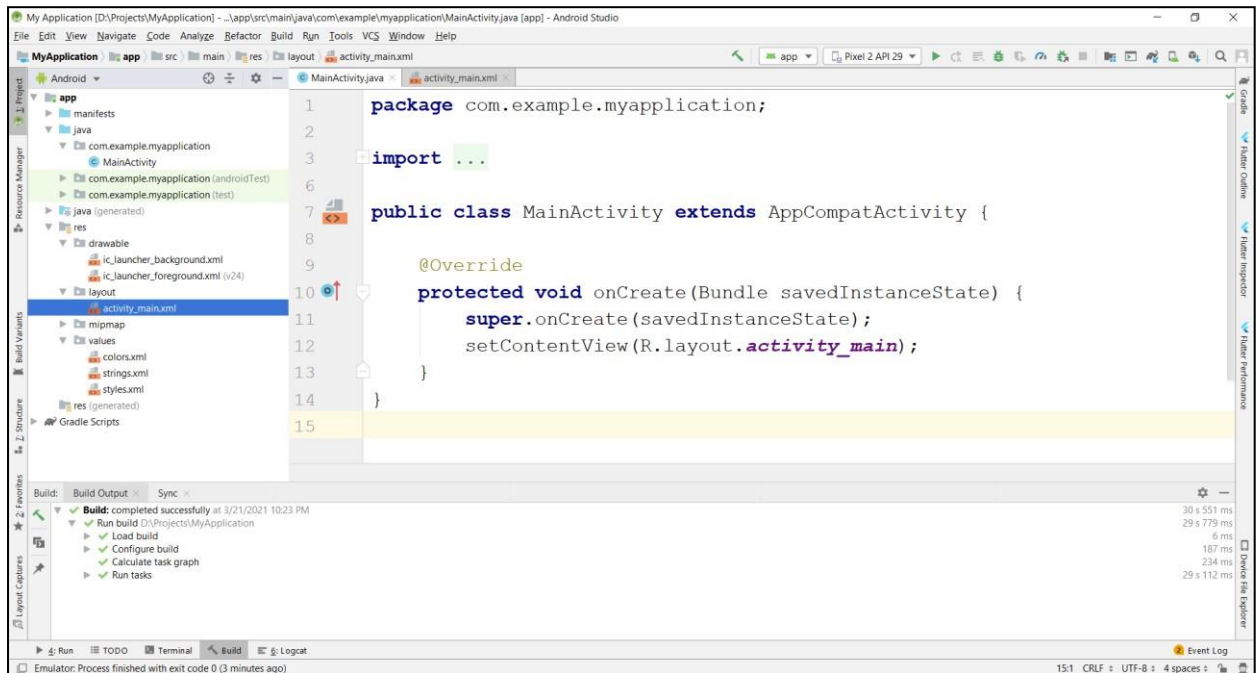
## 1.5 Android Project Structure:



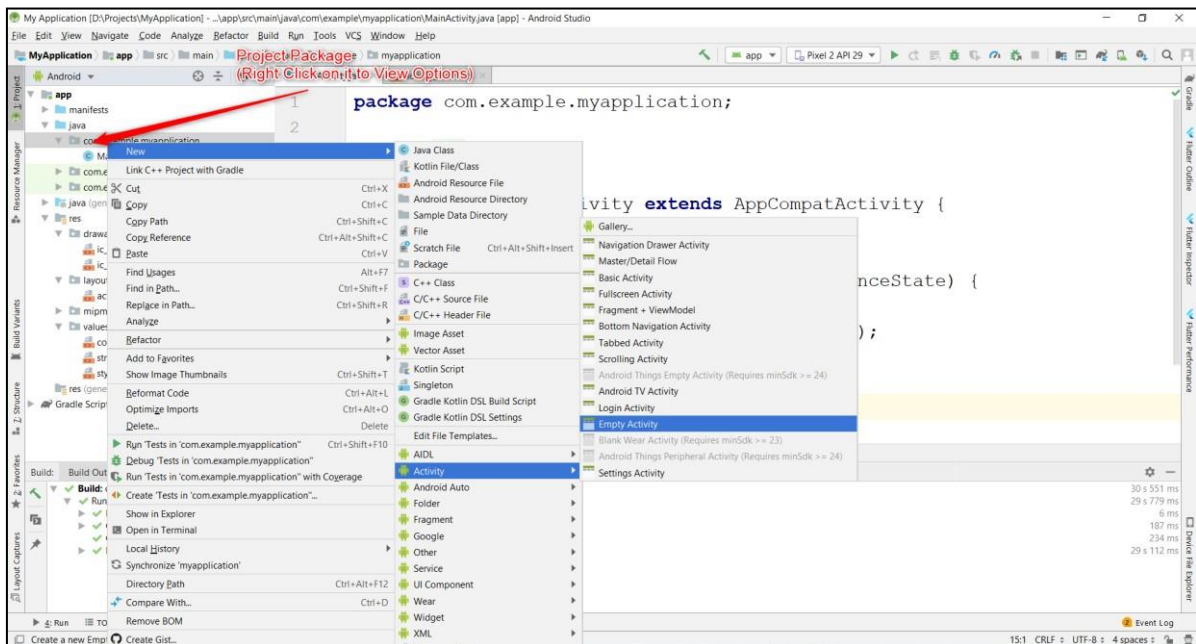
Select Project Explorer and Select Android from Project View



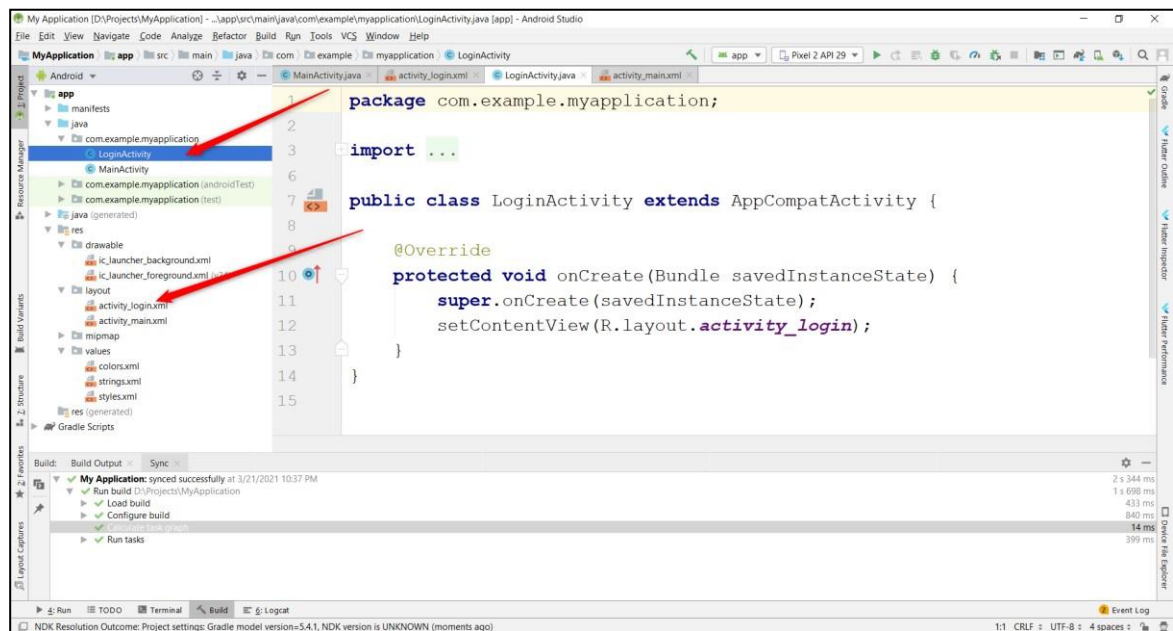
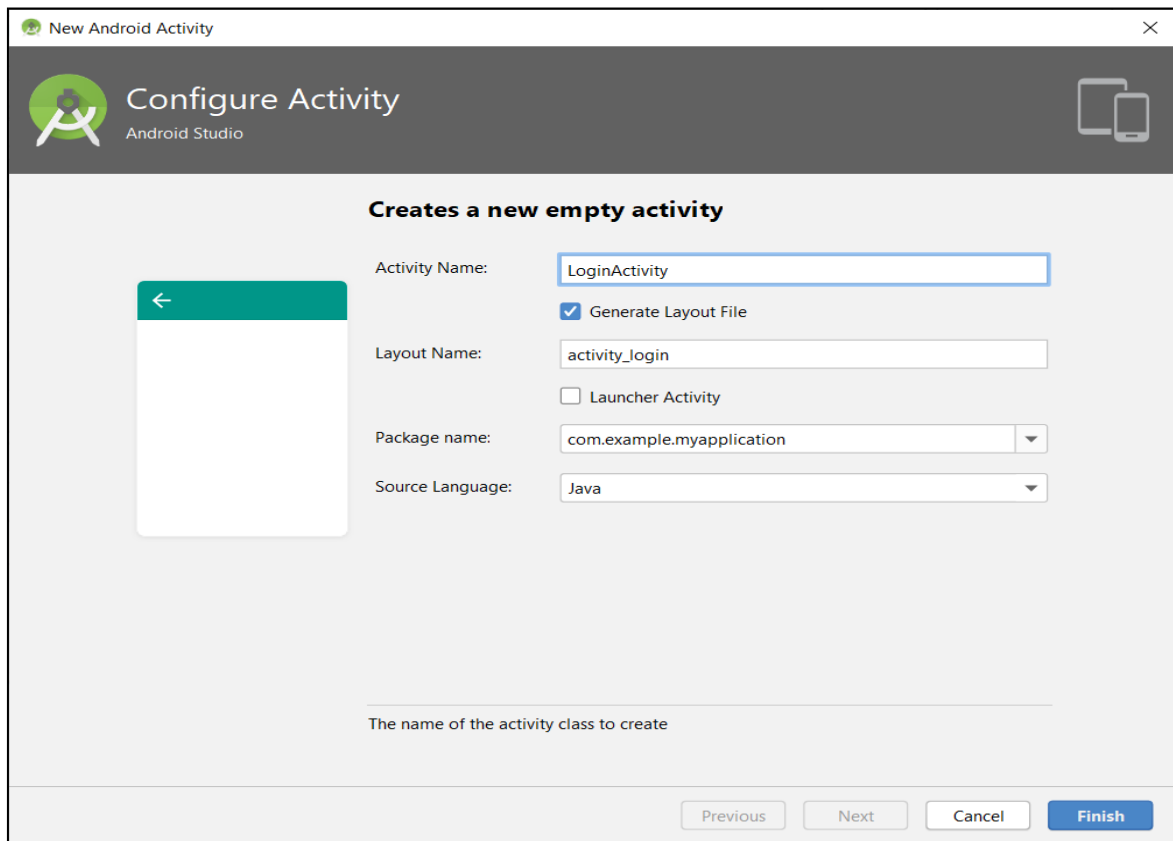
## Basic View:



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

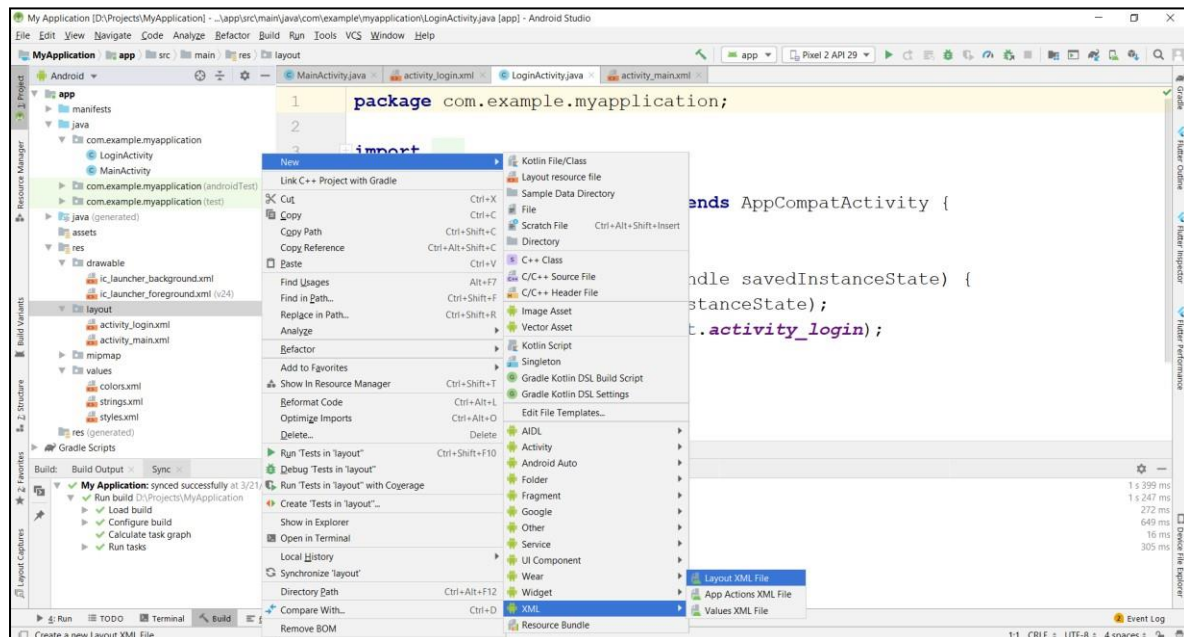


Enter Activity Name and Press Finish

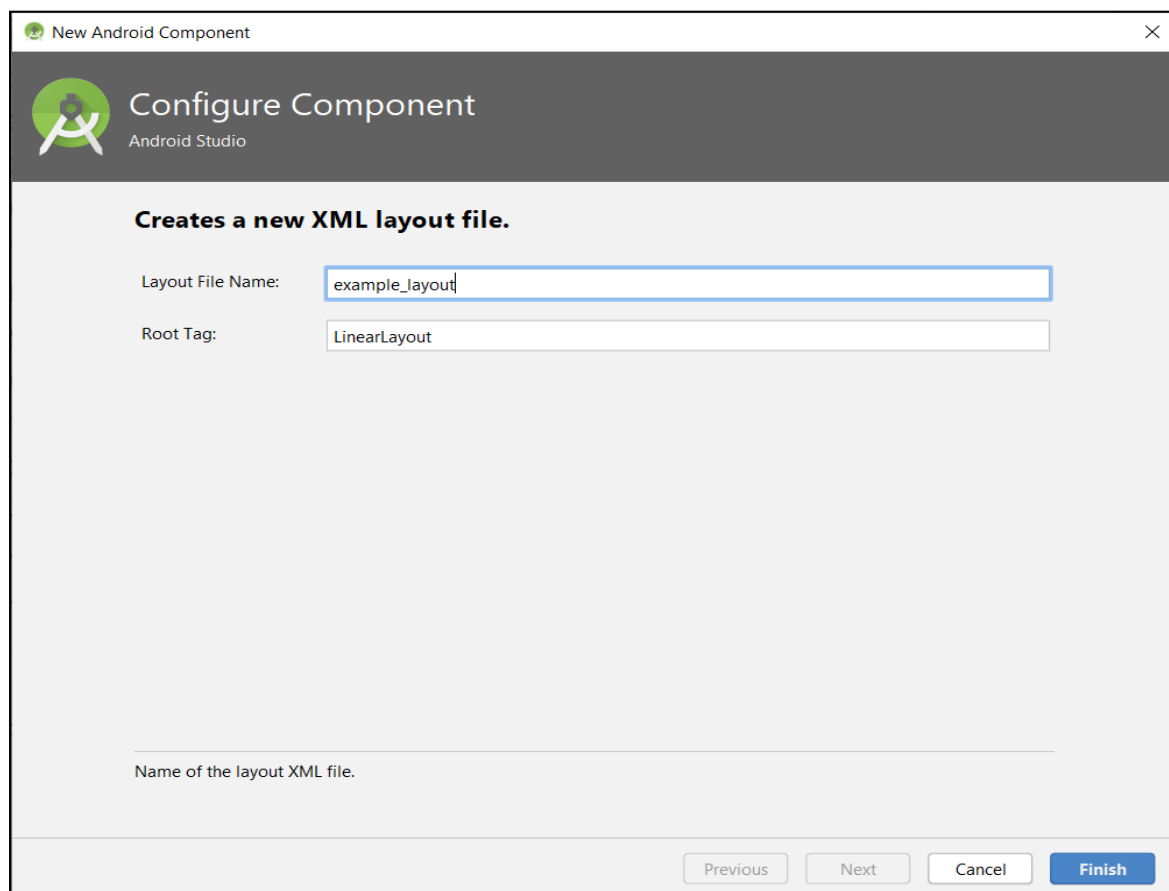


## 1.6 Creating a Layout in Android

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

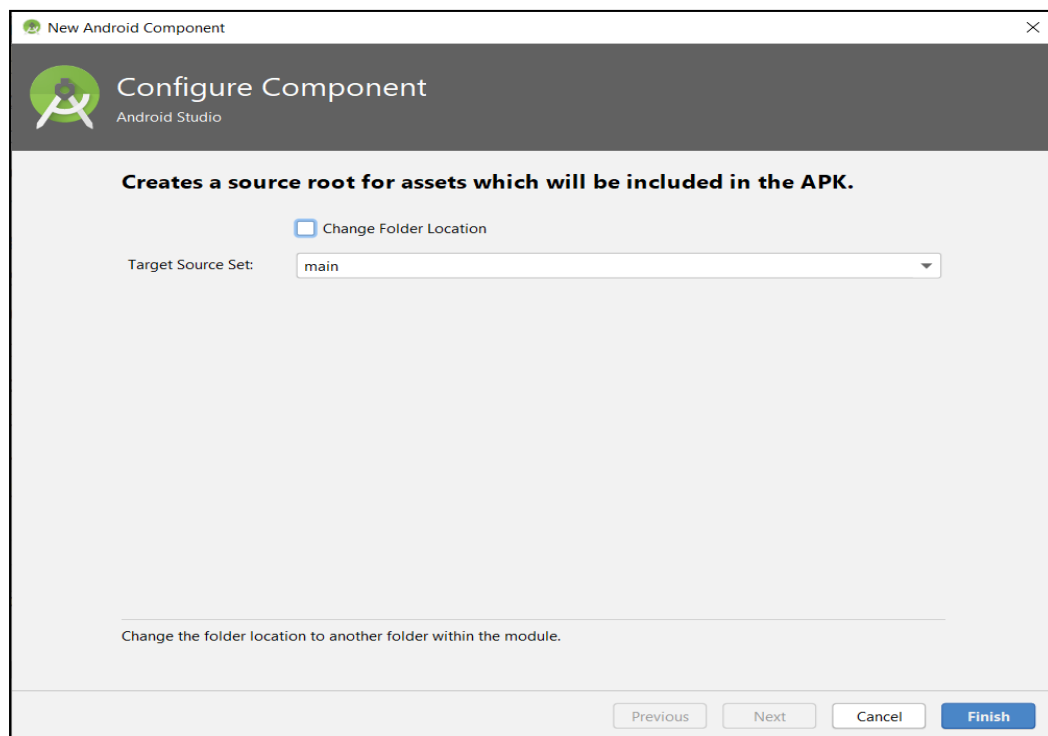
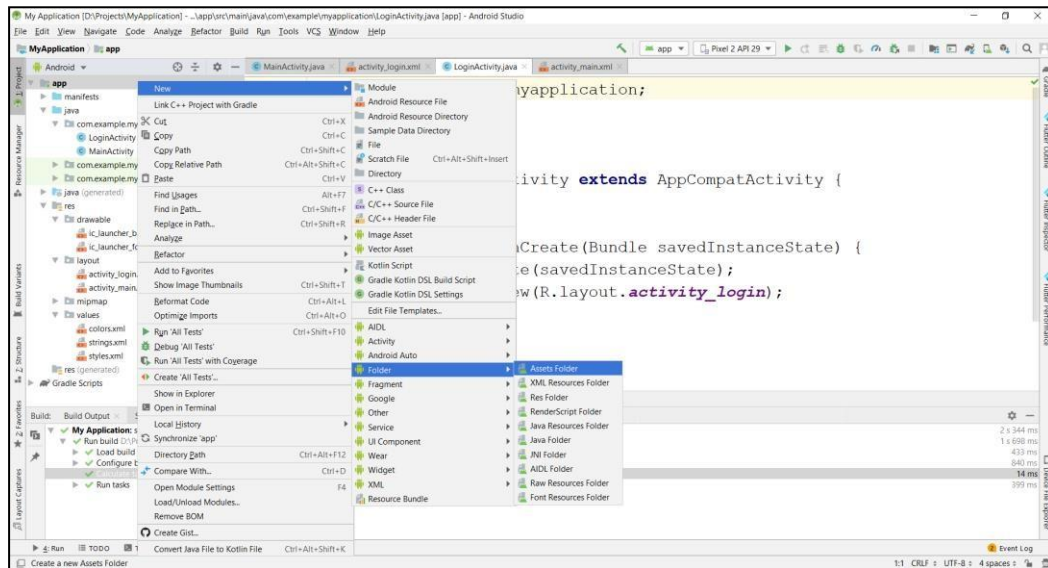


Enter xml file name and press Finish

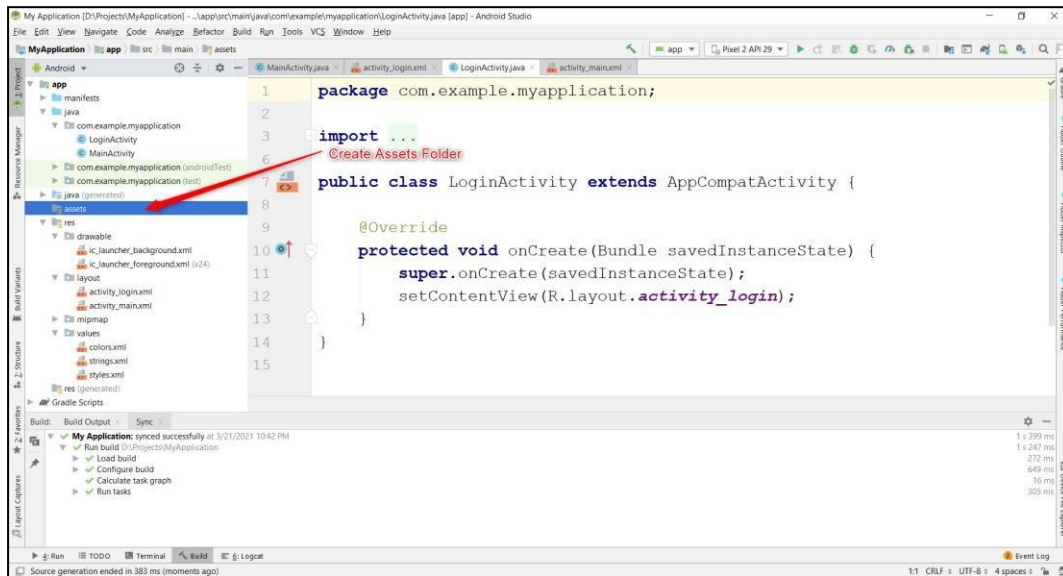


## 1.7 Creating Assets Folder in Android

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

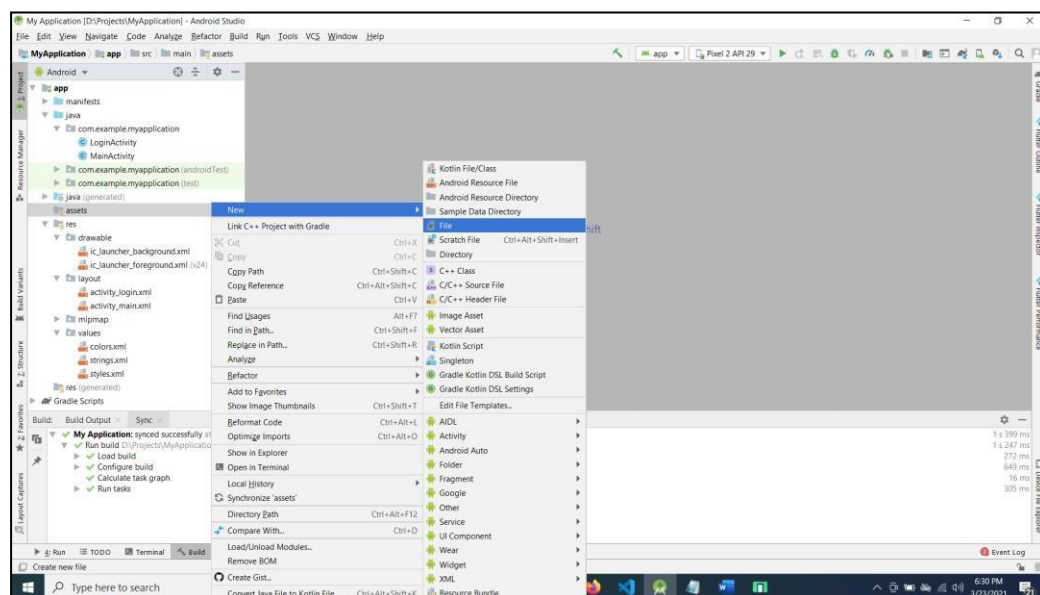




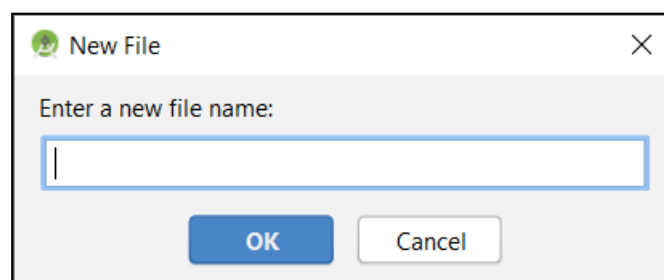


## 1.8 Creating File in assets Folder:

Right Click on assets folder → New → File



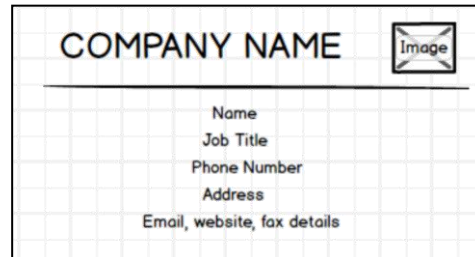
Enter filename with extension (Eg: abc.xml)





## 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 rootview.
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.

## Design



## 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">
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="59dp">

        <TextView
            android:id="@+id/textView"
            android:layout_width="79dp"
            android:layout_height="42dp"
            android:layout_alignParentStart="true"
            android:layout_alignParentBottom="true"
            android:layout_marginStart="113dp"
            android:layout_marginLeft="20dp"
            android:layout_marginBottom="12dp"
            android:gravity="center"
            android:text="RNS IT"
            android:textColor="#E61717"
            android:textSize="20sp" />

        <ImageView
            android:id="@+id/imageView4"
            android:layout_width="48dp"
            android:layout_height="match_parent"
            android:layout_alignParentBottom="true"
            android:layout_marginLeft="88dp"
            android:layout_marginBottom="1dp"
            android:layout_toRightOf="@id/textView"
            app:srcCompat="@drawable/rnsit" />
    </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="Shwetha G N"
    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="Asst prof"
    android:textSize="16dp"
    android:layout_marginBottom="10dp"
    android:layout_marginTop="10dp"
    android:textColor="#000000"
    android:gravity="center" />
<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="+91-9108245345"
    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="Bangalore"
    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:rnsit.ac.in"
        android:textSize="16dp"
        android:layout_marginBottom="10dp"
        android:layout_marginTop="10dp"
        android:textColor="#000000"
        android:gravity="center" />

</LinearLayout>
```

**Sample Output:**

## 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 MainActivity.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



## 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_nine"
    android:layout_width="62dp"
    android:layout_height="48dp"
    android:text="9"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.123"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.467" />

<Button
    android:id="@+id/button_zero"
    android:layout_width="63dp"
    android:layout_height="46dp"
    android:text="0"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.39"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.465"
    tools:ignore="TouchTargetSizeCheck" />

<Button
    android:id="@+id/button_sub"
    android:layout_width="69dp"
    android:layout_height="49dp"
    android:text="-"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.953"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.463" />
```

```
<Button
    android:id="@+id/button_add"
    android:layout_width="65dp"
    android:layout_height="44dp"
    android:text="+"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.656"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.467"
    tools:ignore="TouchTargetSizeCheck" />
```

```
<Button
    android:id="@+id/button_equal"
    android:layout_width="62dp"
    android:layout_height="48dp"
    android:text="="
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.163"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.689" />
```

```
<Button
    android:id="@+id/button_mul"
    android:layout_width="62dp"
    android:layout_height="48dp"
    android:text="*"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.123"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.568" />
```

```
<Button
    android:id="@+id/button_div"
    android:layout_width="63dp"
    android:layout_height="46dp"
    android:text="/"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.39"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.566"
```

```
tools:ignore="TouchTargetSizeCheck" />
```

```
<Button
    android:id="@+id/button_clear"
    android:layout_width="69dp"
    android:layout_height="49dp"
    android:text="C"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.953"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.564" />
```

```
<Button
    android:id="@+id/button_dot"
    android:layout_width="65dp"
    android:layout_height="44dp"
    android:text="."
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.656"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.567"
    tools:ignore="TouchTargetSizeCheck" />
```

```
<Button
    android:id="@+id/button_five"
    android:layout_width="62dp"
    android:layout_height="48dp"
    android:text="5"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.123"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.377" />
```

```
<Button
    android:id="@+id/button_six"
    android:layout_width="63dp"
    android:layout_height="46dp"
    android:text="6"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.39"
    app:layout_constraintStart_toStartOf="parent"
```



```
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.376"
tools:ignore="TouchTargetSizeCheck" />
```

```
<Button
    android:id="@+id/button_eight"
    android:layout_width="69dp"
    android:layout_height="49dp"
    android:text="8"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.953"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.373" />
```

```
<Button
    android:id="@+id/button_seven"
    android:layout_width="65dp"
    android:layout_height="44dp"
    android:text="7"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.656"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.378"
    tools:ignore="TouchTargetSizeCheck" />
```

```
<Button
    android:id="@+id/button_one"
    android:layout_width="62dp"
    android:layout_height="48dp"
    android:text="1"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.123"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.281" />
```

```
<Button
    android:id="@+id/button_two"
    android:layout_width="63dp"
    android:layout_height="46dp"
    android:text="2"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
```

```
app:layout_constraintHorizontal_bias="0.39"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.28"
tools:ignore="TouchTargetSizeCheck" />
```

<Button

```
android:id="@+id/button_three"
android:layout_width="65dp"
android:layout_height="44dp"
android:text="3"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.656"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.282"
tools:ignore="TouchTargetSizeCheck" />
```

<TextView

```
android:id="@+id/textView"
android:layout_width="310dp"
android:layout_height="46dp"
android:text="Simple Calculator"
android:textColor="#C2185B"
android:textSize="25dp"
android:textStyle="bold"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintHorizontal_bias="0.422"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.058" />
```

<TextView

```
android:id="@+id/textView2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Result"
android:textColor="#00796B"
android:textSize="20dp"
android:textStyle="bold"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.07"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.176" />
```

```

<EditText
    android:id="@+id/txt_result"
    android:layout_width="249dp"
    android:layout_height="40dp"
    android:layout_marginTop="20dp"
    android:ems="10"
    android:inputType="textPersonName"
    android:text=""
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.839"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.135"
    tools:ignore="SpeakableTextPresentCheck,TouchTargetSizeCheck" />

<Button
    android:id="@+id/button_four"
    android:layout_width="69dp"
    android:layout_height="49dp"
    android:text="4"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.952"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.277" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

## MainActivity.java

```

package com.example.program2;
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);

    btnZero = (Button) findViewById(R.id.button_zero);
    btnZero.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("");
}
```

@Override

```
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(btnDot))
        txtResult.append(".");
    if (v.equals(btnClear))
        txtResult.setText("");
    if (v.equals(btnAdd))
        txtResult.append("+");
    if (v.equals(btnSub))
```

```

        txtResult.append("-");
    if (v.equals(btnMul))
        txtResult.append("*");
    if (v.equals(btnDiv))
        txtResult.append("/");
    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("-");

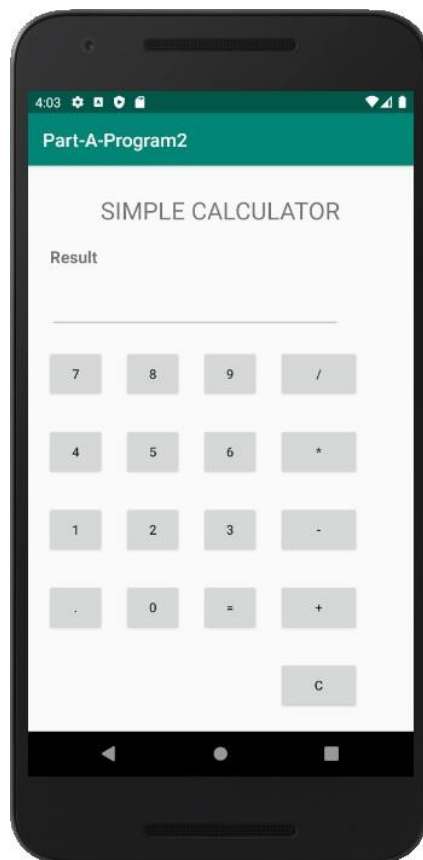
```

```

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

```

## Sample Output



### Program 3

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

- 1. Password should contain uppercase and lowercase letters.**
- 2. Password should contain letters and numbers.**
- 3. Password should contain special characters.**
- 4. 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 messagesaying “Failed Login Attempts” and disable the SIGN IN button. Use Bundle to transfer information from one activity to another.**

The image shows two wireframe layouts for an Android application. The first wireframe, titled 'SIGNUP ACTIVITY', features a light gray grid background. It contains two text input fields: the first is labeled 'Username:' and the second is labeled 'Password:'. Below these fields is a rectangular button labeled 'SIGN UP'. The second wireframe, titled 'LOGIN ACTIVITY', also has a light gray grid background. It contains two text input fields: the first is labeled 'Username:' and the second is labeled 'Password:'. Below these fields is a rectangular button labeled 'SIGN IN'.

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 Signup Layout using Drag and Drop framework design the layout.
4. Create One more Empty Activity LoginActivity using Android Studio Create ActivityFlow (Refer Android Studio Tutorial)
5. Open activity\_login.xml file from res → layout folder, check/add Constraint Layout as the root view.
6. Create Login Layout using Drag and Drop framework.
7. 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.
8. Use Regular Expression  
"`^(?=.*[A-Z])(?=.*[a-z])(?=.*\d)(?=.*[@$!])[A-Za-z\d@$!]{8,}$`" to validate the password.



## Design

The image displays two side-by-side wireframe designs for a mobile application. The left wireframe is for a 'SIGN UP' screen, featuring a title 'SIGN UP' at the top, followed by two input fields labeled 'USERNAME' and 'PASSWORD', and a 'SIGN UP' button at the bottom. The right wireframe is for a 'Login' screen, featuring a title 'Login' at the top, followed by two input fields labeled 'Username' and 'PASSWORD', and a 'LOGIN' button at the bottom. Both screens have a light gray background and a blue border.

## 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="82dp" android:layout_height="34dp"
        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>

```

### Activity\_login.xml

```

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

```

### 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 txt Username;
    EditText txt Password;
    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("Lab@2020", password);
        Intent it = new Intent(this, LoginActivity.class);
        it.putExtra("data", bundle);
        startActivity(it); }

    Toast.makeText(getApplicationContext(), "Invalid Password",
    Toast.LENGTH_LONG).show();
}
}

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

}
}

```

### **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 txt_LoginUsername;
    EditText txt_LoginPassword;
    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("rnsit");
    pass=bundle.getString("Lab@2020");
}

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();
        }
    }
}
}

```

Sample Output

9:56

92%

Program3

SIGN UP

USERNAME

rnsit

PASSWORD

.....

SUBMIT

10:02

91%

Program3

SIGN UP

USERNAME

rnsit

PASSWORD

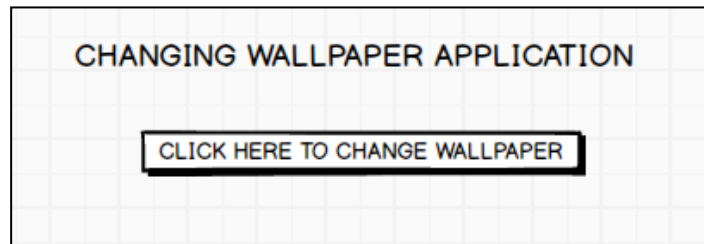
.....

Login Successful

SUBMIT

#### Program 4

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



1. Create a New Android Project with Empty Activity.
2. Open activity\_main.xml file from res→layout folder, check/add LinearLayout as the rootview.
3. Create the layout
4. Add 3 or More images to drawable folder (res→drawable)
5. Declare uses permission android.permission.SET\_WALLPAPER in The AndroidManifest.xml file
6. Schedule Timer task to change the wallpaper on every 30 seconds interval.
7. Initialize and use WallpaperManager.setBitmap() method to change the wallpaper.

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

</LinearLayout>
```

#### MainActivity.java

```
package com.example.program5;
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.img1,
        R.drawable.img2,
        R.drawable.img3,
        R.drawable.img4,
        R.drawable.img5,
        R.drawable.img6,
        R.drawable.img7,
        R.drawable.img8,
        R.drawable.img9,
        R.drawable.img10,

    };
    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);
    }

    @Override
    public void onClick(View view)
    {
        if(!running)
        {
            new Timer().schedule (new MyTimer(),0,3000);
            running=true;
        }
    }
    class MyTimer extends TimerTask {

        public void run() {
            try {
                WallpaperManager wallpaperManager =
                WallpaperManager.getInstance(getBaseContext());

                if(i==12) i=1;

```

```

        if(i==11) i=2;
        if(i==10) i=3;
        if(i==9) i=4;
        if(i==8) i=5;
        if(i==7) i=6;
        if(i==6) i=7;
        if(i==5) i=8;
        if(i==4) i=9;
        if(i==3) i=10;

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

        {
        }

    }
}
}

```

### **AndriodManifest.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_ro
und"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>

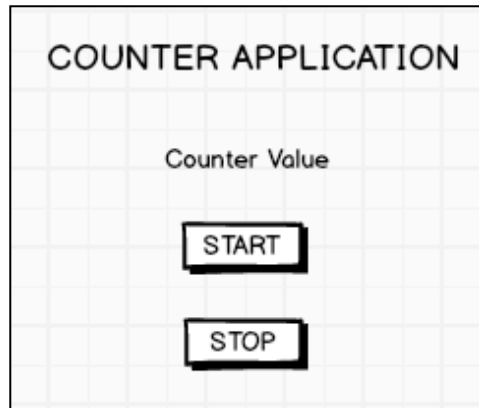
```

## Sample Output



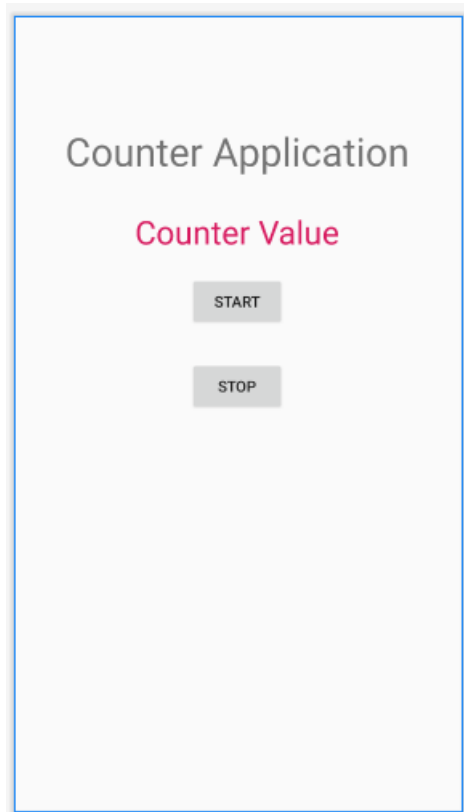
### Program 5

**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.**



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 a Thread to start the counter logic.
6. Steps to Create a Thread
  - Create a class that extends Thread Class.
  - Override run method of Thread Class.
  - Use start() method of thread class to start the thread.
7. Create Handler class to receive message from child thread, Handler executes in MainThread.
8. Steps to Create Handler
  - Create Object of type Handler.
  - Override handleMessage() of handler class.
9. Pass the counter value to be displayed to the handler.
10. Update the UI to display the counter value received from thread.

## Design



## 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="36dp"
        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="30dp"
```

```

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>

```

### 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;

import org.w3c.dom.Text;
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;
    }
}

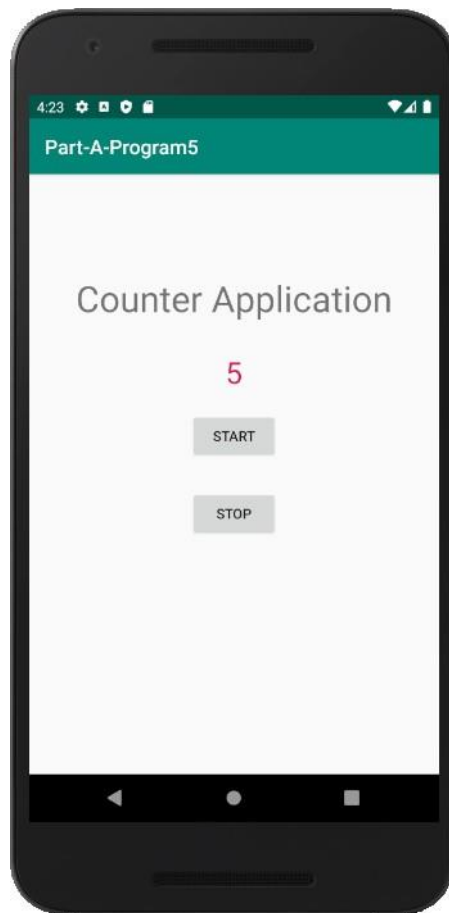
Handler handler=new Handler()
{
    public void handleMessage(Message m)
    {
        lblCounter.setText(String.valueOf(m.what));
    }
};

class MyCounter extends Thread
{
    public void run()
    {
        while(running)
        {
            counter++;
            handler.sendMessage(counter);

            try { Thread.sleep(1000);
            }
            catch(Exception e) { }
        }
    }
}

```

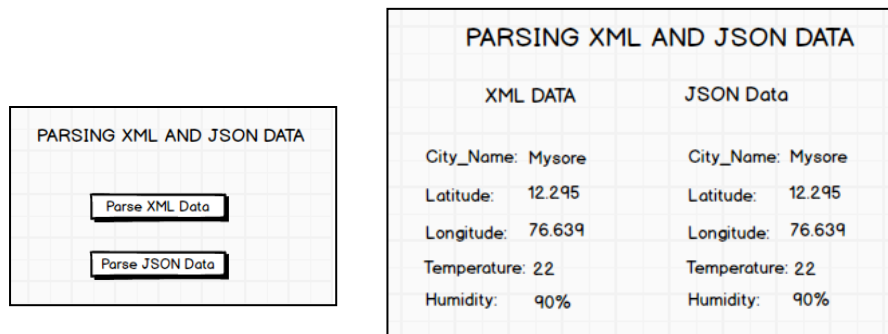
## Sample Output





## 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 input.xml file inside assets folder and paste the below Xml Data

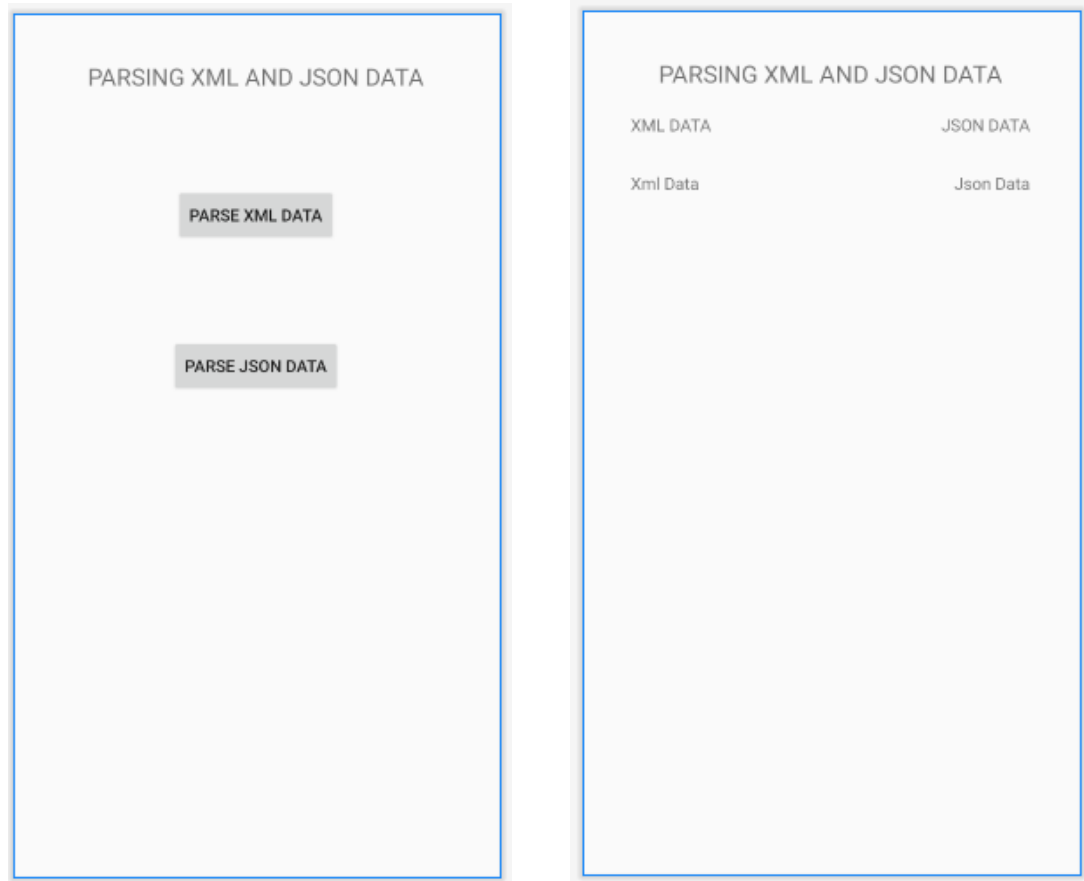
```
<?xml version="1.0"?>
<records>
<employee>
<city_name>Mysore</city_name>
<Latitude>12.295</Latitude>
<Longitude>76.639</Longitude>
<Temperature>22</Temperature>
<Humidity>90%</Humidity>
</employee>
</records>
```
7. Create input.json file inside assets folder and paste the below Json Data

```
{
"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

## Design



## 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=".MainActivity">

    <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>

```

## 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);
    }
}
}
}

```

### **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;
    @Override
    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");
                    lblXmlData.append("Latitude : " + getValue("Latitude", element2) + "\n");
                    lblXmlData.append("Longitude : " + getValue("Longitude", element2) + "\n");
                    lblXmlData.append("Temperature : " + getValue("Temperature", element2) + "\n");
                    lblXmlData.append("Humidity : " + getValue("Humidity", element2) + "\n");
                }
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
        return null;
    }

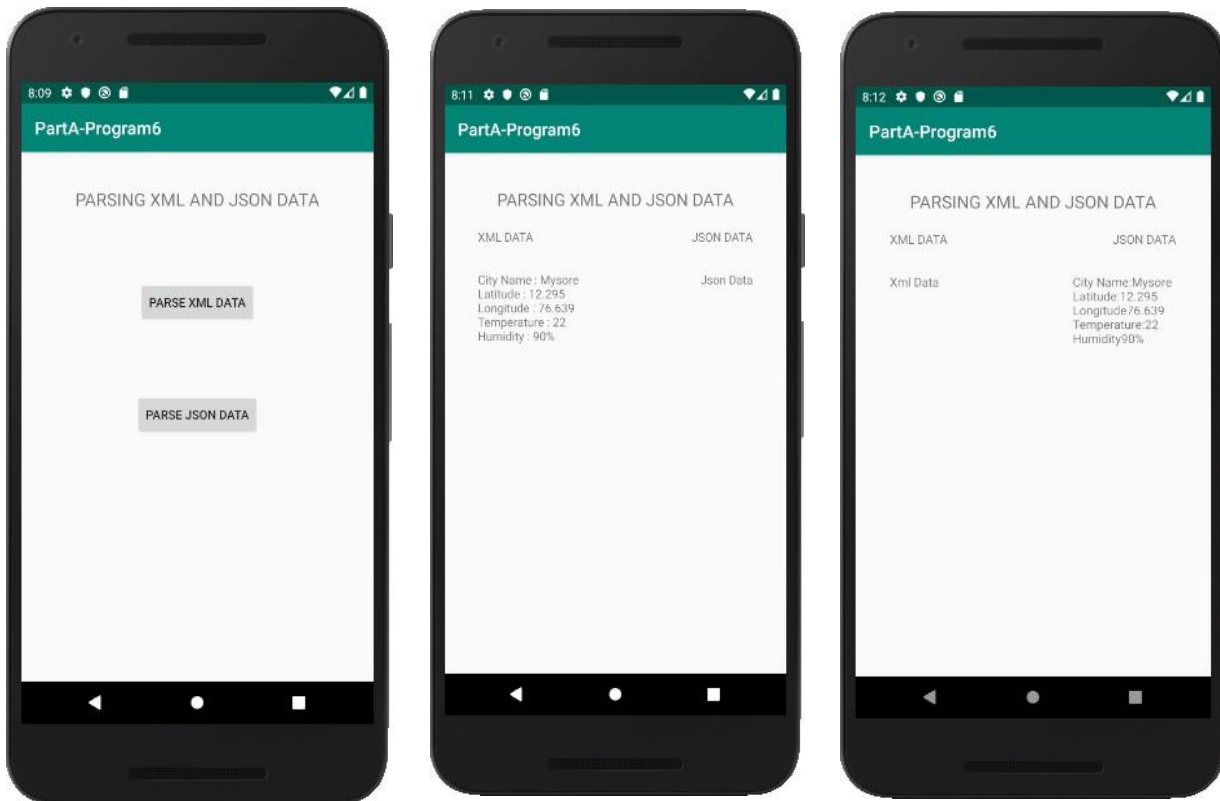
    public void parseJson()
    {
        try {
            InputStream inputStream=getAssets().open("input.json");
            byte[] data=new byte[inputStream.available()];
            inputStream.read(data);

            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();}
    }
}

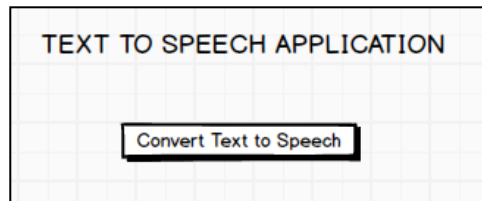
```

## Sample Output



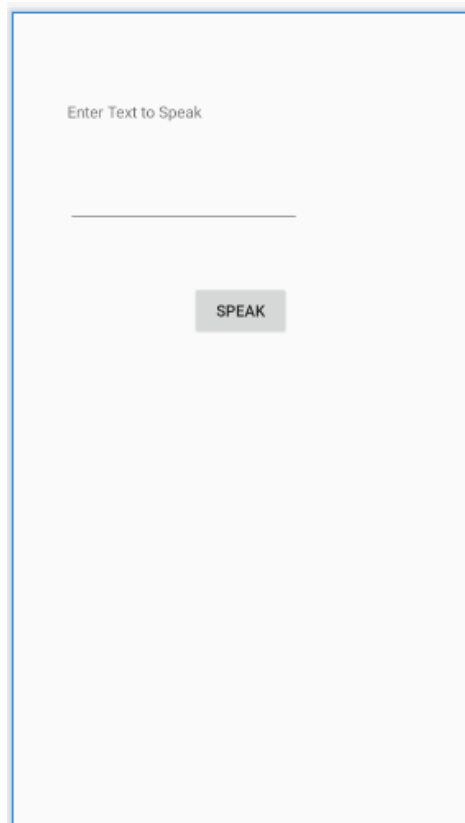
### Program 7

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.



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. Initialize TextToSpeech Engine and the Language to Speak using setLanguage() method
6. Use Speak() method to speak the text passed to it.

### Design





## **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>
```

## **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;

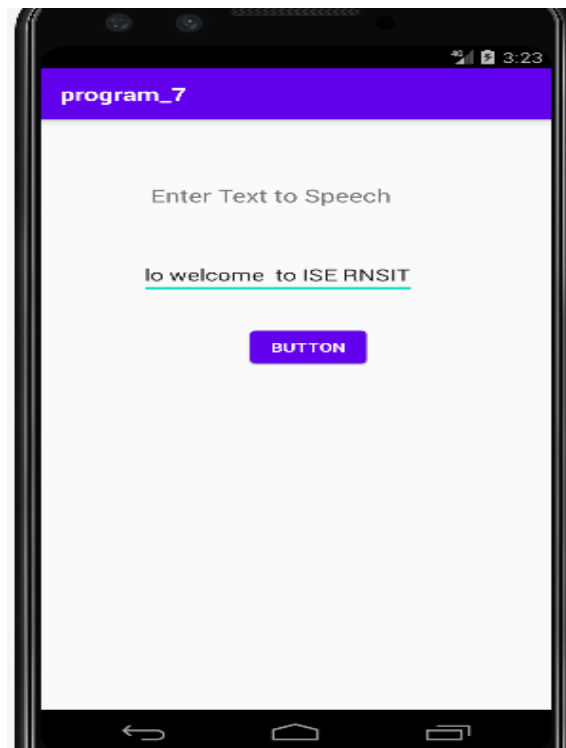
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        txtSpeak=(EditText)findViewById(R.id.editText);
        btnSpeak=(Button)findViewById(R.id.btn_speak);
        btnSpeak.setOnClickListener(this);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

    }
}

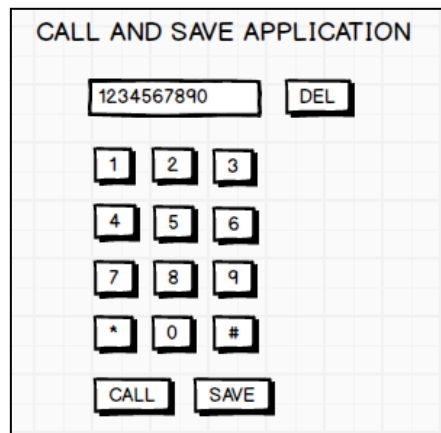
```

## 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.



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. Declare the use permission android.permission.CALL\_PHONE in the manifest file.
6. Use ACTION\_CALL intent name and pass the "tel:<phone-number>" as URI in intent data and start the call activity.
7. Use intent name and pass the "Telephone Number" and "unknown" as name as intent data call Contacts Save Activity.

## Design



### 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/txt_phonenumber" />
```

```
<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/txt_phonenumber" />
```

```
<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>

```

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

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

btnCall.setOnClickListener(new View.OnClickListener()
{
@Override

```

```

public void onClick(View v) {

String data = txtPhonenumber.getText().toString(); Intent
intent=new Intent(Intent.ACTION_DIAL);
intent.setData(Uri.parse("tel:"+ data)); startActivity(intent);
}});

}

}

```

### **AndriodManifest.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

