MOBILE APPLICATION DEVELOPMENT
LABORATORY MANUAL

VI Semester
CourseCode:18AI643

Table of Content

Vision of the Institute

Become a premier institution imparting quality education in engineering and management to meet the changing needs of society

Mission of the Institute

- Create environment conducive for continuous learning through quality teaching andlearning processes supported by moderninfrastructure
- Promote Research and Innovation through collaboration withindustries
- Inculcate ethical values and environmental consciousness through holistic educationprograms

Vision of the Department

“Be a premier department in the field of Artificial Intelligency & Machine Learning to meet the technological challenges of the society”

Mission of the Department

MD 1 To provide state of the art infrastructure facilities

MD 2 To provide exposure to the latest tools in the area of computer hardware and software

MD 3 To strive for academic excellence through research in Artificial Intelligency & Machine Learning with creative teaching-learning pedagogy

MD 4 To establish Industry Institute Interaction and make students ready for the Industrial environment

MD 5 To transform students into entrepreneurial, technically competent, socially responsible and ethical computer science professional

Program Educational Objectives (PEOs)

After the course completion, AI & ML graduates will be able to:

PEO1: Succeed in engineering/management positions with professional ethics.

PEO2: Engage in improving professional knowledge through certificate/post- graduate programs in engineering or management.

PEO3: Establish themselves as entrepreneurs and contribute to the Society.

Program Specific Outcomes (PSOs)

PSO1: Design, implement and test System Software and Application Software to meet the desired needs.

PSO2: Develop solutions in the area of Communication Networks, Database Systems and Computing Systems.

Program Outcomes (POs)

Engineering Graduates will be able to:

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

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

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

4. **Conduct investiAMCECions of complex problems**: Use research-based knowledge and research methodsincluding design of experiments, analysis and interpretation of data, and synthesis of the information to provide validconclusions.

5. **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 thelimitations.

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

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

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

9. **Individualandteamwork**:Functioneffectivelyasanindividual,andasamemberorleaderindiverse teams, and in multidisciplinarysettings.

10. **Communication**: Communicate effectively on complex engineering activities with the engineering communityandwithsocietyatlarge,suchas,beingabletocomprehendandwriteeffectivereportsand design documentation, make effective presentations, and give and receive clearinstructions.

11. **Projectmanagementandfinance**:Demonstrateknowledgeandunderstandingoftheengineeringand management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinaryenvironments.

12. **Life-longlearning**:Recognizetheneedfor,andhavethepreparationandabilitytoengagein independent and life-long learning in the broadest context of technologicalchange.

Course Details

CourseName:Mobile Application Development

CourseCode:18AI643

Course Prerequisite: Core Java

Course Objectives

Upon completion of this course, students are expected to:

1. Learn and acquire the art of AndroidProgramming.
2. Configure Android studio to run theapplications.
3. Understand and implement Android's User interfacefunctions.
4. Create, modify and query on SQLitedatabase.
5. Inspect different methods of sharing data usingservices.

Course Outcomes

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

18AI643.1	Create, test and debug Android application by setting up Android development environment.
18AI643.2	Implement adaptive, responsive user interfaces that work across a wide range of devices.
18AI643.3	Demonstrate methods in storing, sharing and retrieving data in Android applications.
18AI643.4	Infer the role of permissions and security for Android applications.

SYLLABUS

MOBILE APPLICATION DEVELOPMENT

SubjectCode:18AI643	**IA Marks:40**
No. of Practical Hrs. /Week:0:0:2	**Exam Marks: 60**
Total No. of Practical Hrs:3Hours/Week	**Exam Hours: 03**
No. of Credits:02	

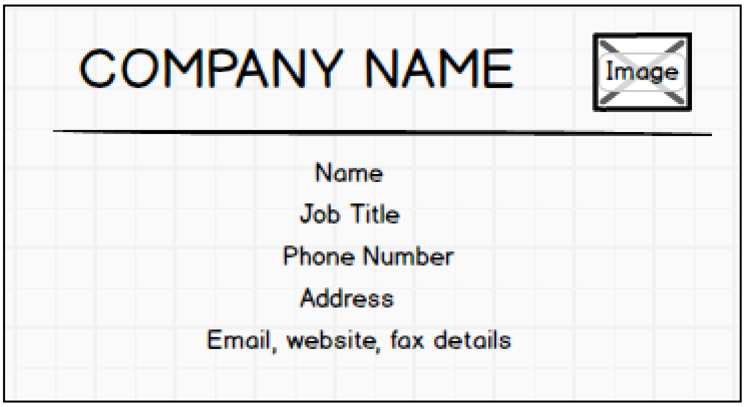
Descriptions (if any):

1. The installation procedure of the Android Studio/Java software must be demonstrated and carried out ingroups.
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 PartB).

PART A

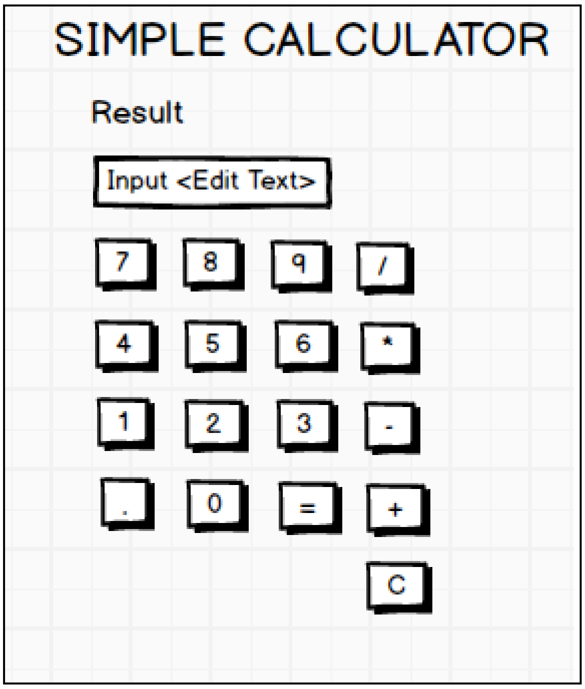
Program 1

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.

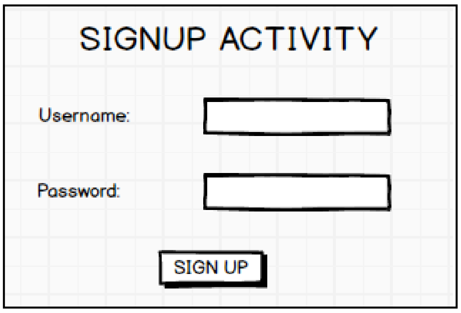


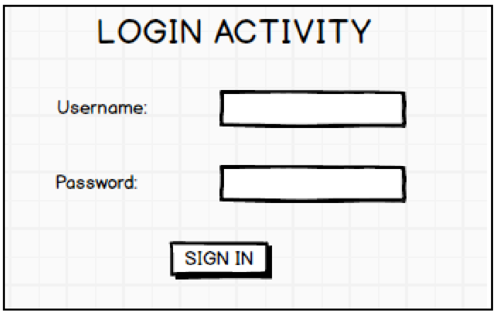
Program 3

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

- Password should contain uppercase and lowercaseletters.
- Password should contain letters andnumbers.
- Password should contain specialcharacters.
- Minimum length of the password (the default value is8).

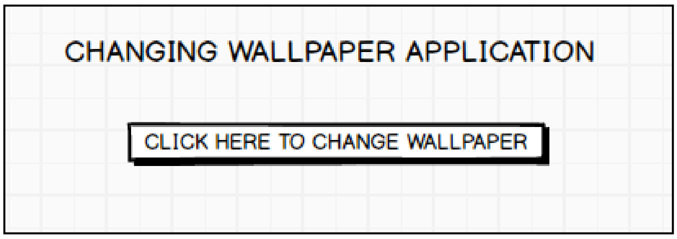
On successful **SIGN UP** proceed to the next Login activity. Here the user should **SIGN IN** using the Username and Password created during signup activity. If the Username and Password are matched then naviAMCECe to the next activity which displays a message saying “Successful Login” or else display a toast message saying “Login Failed”. The user is given only two attempts and afterthatdisplayatoastmessagesaying“FailedLoginAttempts”anddisabletheSIGNINbutton. UseBundle to transfer information from one activity toanother.





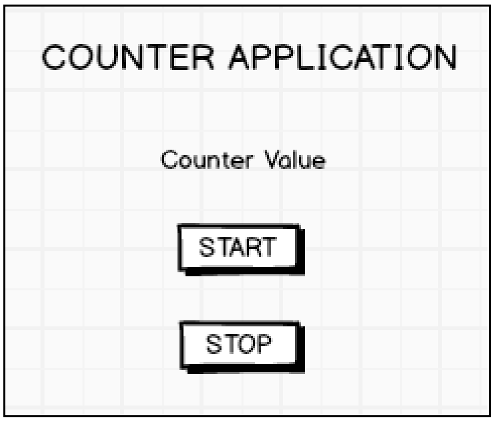
Program 4

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


CHANGING WALLPAPER APPLICATION
CLICK HERE TO CHANGE WALLPAPER

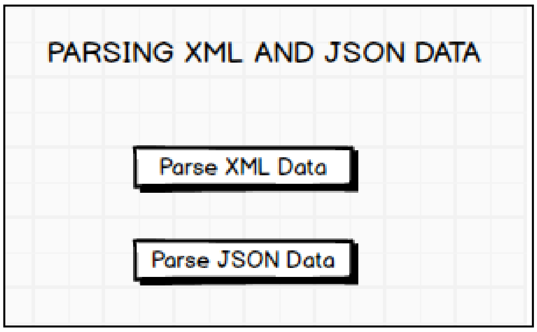
Program 5

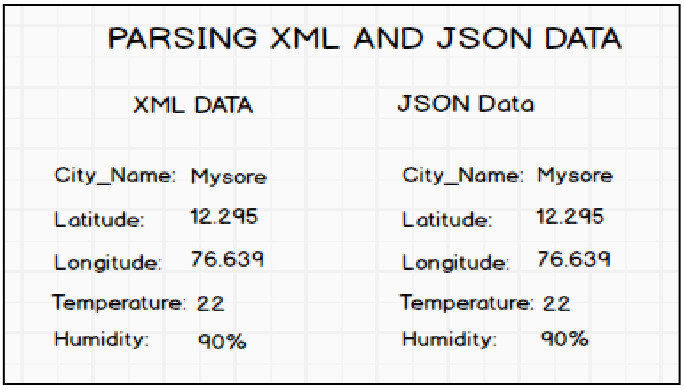
Write a program to create an activity with two buttons START and STOP. On Pressing of the START button, the activity must start the counter by displaying the numbers from One and the counter must keep on counting until the STOP button is pressed. Display the counter value in a TextView control.



Program 6

Create two files of XML and JSON type with values for City_Name, Latitude, Longitude, Temperature,andHumidity.Developanapplicationtocreateanactivitywithtwobuttonstoparse the XML and JSON files which when clicked should display the data in their respective layouts side by side.

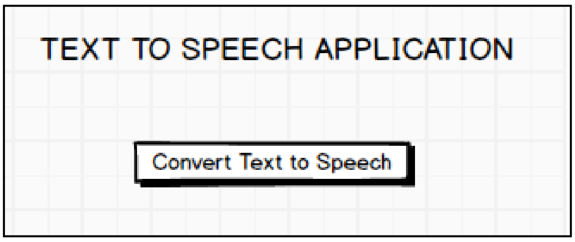



PARSING XML AND JSON DATA

XML DATA		JSON Data	
City_Name:	Mysore	City_Name:	Mysore
Latitude:	12.295	Latitude:	12.295
Longitude:	76.639	Longitude:	76.639
Temperature:	22	Temperature:	22
Humidity:	90%	Humidity:	90%

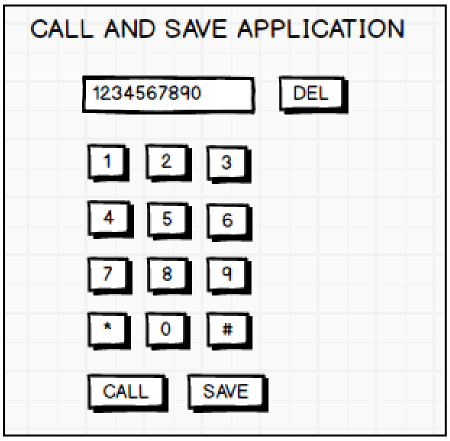
Program 7

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

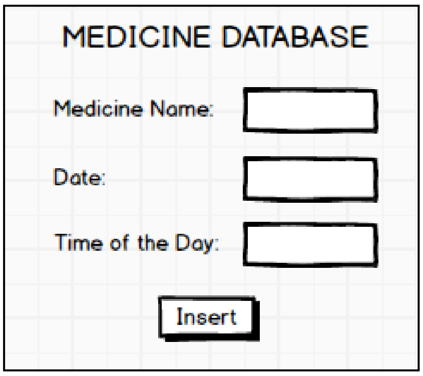
Create an activity like a phone dialer with CALL and SAVE buttons. On pressing the CALL button, it must call the phone number and on pressing the SAVE button it must save the number to the phone contacts.



PART B

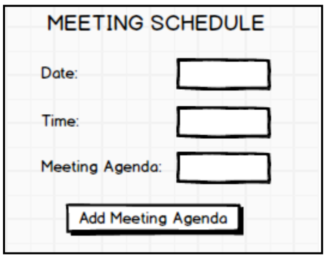
Program 1

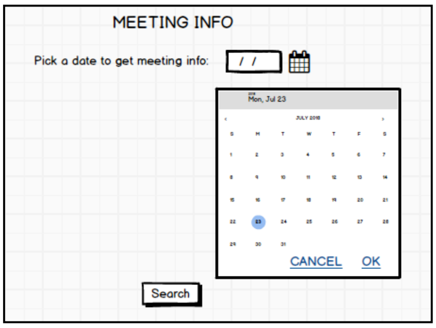
Write a program to enter Medicine Name, Date and Time of the Day as input from the user and store it in the SQLite database. Input for Time of the Day should be either Morning or Afternoon or Evening or Night. Trigger an alarm based on the Date and Time of the Day and display the Medicine Name.



Program 2

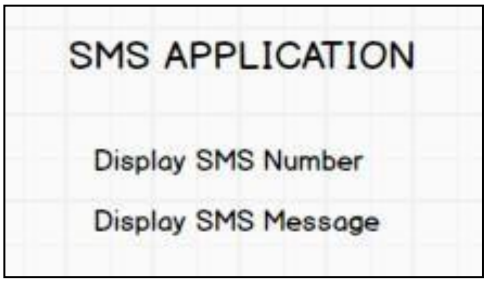
Develop a content provider application with an activity called “Meeting Schedule” which takes Date,TimeandMeetingAgendaasinputfromtheuserandstorethisinformationintotheSQLite database. Create another application with an activity called “Meeting Info” having DatePicker control, which on the selection of a date should display the Meeting Agenda information for that particular date, else it should display a toast message saying “No Meeting on thisDate”.





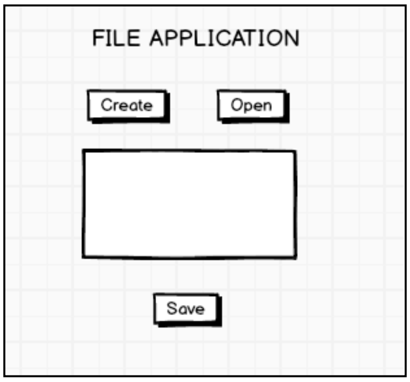
Program 3

Create an application to receive an incoming SMS which is notified to the user. On clicking this SMS notification, the message content and the number should be displayed on the screen. Use appropriate emulator control to send the SMS message to your application.



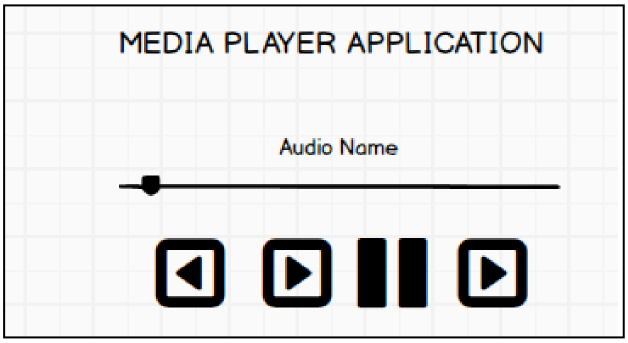
Program 4

Write a program to create an activity having a Text box, and also Save, Open and Create buttons. The user has to write some text in the Text box. On pressing the Create button the text should be saved as a text file in MkSDcard. On subsequent changes to the text, the Save button should be pressed to store the latest content to the same file. On pressing the Open button, it should display the contents from the previously stored files in the Text box. If the user tries to save the contents in the Textbox to a file without creating it, then a toast message has to be displayed saying "FirstCreate a File".



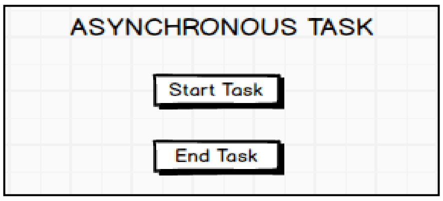
Program 5

Create an application to demonstrate a basic media player that allows the user to Forward, Backward, Play and Pause an audio. Also, make use of the indicator in the seek bar to move the audio forward or backward as required.



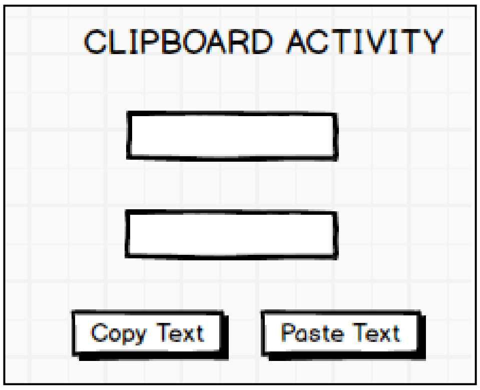
Program 6

Develop an application to demonstrate the use of Asynchronous tasks in android. The asynchronoustaskshouldimplementthefunctionalityofasimplemovingbanner.Onpressingthe **StartTask**button,thebannermessageshouldscrollfromrighttoleft.Onpressingthe**StopTask** button, the banner message should stop. Let the banner message be "Demonstration of AsynchronousTask".



Program 7

Develop an application that makes use of the clipboard framework for copying and pasting ofthe text. The activity consists of two EditText controls and two Buttons to trigger the copy and paste functionality.



Program 8

Create an AIDL service that calculates Car Loan EMI. The formula to calculate EMI is

$$\mathbf{E = P * (r(1+r)^n)/((1+r)^n-1)}$$

where

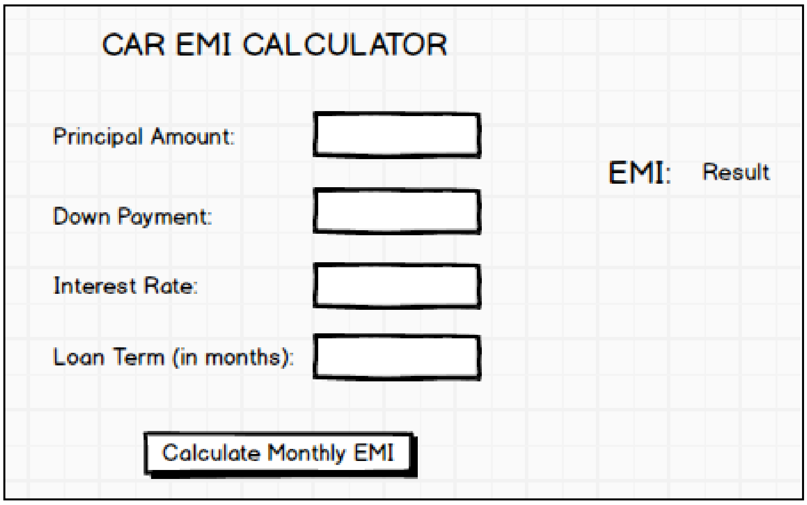
E = The EMI payable on the car loan amount

P = The Car loan Principal Amount

r = The interest rate value computed on a monthly basis

n = The loan tenure in the form of months

Thedownpaymentamounthastobedeductedfromtheprincipalamountpaidtowardsbuyingthe Car. Develop an application that makes use of this AIDL service to calculate the EMI. This applicationshouldhavefourEditTexttoreadthePrincipalAmount,DownPayment,InterestRate, Loan Term (in months) and a button named as “Calculate Monthly EMI”.On click of this button, theresultshouldbeshowninaTextView.Also,calculatetheEMIbyvaryingtheLoanTermand Interest Ratevalues.



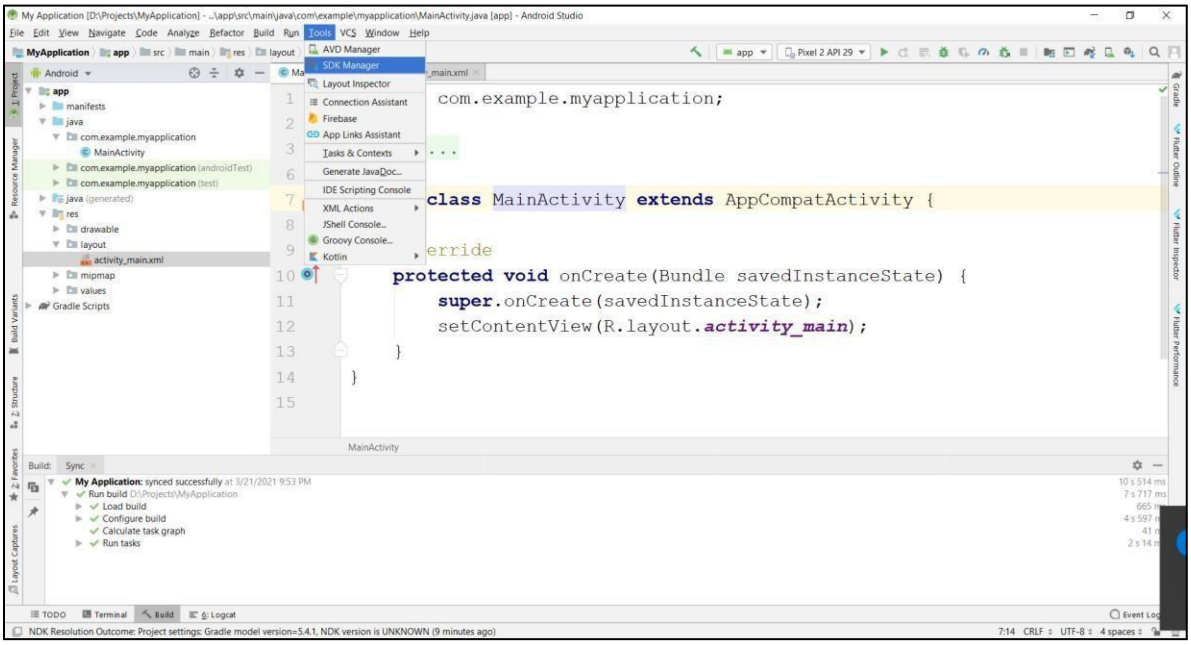
1.Android StudioTutorials

Install Android Studio andPackages:

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

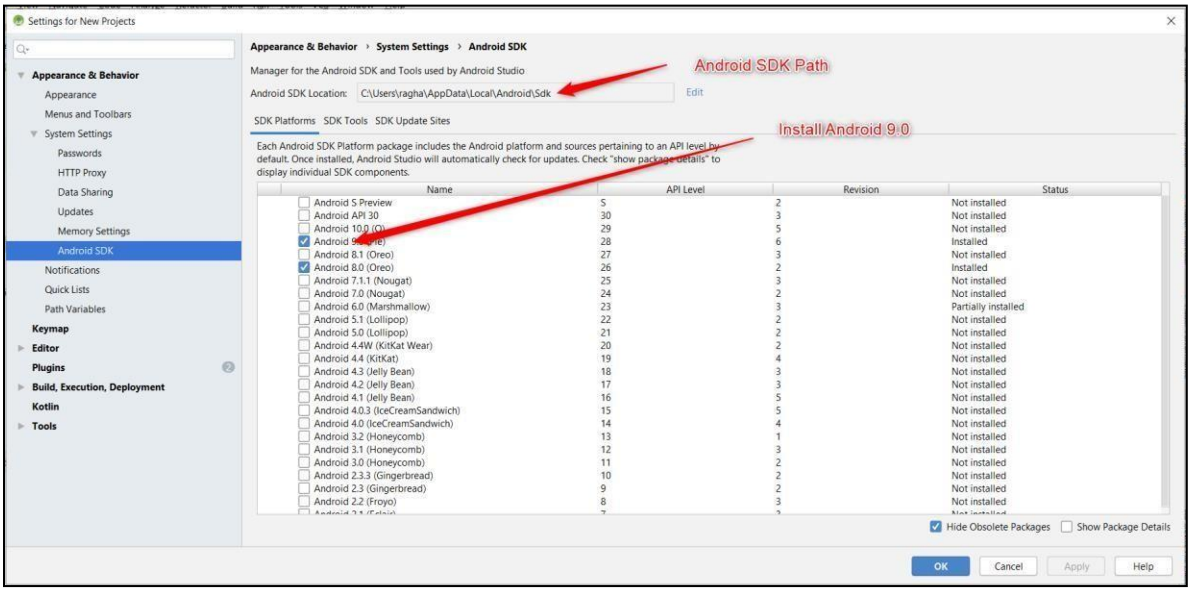
Configure Android SDKpackages:

Go to Tools ➐ SDK Manager



```
com.example.myapplication;

...

class MainActivity extends AppCompatActivity {

erride
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

```
Build: Sync
My Application: synced successfully at 3/21/2021 9:53 PM      10 s 514 ms
  Run build D:\Projects\MyApplication                          7 s 717 ms
    Load build                                                 665
    Configure build                                            4 s 597
    Calculate task graph                                       41
    Run tasks                                                  2 s 14
NDK Resolution Outcome: Project settings: Gradle model version=5.4.1, NDK version is UNKNOWN (9 minutes ago)
7:14 CRLF UTF-8 4 spaces
```



Appearance & Behavior › System Settings › Android SDK

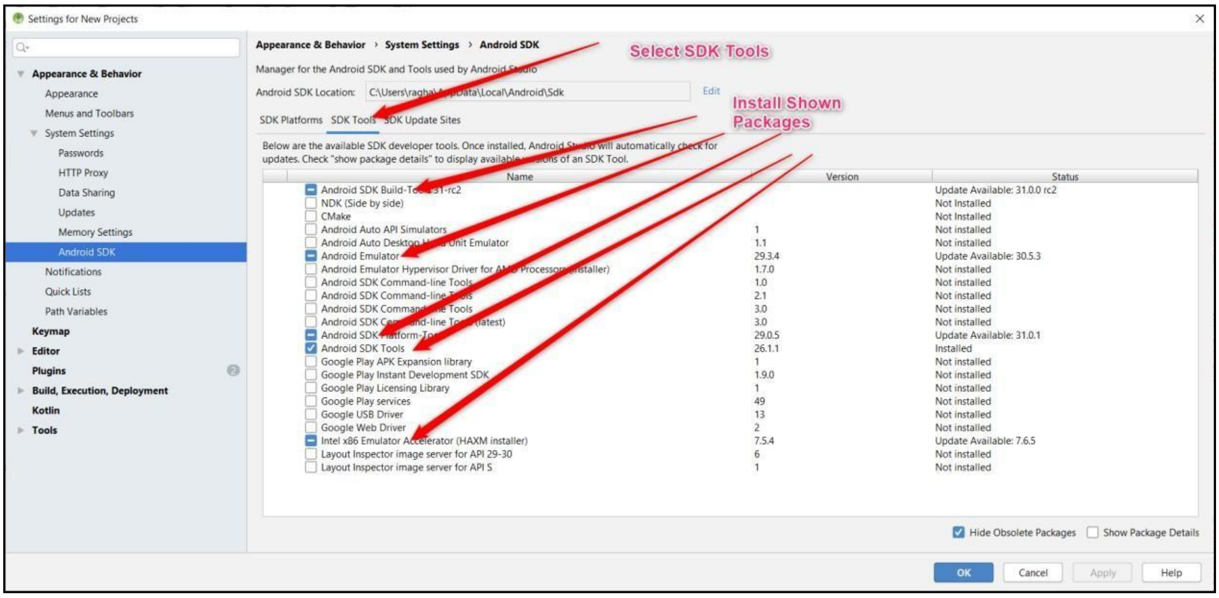
Manager for the Android SDK and Tools used by Android Studio

Android SDK Location: C:\Users\ragha\AppData\Local\Android\Sdk Edit

Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, Android Studio will automatically check for updates. Check "show package details" to display individual SDK components.

Name	API Level	Revision	Status
□ Android S Preview	S	2	Not installed
□ Android API 30	30	3	Not installed
□ Android 10.0 (Q)	29	5	Not installed
☑ Android 9.0 (Pie)	28	6	Installed
□ Android 8.1 (Oreo)	27	3	Not installed
☑ Android 8.0 (Oreo)	26	2	Installed
□ Android 7.1.1 (Nougat)	25	3	Not installed
□ Android 7.0 (Nougat)	24	2	Not installed
□ Android 6.0 (Marshmallow)	23	3	Partially installed
□ Android 5.1 (Lollipop)	22	2	Not installed
□ Android 5.0 (Lollipop)	21	2	Not installed
□ Android 4.4W (KitKat Wear)	20	2	Not installed
□ Android 4.4 (KitKat)	19	4	Not installed
□ Android 4.3 (Jelly Bean)	18	3	Not installed
□ Android 4.2 (Jelly Bean)	17	3	Not installed
□ Android 4.1 (Jelly Bean)	16	5	Not installed
□ Android 4.0.3 (IceCreamSandwich)	15	5	Not installed
□ Android 4.0 (IceCreamSandwich)	14	4	Not installed
□ Android 3.2 (Honeycomb)	13	1	Not installed
□ Android 3.1 (Honeycomb)	12	3	Not installed
□ Android 3.0 (Honeycomb)	11	2	Not installed
□ Android 2.3.3 (Gingerbread)	10	2	Not installed
□ Android 2.3 (Gingerbread)	9	2	Not installed
□ Android 2.2 (Froyo)	8	3	Not installed
□ Android 2.1 (Eclair)	7	2	Not installed

☑ Hide Obsolete Packages □ Show Package Details



Manager for the Android SDK and Tools used by Android Studio

Android SDK Location: C:\Users\ragha\AppData\Local\Android\Sdk

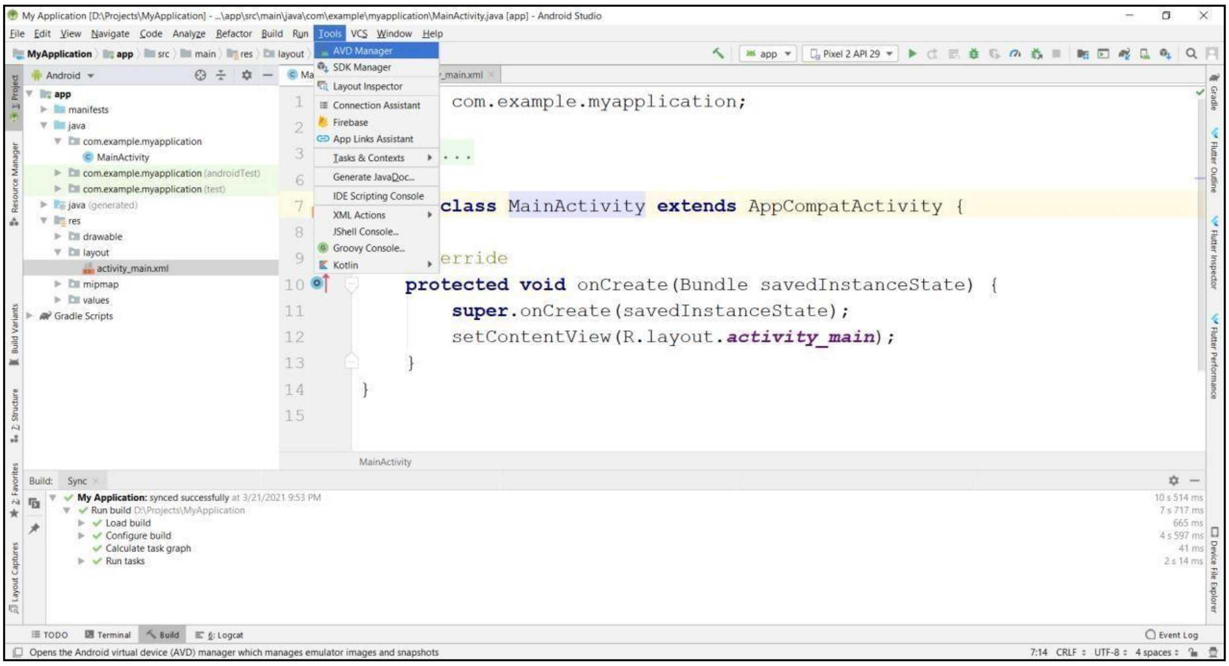
Below are the available SDK developer tools. Once installed, Android Studio will automatically check for updates. Check "show package details" to display available versions of an SDK Tool.

Name	Version	Status
▣ Android SDK Build-Tools 31-rc2		Update Available: 31.0.0 rc2
□ NDK (Side by side)		Not Installed
□ CMake		Not Installed
□ Android Auto API Simulators	1	Not installed
□ Android Auto Desktop Head Unit Emulator	1.1	Not installed
▣ Android Emulator	29.3.4	Update Available: 30.5.3
□ Android Emulator Hypervisor Driver for AMD Processors (installer)	1.7.0	Not installed
□ Android SDK Command-line Tools	1.0	Not installed
□ Android SDK Command-line Tools	2.1	Not installed
□ Android SDK Command-line Tools	3.0	Not installed
□ Android SDK Command-line Tools (latest)	3.0	Not installed
▣ Android SDK Platform-Tools	29.0.5	Update Available: 31.0.1
☑ Android SDK Tools	26.1.1	Installed
□ Google Play APK Expansion library	1	Not installed
□ Google Play Instant Development SDK	1.9.0	Not installed
□ Google Play Licensing Library	1	Not installed
□ Google Play services	49	Not installed
□ Google USB Driver	13	Not installed
□ Google Web Driver	2	Not installed
▣ Intel x86 Emulator Accelerator (HAXM installer)	7.5.4	Update Available: 7.6.5
□ Layout Inspector image server for API 29-30	6	Not installed
□ Layout Inspector image server for API S	1	Not installed

☑ Hide Obsolete Packages □ Show Package Details

CreatingEmulator

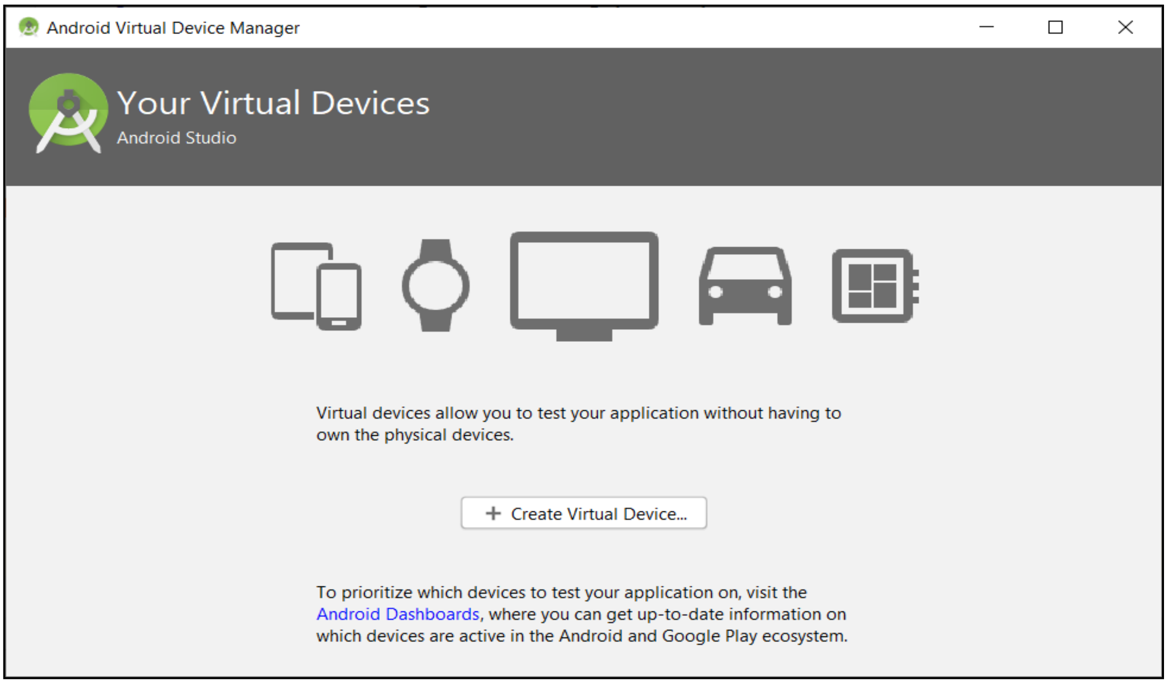
GotoTools ❼ SelectAVDManager



```
     com.example.myapplication;

     ...

    class MainActivity extends AppCompatActivity {

    erride
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

```
My Application: synced successfully at 3/21/2021 9:53 PM    10 s 514 ms
  Run build D:\Projects\MyApplication                        7 s 717 ms
    Load build                                                   665 ms
    Configure build                                          4 s 597 ms
    Calculate task graph                                          41 ms
    Run tasks                                                 2 s 14 ms
```

Opens the Android virtual device (AVD) manager which manages emulator images and snapshots



Android Virtual Device Manager

Your Virtual Devices

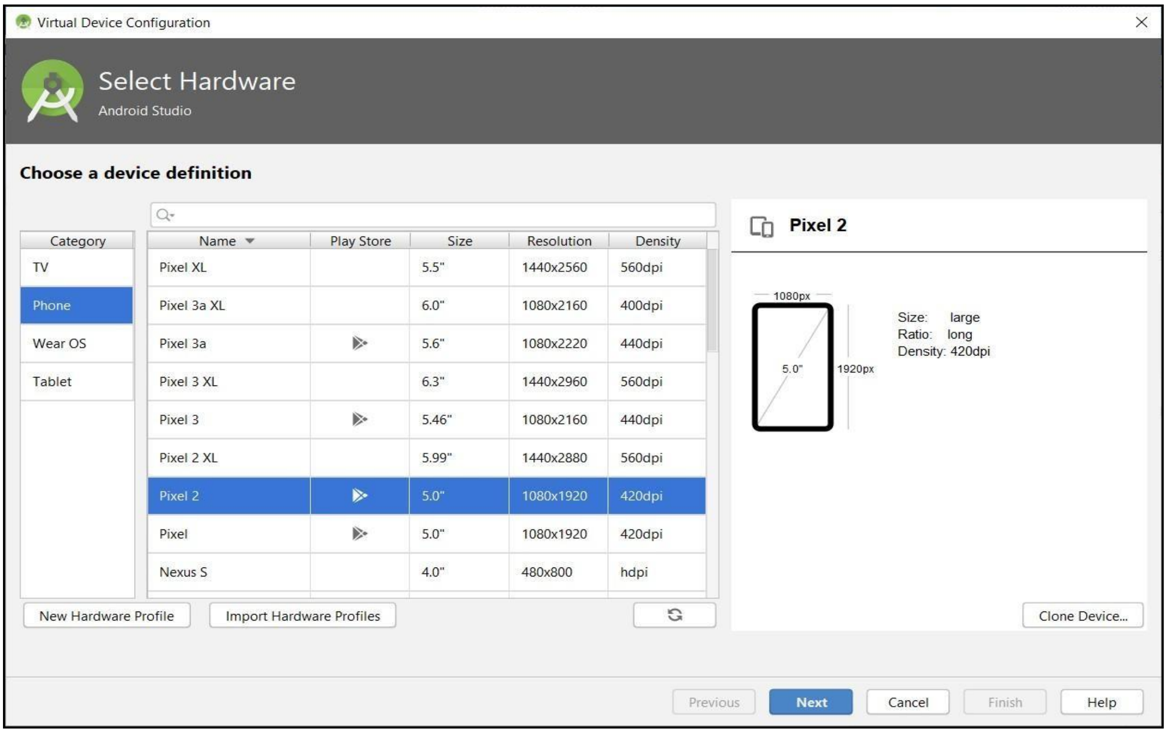
Android Studio

Virtual devices allow you to test your application without having to own the physical devices.

+ Create Virtual Device...

To prioritize which devices to test your application on, visit the Android Dashboards, where you can get up-to-date information on which devices are active in the Android and Google Play ecosystem.

Select CreateVirtual Device ➐ Select Phone ➐ Pixel 2 ➐ PressNext



Virtual Device Configuration

Select Hardware

Android Studio

Choose a device definition

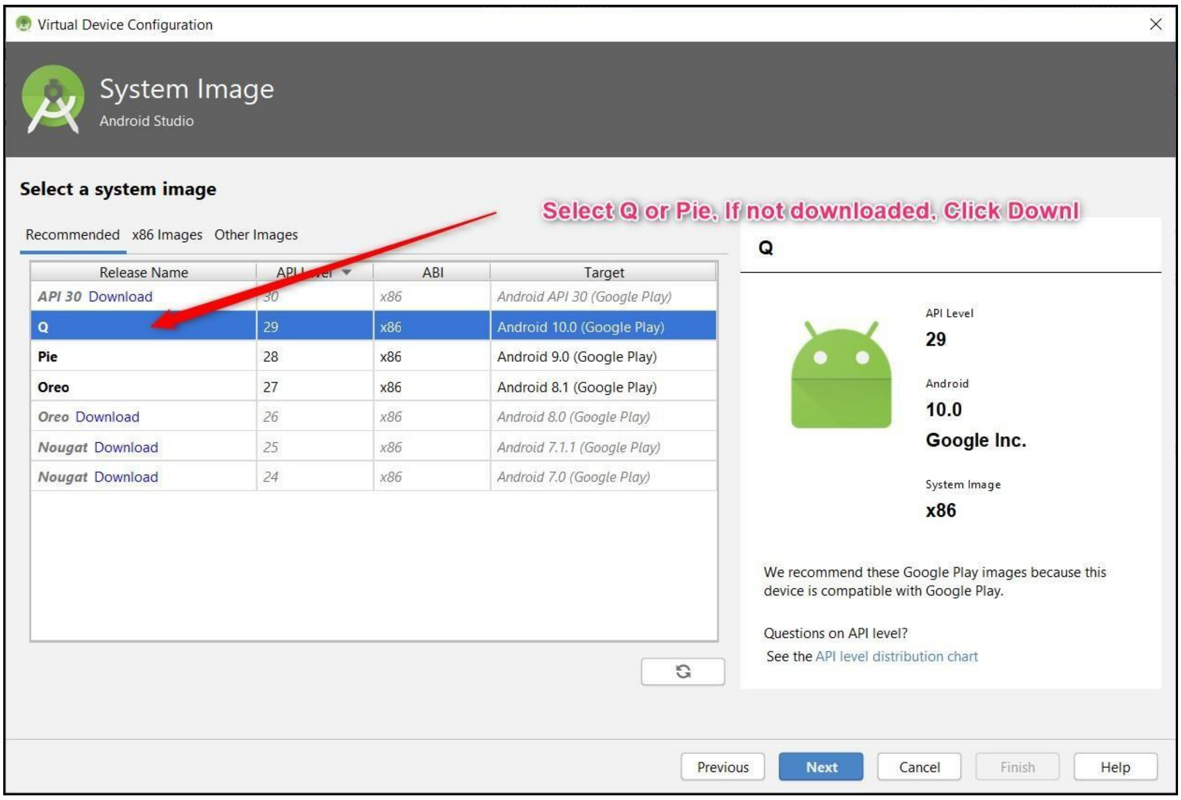
Category
TV
Phone
Wear OS
Tablet

Name	Play Store	Size	Resolution	Density
Pixel XL		5.5"	1440x2560	560dpi
Pixel 3a XL		6.0"	1080x2160	400dpi
Pixel 3a	▶	5.6"	1080x2220	440dpi
Pixel 3 XL		6.3"	1440x2960	560dpi
Pixel 3	▶	5.46"	1080x2160	440dpi
Pixel 2 XL		5.99"	1440x2880	560dpi
Pixel 2	▶	5.0"	1080x1920	420dpi
Pixel	▶	5.0"	1080x1920	420dpi
Nexus S		4.0"	480x800	hdpi

Pixel 2

Size: large
Ratio: long
Density: 420dpi

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



Virtual Device Configuration

System Image

Android Studio

Select a system image

Select Q or Pie, If not downloaded, Click Downl

Release Name	API Level	ABI	Target
API 30 Download	30	x86	Android API 30 (Google Play)
Q	29	x86	Android 10.0 (Google Play)
Pie	28	x86	Android 9.0 (Google Play)
Oreo	27	x86	Android 8.1 (Google Play)
Oreo Download	26	x86	Android 8.0 (Google Play)
Nougat Download	25	x86	Android 7.1.1 (Google Play)
Nougat Download	24	x86	Android 7.0 (Google Play)

Q

API Level
29

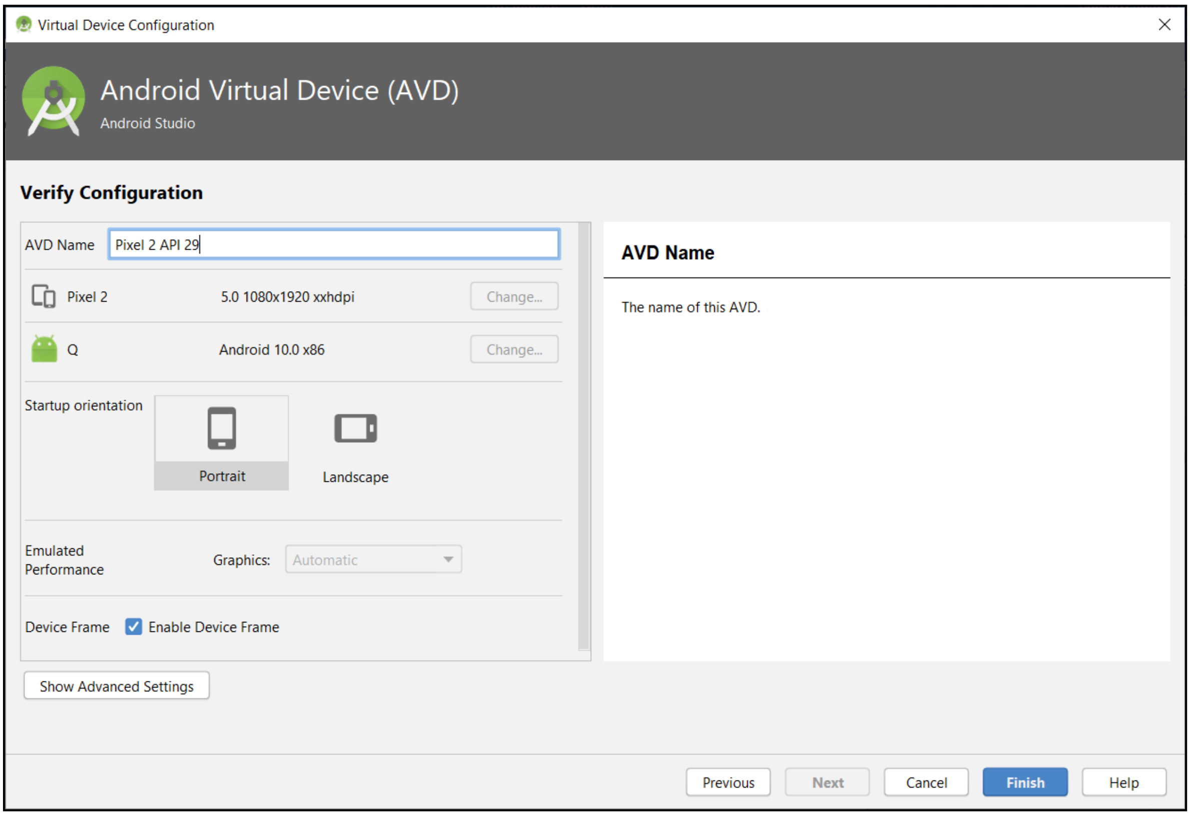
Android
10.0
Google Inc.

System Image
x86

We recommend these Google Play images because this device is compatible with Google Play.

Questions on API level?
See the API level distribution chart

Enter AVD Name and PressFinish.



Virtual Device Configuration

Android Virtual Device (AVD)

Android Studio

Verify Configuration

AVD Name: Pixel 2 API 29

Pixel 2 — 5.0 1080x1920 xxhdpi — Change...

Q — Android 10.0 x86 — Change...

Startup orientation: Portrait, Landscape

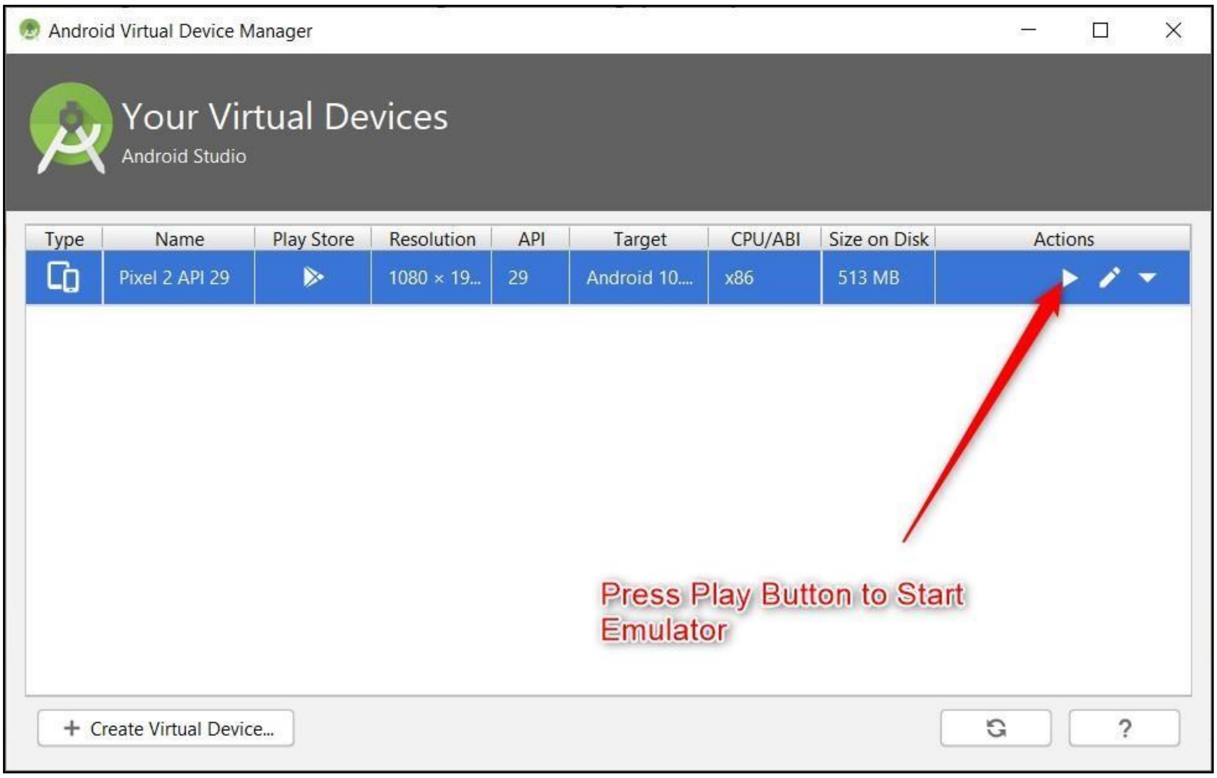
Emulated Performance — Graphics: Automatic

Device Frame ☑ Enable Device Frame

AVD Name

The name of this AVD.

Press Play Button to StartEmulator

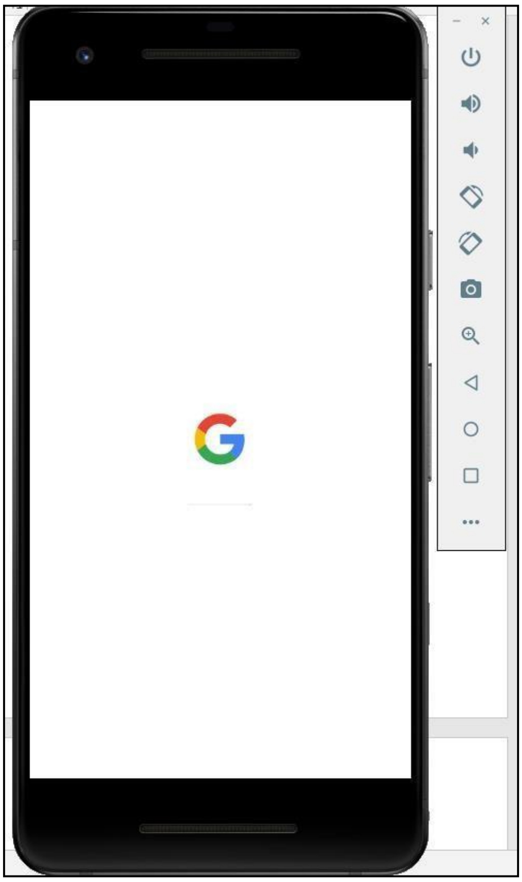


Your Virtual Devices

Android Studio

Type	Name	Play Store	Resolution	API	Target	CPU/ABI	Size on Disk	Actions
	Pixel 2 API 29		1080 × 19...	29	Android 10....	x86	513 MB	▶ ✎ ▾

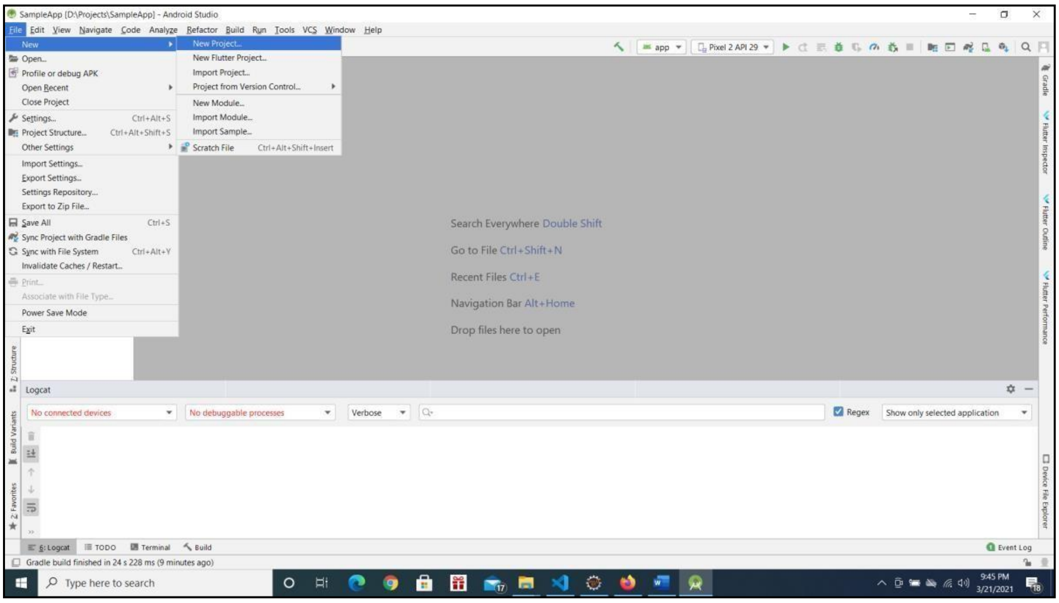
Press Play Button to Start Emulator



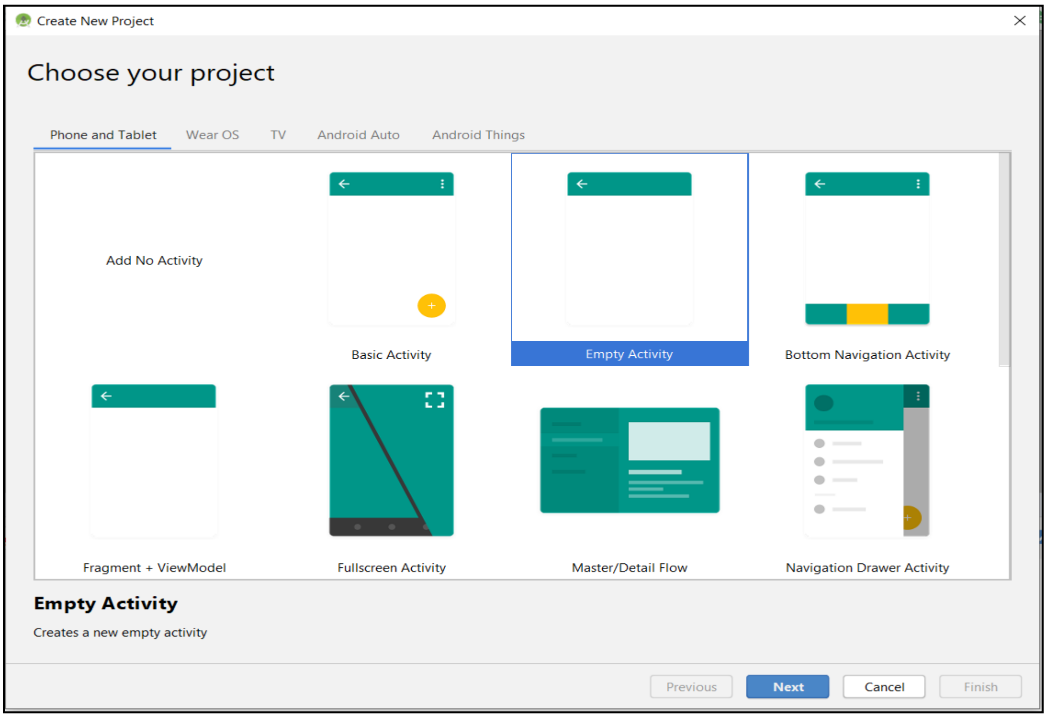
Creating a New Project inAndroid

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 ➔ NewProject



Choose Phone and Tablet ➔ Empty Activity ➔ Press Next



In Configure your Project Screen, Enter below details and Press Finish Button. EnterNameoftheApplication ❼ ThiswillbeapplicationnamethiswillbevisiblewithHomeScreenIcon.

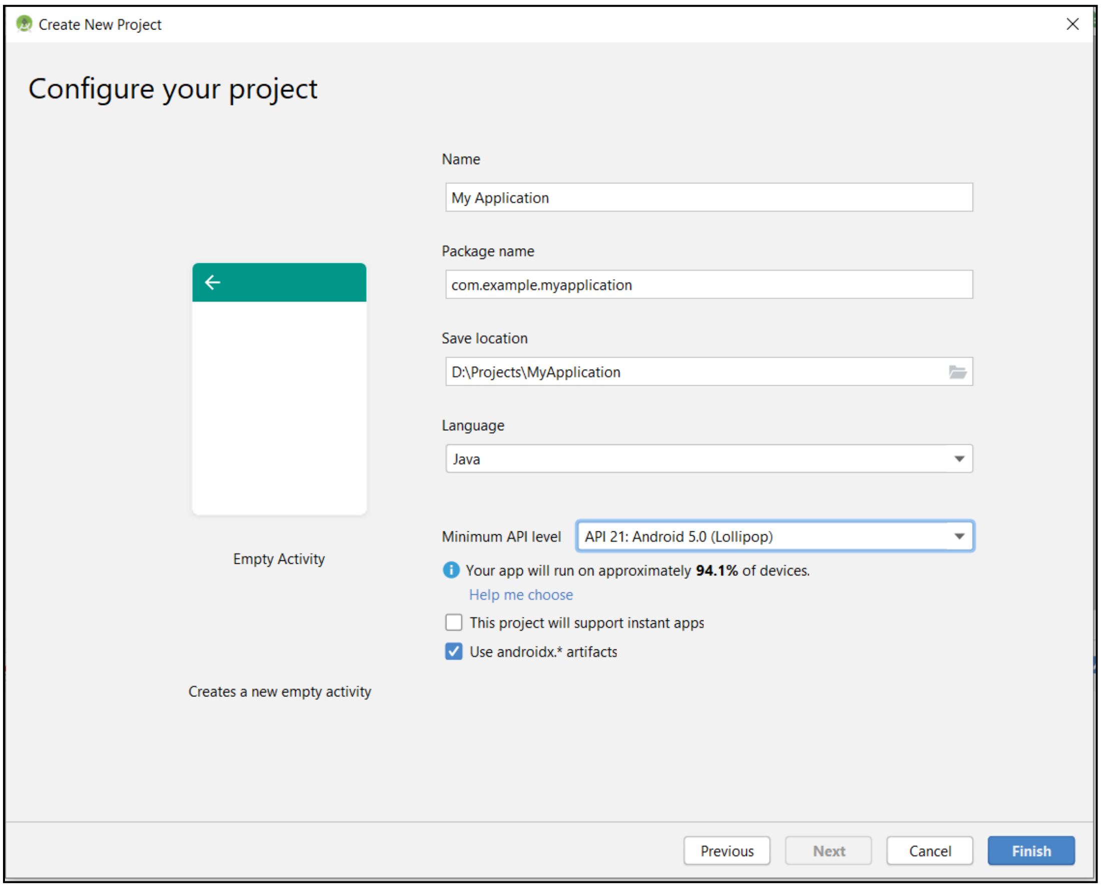
PackageName❼Enterpackagenameatleasttwoidentifier(Eg:com.example).BestPractice is3ormoreidentifier(Eg:com.example.firstapp).

Save Location ❼ Location where to save the

Project Language ❼ Choose Java

Minimum API Level ❼ Android 5.0

Select Checkbox Use androidx.artifacts folder as below screenshot.



Create New Project

Configure your project

Name: My Application

Package name: com.example.myapplication

Save location: D:\Projects\MyApplication

Language: Java

Minimum API level: API 21: Android 5.0 (Lollipop)

Your app will run on approximately **94.1%** of devices.

Help me choose

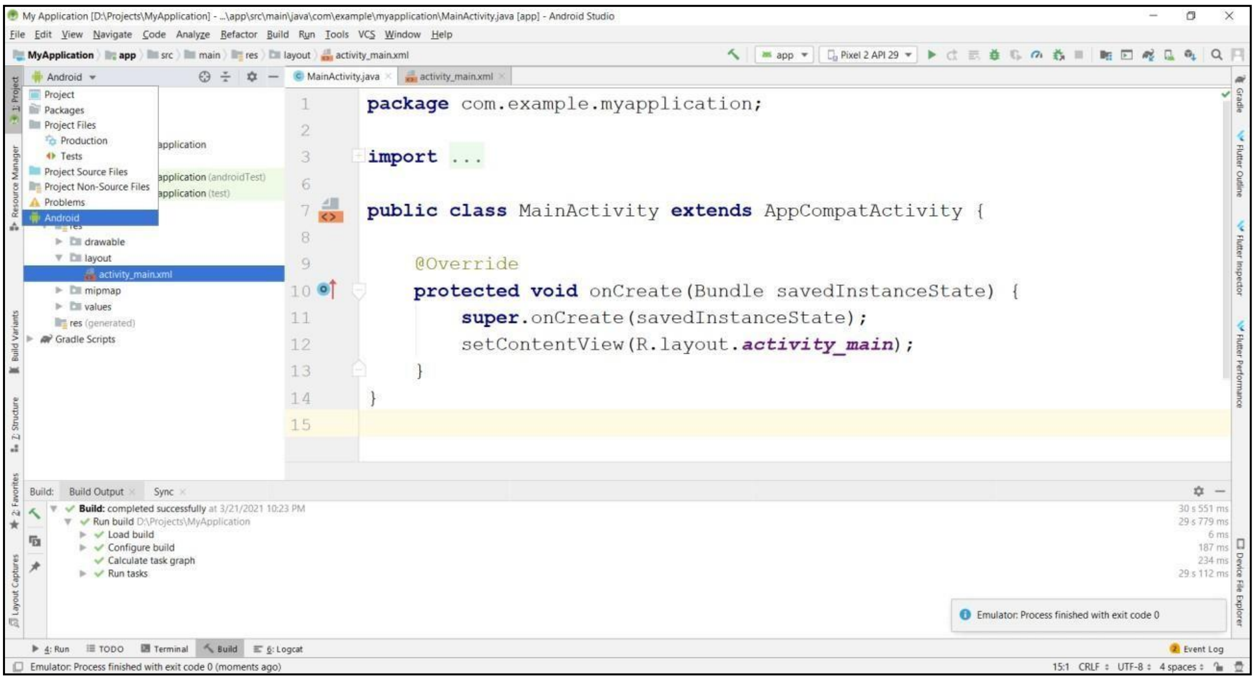
□ This project will support instant apps

☑ Use androidx.* artifacts

Empty Activity

Creates a new empty activity

Android ProjectStructure:



My Application [D:\Projects\MyApplication] - ...\app\src\main\java\com\example\myapplication\MainActivity.java [app] - Android Studio

```java
package com.example.myapplication;

import ...

public class MainActivity extends AppCompatActivity {

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

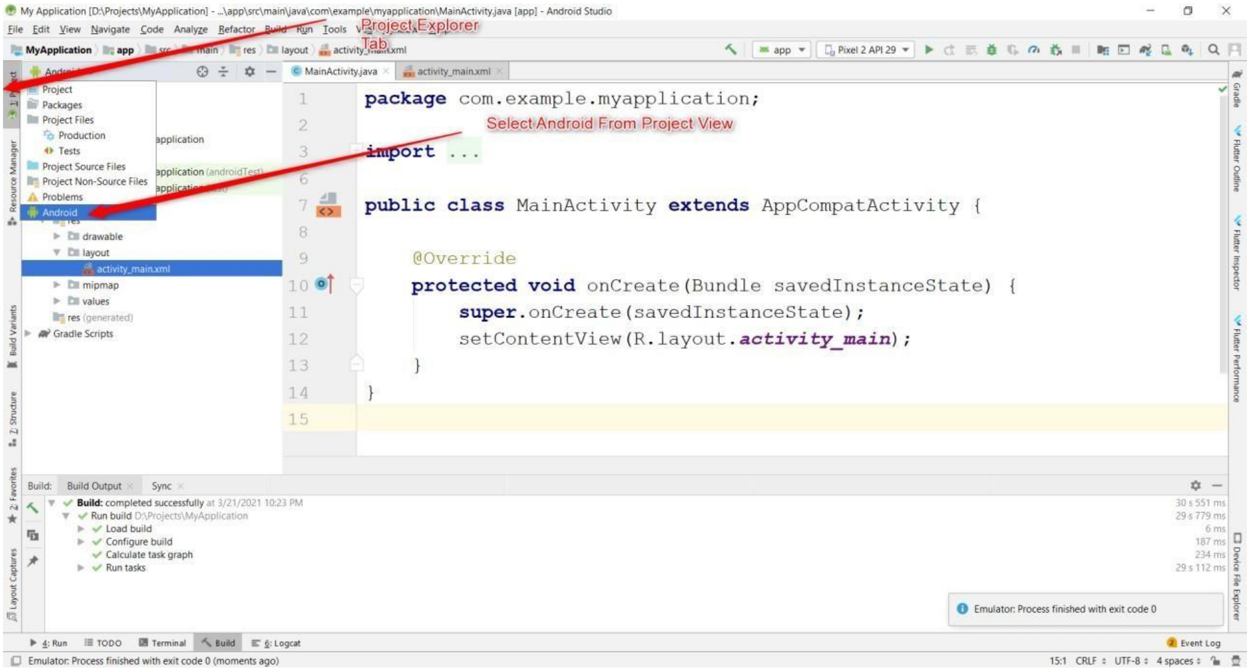
```
Build: completed successfully at 3/21/2021 10:23 PM      30 s 551 ms
  Run build D:\Projects\MyApplication                    29 s 779 ms
    Load build                                                 6 ms
    Configure build                                          187 ms
    Calculate task graph                                     234 ms
    Run tasks                                           29 s 112 ms
```

Emulator: Process finished with exit code 0

Emulator: Process finished with exit code 0 (moments ago)

15:1 CRLF UTF-8 4 spaces

Select Project Explorer and Select Android from Project View



My Application [D:\Projects\MyApplication] - ...\app\src\main\java\com\example\myapplication\MainActivity.java [app] - Android Studio

Project Explorer Tab

Select Android From Project View

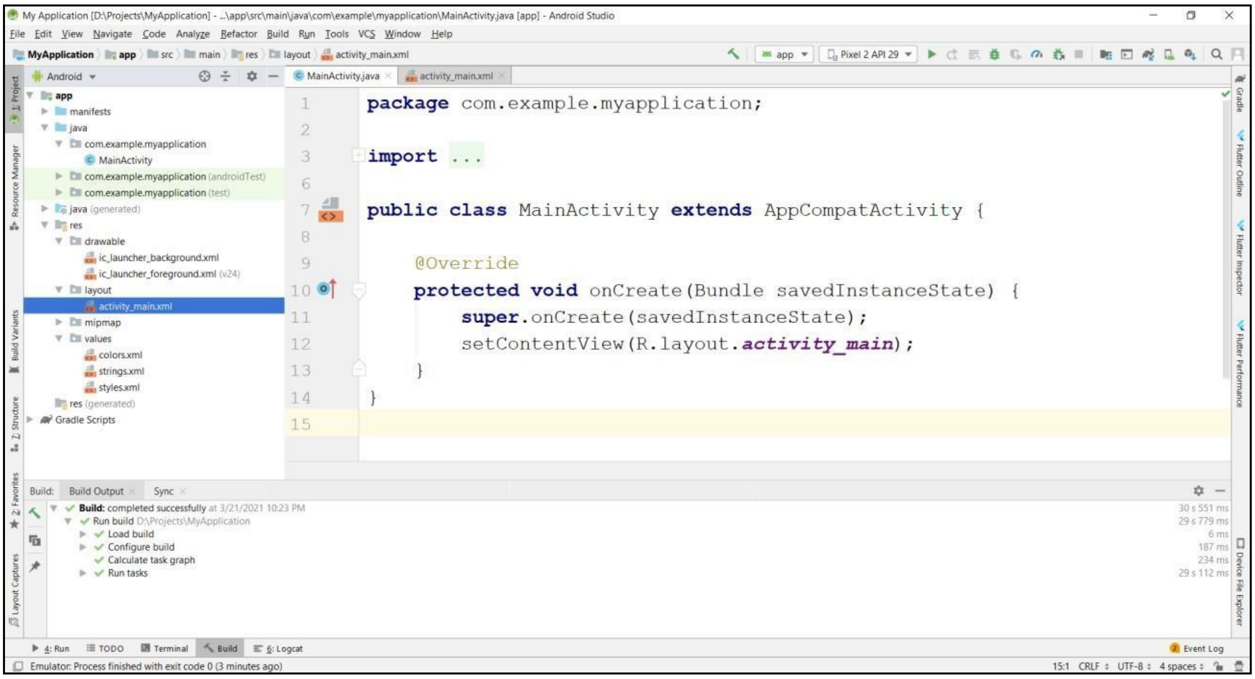
```java
package com.example.myapplication;

import ...

public class MainActivity extends AppCompatActivity {

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

```
Build: completed successfully at 3/21/2021 10:23 PM      30 s 551 ms
  Run build D:\Projects\MyApplication                    29 s 779 ms
    Load build                                                 6 ms
    Configure build                                          187 ms
    Calculate task graph                                     234 ms
    Run tasks                                           29 s 112 ms
```

Emulator: Process finished with exit code 0

Emulator: Process finished with exit code 0 (moments ago)

Basic View:


My Application [D:\Projects\MyApplication] - ...\app\src\main\java\com\example\myapplication\MainActivity.java [app] - Android Studio
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
MyApplication app src main res layout activity_main.xml
app
Pixel 2 API 29
Android
MainActivity.java
activity_main.xml
app
manifests
java
com.example.myapplication
MainActivity
com.example.myapplication (androidTest)
com.example.myapplication (test)
java (generated)
res
drawable
ic_launcher_background.xml
ic_launcher_foreground.xml (v24)
layout
activity_main.xml
mipmap
values
colors.xml
strings.xml
styles.xml
res (generated)
Gradle Scripts
Build: Build Output Sync
4: Run
TODO
Terminal
Build
6: Logcat
Event Log
15:1 CRLF UTF-8 4 spaces

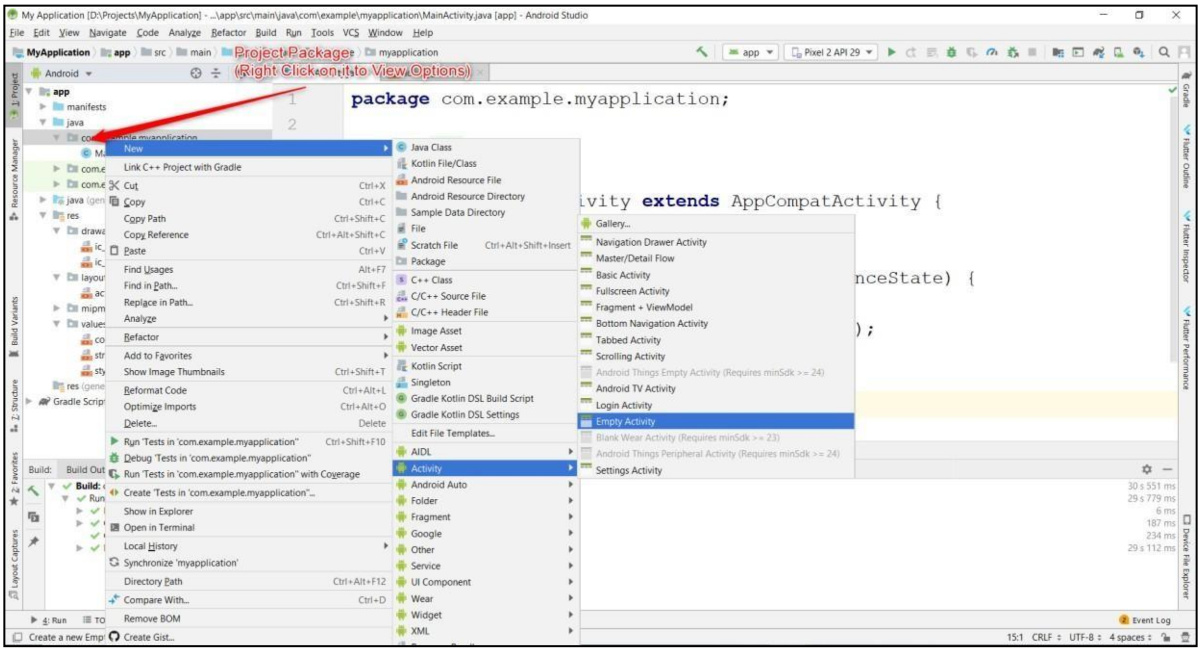
```
package com.example.myapplication;

import ...

public class MainActivity extends AppCompatActivity {

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

```
Build: completed successfully at 3/21/2021 10:23 PM    30 s 551 ms
  Run build D:\Projects\MyApplication                  29 s 779 ms
    Load build                                         6 ms
    Configure build                                    187 ms
    Calculate task graph                               234 ms
    Run tasks                                          29 s 112 ms
```

Emulator: Process finished with exit code 0 (3 minutes ago)

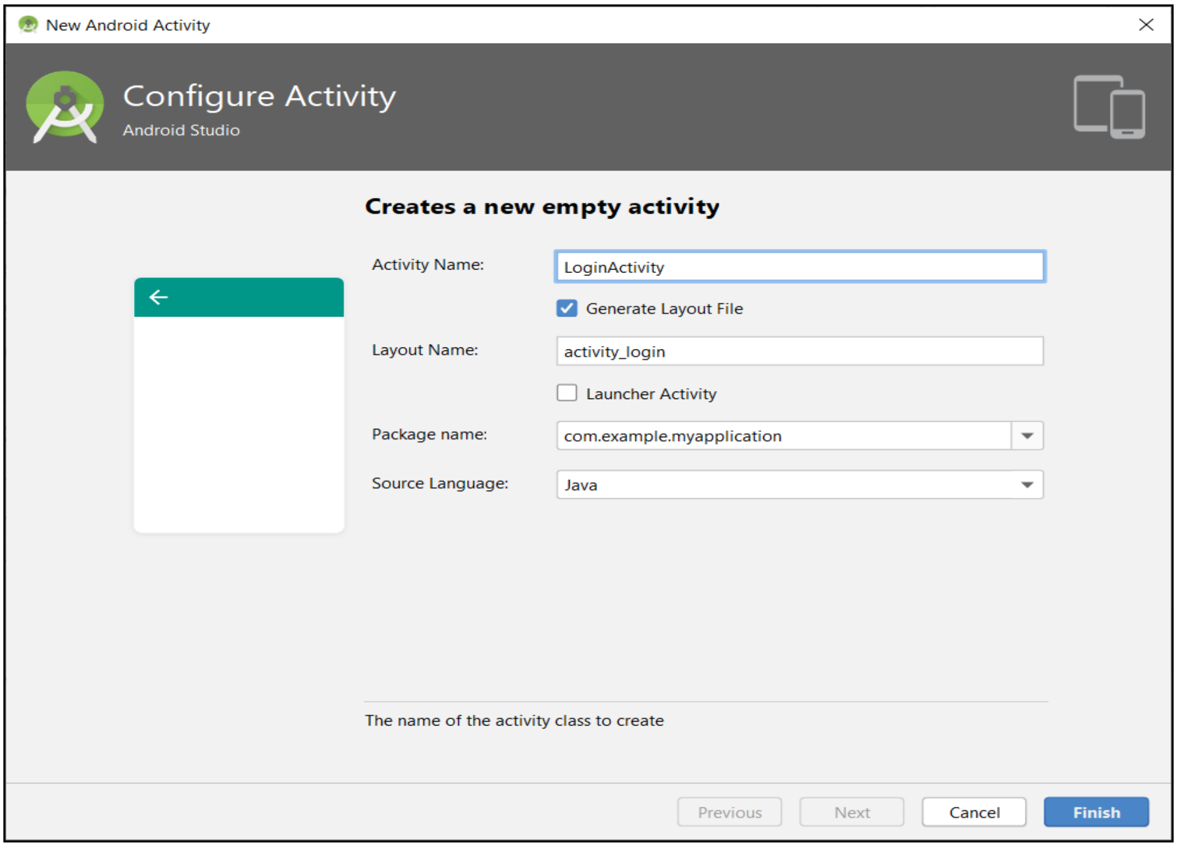
1. Importing an Existing Project in AndroidStudio
2. Creating an Activity inAndroid

   Right Click on Package ➐ New ➐ Activity ➐ Empty Activity


My Application [D:\Projects\MyApplication] - ...\app\src\main\java\com\example\myapplication\MainActivity.java [app] - Android Studio
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
MyApplication app src main
Project Package
(Right Click on it to View Options)
myapplication
Android
New
Link C++ Project with Gradle
Cut Ctrl+X
Copy Ctrl+C
Copy Path Ctrl+Shift+C
Copy Reference Ctrl+Alt+Shift+C
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Analyze
Refactor
Add to Favorites
Show Image Thumbnails Ctrl+Shift+T
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Run 'Tests in 'com.example.myapplication'' Ctrl+Shift+F10
Debug 'Tests in 'com.example.myapplication''
Run 'Tests in 'com.example.myapplication'' with Coverage
Create 'Tests in 'com.example.myapplication''...
Show in Explorer
Open in Terminal
Local History
Synchronize 'myapplication'
Directory Path Ctrl+Alt+F12
Compare With... Ctrl+D
Remove BOM
Create Gist...
Java Class
Kotlin File/Class
Android Resource File
Android Resource Directory
Sample Data Directory
File
Scratch File Ctrl+Alt+Shift+Insert
Package
C++ Class
C/C++ Source File
C/C++ Header File
Image Asset
Vector Asset
Kotlin Script
Singleton
Gradle Kotlin DSL Build Script
Gradle Kotlin DSL Settings
Edit File Templates...
AIDL
Activity
Android Auto
Folder
Fragment
Google
Other
Service
UI Component
Wear
Widget
XML
Gallery...
Navigation Drawer Activity
Master/Detail Flow
Basic Activity
Fullscreen Activity
Fragment + ViewModel
Bottom Navigation Activity
Tabbed Activity
Scrolling Activity
Android Things Empty Activity (Requires minSdk >= 24)
Android TV Activity
Login Activity
Empty Activity
Blank Wear Activity (Requires minSdk >= 23)
Android Things Peripheral Activity (Requires minSdk >= 24)
Settings Activity
Create a new Empty Activity
Event Log
15:1 CRLF UTF-8 4 spaces

```
package com.example.myapplication;

...ivity extends AppCompatActivity {

...nceState) {

...);
```

Enter Activity Name and Press Finish



Creates a new empty activity

Activity Name: LoginActivity

☑ Generate Layout File

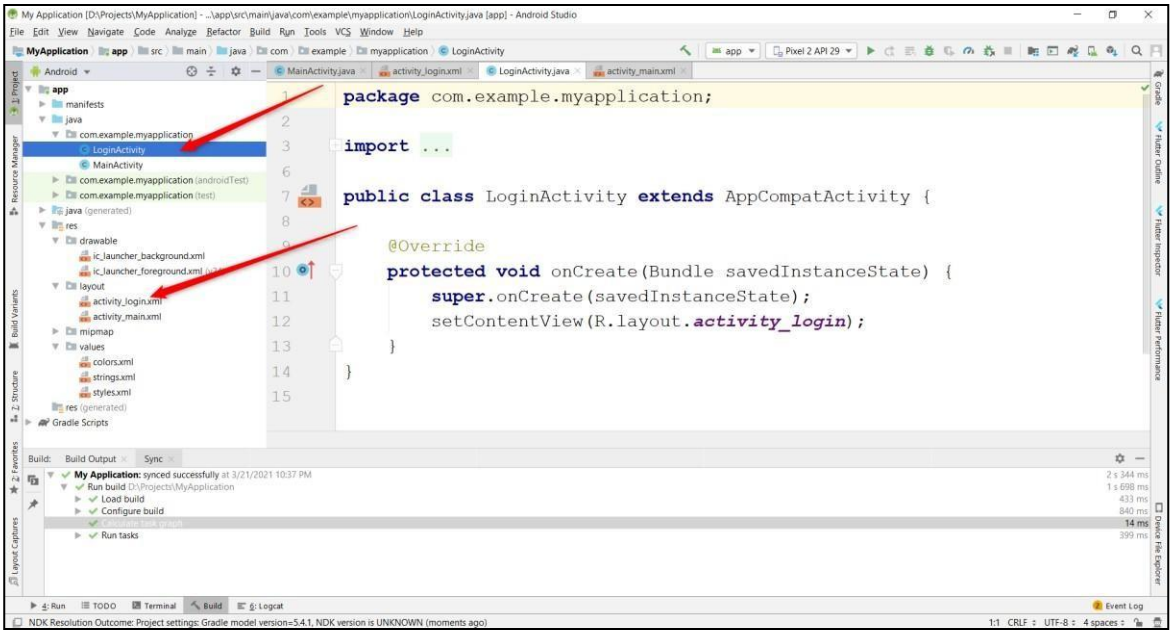
Layout Name: activity_login

□ Launcher Activity

Package name: com.example.myapplication

Source Language: Java

The name of the activity class to create



My Application [D:\Projects\MyApplication] - ...\app\src\main\java\com\example\myapplication\LoginActivity.java [app] - Android Studio
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
MyApplication › app › src › main › java › com › example › myapplication › LoginActivity
Android
app
manifests
java
com.example.myapplication
LoginActivity
MainActivity
com.example.myapplication (androidTest)
com.example.myapplication (test)
java (generated)
res
drawable
ic_launcher_background.xml
ic_launcher_foreground.xml
layout
activity_login.xml
activity_main.xml
mipmap
values
colors.xml
strings.xml
styles.xml
res (generated)
Gradle Scripts
MainActivity.java
activity_login.xml
LoginActivity.java
activity_main.xml
4: Run
TODO
Terminal
Build
6: Logcat
Event Log

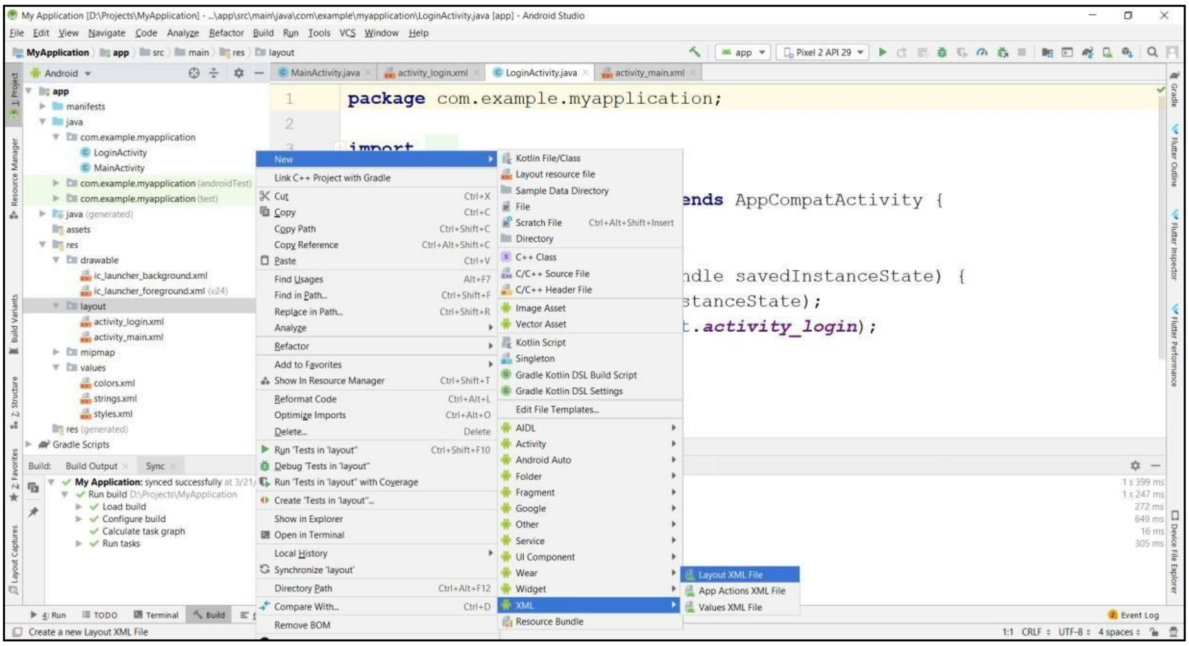
```java
package com.example.myapplication;

import ...

public class LoginActivity extends AppCompatActivity {

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

```
Build: Build Output  Sync
My Application: synced successfully at 3/21/2021 10:37 PM    2 s 344 ms
  Run build D:\Projects\MyApplication                        1 s 698 ms
    Load build                                                   433 ms
    Configure build                                              840 ms
    Calculate task graph                                          14 ms
    Run tasks                                                    399 ms
```

NDK Resolution Outcome: Project settings: Gradle model version=5.4.1, NDK version is UNKNOWN (moments ago)

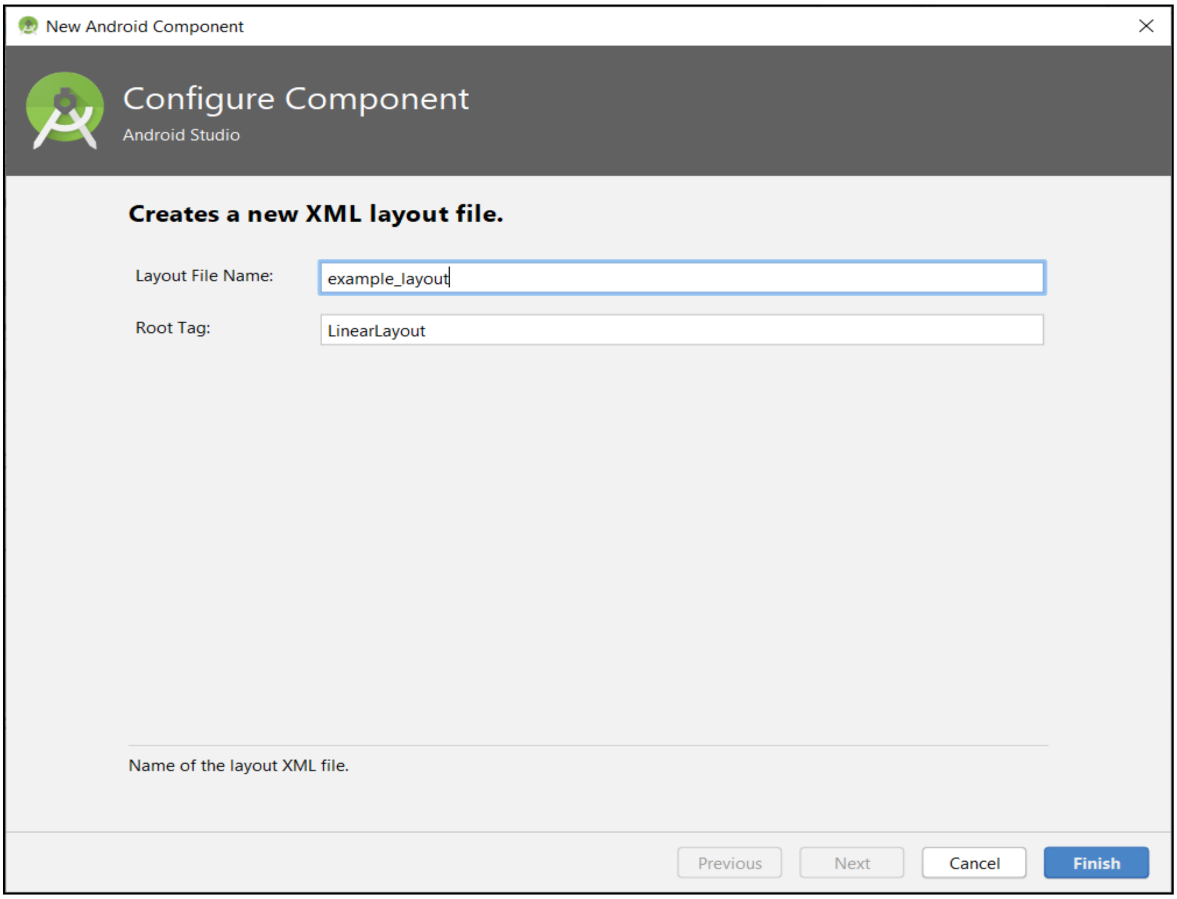
Creating a Layout inAndroid

Right Click on Layout Folder ❼ New ❼ XML ❼ Layout XMLFile


My Application [D:\Projects\MyApplication] - ...\app\src\main\java\com\example\myapplication\LoginActivity.java [app] - Android Studio
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
MyApplication app src main res layout
Android
app
manifests
java
com.example.myapplication
LoginActivity
MainActivity
com.example.myapplication (androidTest)
com.example.myapplication (test)
java (generated)
assets
res
drawable
ic_launcher_background.xml
ic_launcher_foreground.xml (v24)
layout
activity_login.xml
activity_main.xml
mipmap
values
colors.xml
strings.xml
styles.xml
res (generated)
Gradle Scripts
MainActivity.java
activity_login.xml
LoginActivity.java
activity_main.xml
New
Link C++ Project with Gradle
Cut Ctrl+X
Copy Ctrl+C
Copy Path Ctrl+Shift+C
Copy Reference Ctrl+Alt+Shift+C
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Analyze
Refactor
Add to Favorites
Show In Resource Manager Ctrl+Shift+T
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Run 'Tests in 'layout'' Ctrl+Shift+F10
Debug 'Tests in 'layout''
Run 'Tests in 'layout'' with Coverage
Create 'Tests in 'layout''...
Show in Explorer
Open in Terminal
Local History
Synchronize 'layout'
Directory Path Ctrl+Alt+F12
Compare With... Ctrl+D
Remove BOM
Kotlin File/Class
Layout resource file
Sample Data Directory
File
Scratch File Ctrl+Alt+Shift+Insert
Directory
C++ Class
C/C++ Source File
C/C++ Header File
Image Asset
Vector Asset
Kotlin Script
Singleton
Gradle Kotlin DSL Build Script
Gradle Kotlin DSL Settings
Edit File Templates...
AIDL
Activity
Android Auto
Folder
Fragment
Google
Other
Service
UI Component
Wear
Widget
XML
Resource Bundle
Layout XML File
App Actions XML File
Values XML File
Build: Build Output Sync
My Application: synced successfully
Run build D:\Projects\MyApplication
Load build
Configure build
Calculate task graph
Run tasks
4: Run TODO Terminal Build
Event Log
Create a new Layout XML File
1:1 CRLF UTF-8 4 spaces

```
1  package com.example.myapplication;
2
3  import
                                   ends AppCompatActivity {

                                   ndle savedInstanceState) {
                                   stanceState);
                                   t.activity_login);
```

Enter xml file name and press Finish



Creates a new XML layout file.

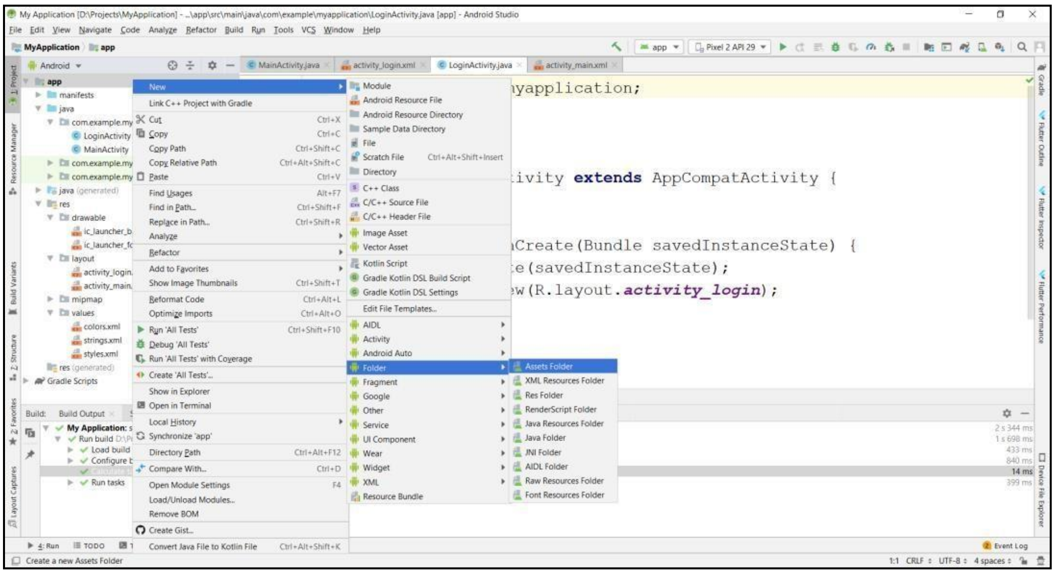
Layout File Name: example_layout

Root Tag: LinearLayout

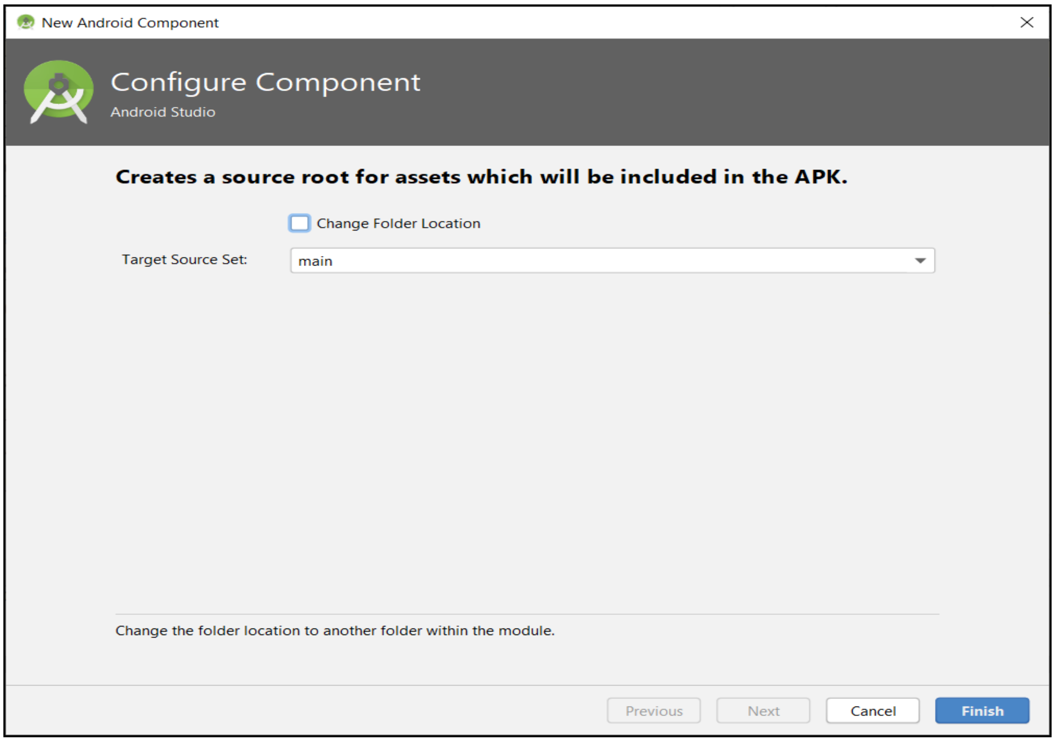
Name of the layout XML file.

Creating Assets Folder inAndroid

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



```
yapplication;

:ivity extends AppCompatActivity {

Create(Bundle savedInstanceState) {
:e(savedInstanceState);
w(R.layout.activity_login);
```


New Android Component

Configure Component

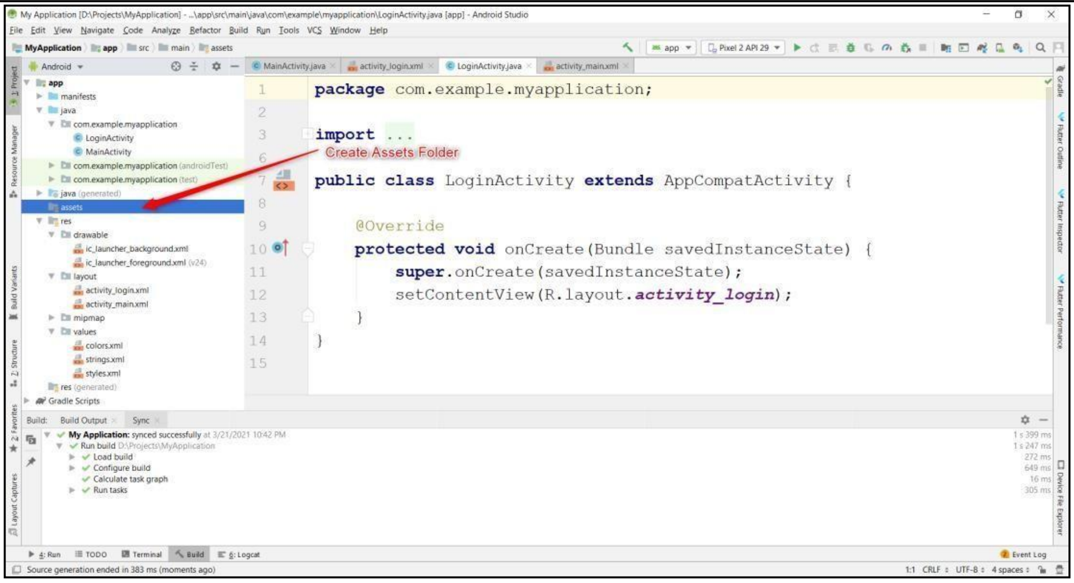
Android Studio

Creates a source root for assets which will be included in the APK.

☐ Change Folder Location

Target Source Set: main

Change the folder location to another folder within the module.



```
package com.example.myapplication;

import ...

public class LoginActivity extends AppCompatActivity {

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

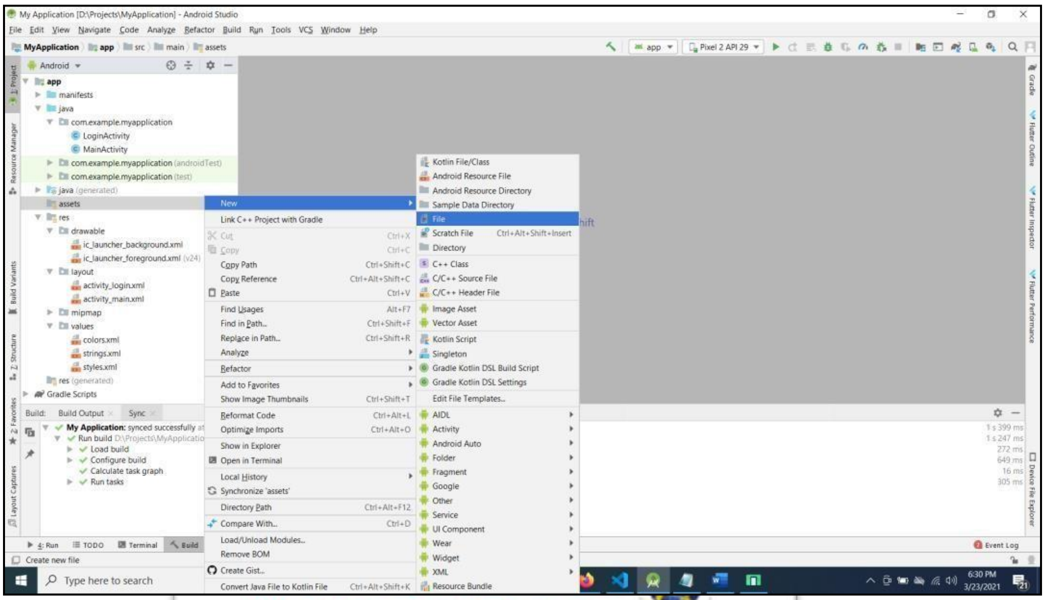
Create Assets Folder

```
My Application: synced successfully at 3/21/2021 10:42 PM
  Run build D:\Projects\MyApplication
    Load build
    Configure build
    Calculate task graph
    Run tasks
```

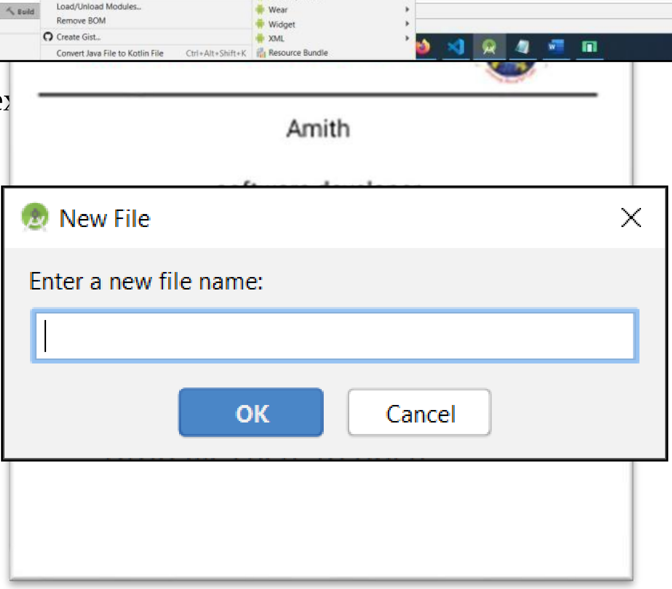
Source generation ended in 383 ms (moments ago)

Creating File in assetsFolder:

RightClickonassetsfolder ➔ New ➔ File



Enter filename with ex

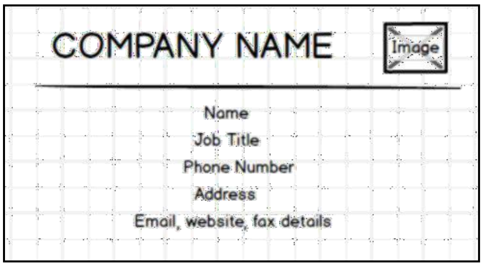


Programs

PART A

Program 1

CreateanapplicationtodesignaVisitingCard.TheVisitingcardshouldhaveacompanylogoat 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 phonenumber.



1. Create a New Android Project with EmptyActivity.
2. Openactivity_main.xmlfilefromres ➐ layoutfolder,check/addLinearLayoutastherootview.
3. Create layout using nested Relative Layout andTextView.
4. Use View background property to draw theline
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 thecomponents.

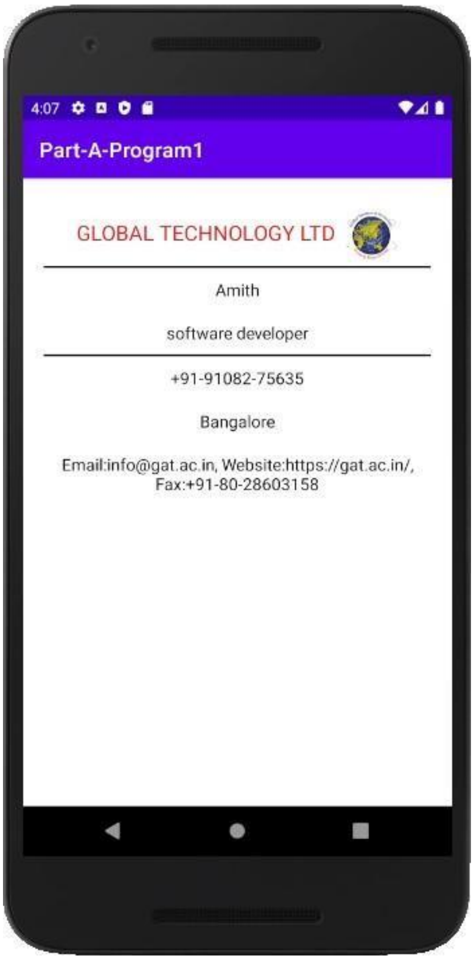
Design

activity_main.xml

```
<?xmlversion="1.0"encoding="utf-8"?>
<LinearLayoutxmlns: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:paddingLeft="20dp"
android:paddingTop="25dp"
android:paddingRight="20dp"
tools:context=".MainActivity">


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


<TextViewandroid:id="@+id/textView"
android:layout_width="wrap_content"
android:layout_height="44dp"
android:layout_alignParentStart="true"
android:layout_alignParentBottom="true"
android:layout_marginStart="31dp"
android:layout_marginLeft="20dp"
android:layout_marginBottom="10dp"
android:gravity="center"
android:text="GLOBALTECHNOLOGYLTD"
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="11dp"
android:layout_marginBottom="0dp"
android:layout_toRightOf="@id/textView"
app:srcCompat="@drawable/AMCEC_logo"/>

</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="Amith"
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="softwaredeveloper"
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="+91-91082-75635"
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:info@AMCEC.ac.in,Website:https://AMCEC.ac.in/,
                                                                                 Fax:+91-80-
28603158"
android:textSize="16dp"
android:layout_marginBottom="10dp"
android:layout_marginTop="10dp"
android:textColor="#000000"
android:gravity="center"
/>

</LinearLayout>
```

Sample Output



GLOBAL TECHNOLOGY LTD

Amith

software developer

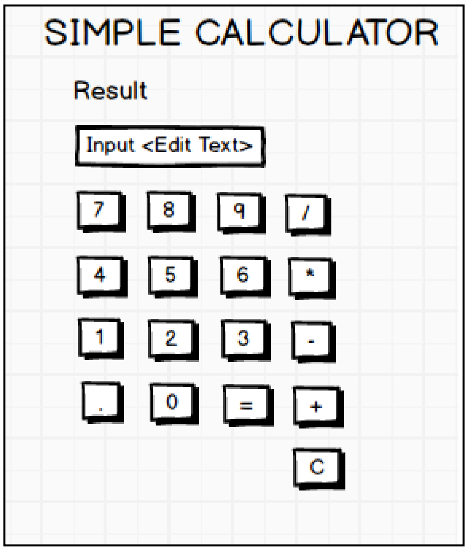
+91-91082-75635

Bangalore

Email:info@gat.ac.in, Website:https://gat.ac.in/,
Fax:+91-80-28603158

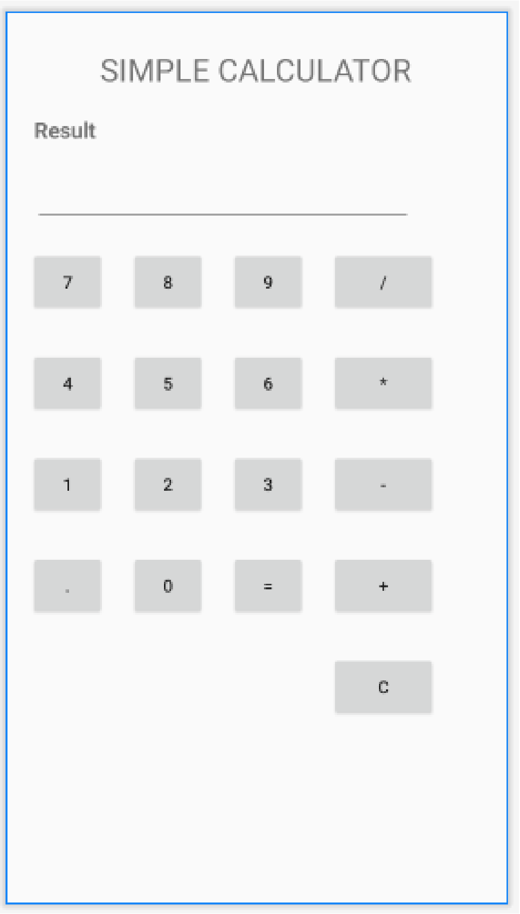
Program 2

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 EmptyActivity.
2. Open activity_main.xml filefromres ❼ layout folder, check/add ConstraintLayoutastherootview.
3. Create Layout using Drag and Dropframework.
4. OpenMainActivty.javafile,OverrideonCreate()methodandbringactivity_main.xmlfile on screen using setContentView() and bring the view references using findViewById() method.
5. Add Listeners to Button ClickEvent:
6. Create a class which implements OnClickListenerinterface.
7. Override onClick() method of OnClickListenerInterface.
8. Register the button for click event by calling setOnClickListener() method of View class and pass the object of the class that implemented OnClickListenerInterface.
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

```
<?xmlversion="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=".MainActivity">

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

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

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

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

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

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

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

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

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

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

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

<Button
android:id="@+id/button_seven"
android:layout_width="62dp"
android:layout_height="53dp"
android:layout_marginStart="20dp"
android:layout_marginTop="20dp"
android:text="7"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/txt_result"/>

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

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

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

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

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

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

<Button
android:id="@+id/button_div"
android:layout_width="87dp"
android:layout_height="53dp"
android:layout_marginStart="20dp"
android:layout_marginTop="20dp"



android:text="/"
```

```
app:layout_constraintStart_toEndOf="@+id/button_nine"
app:layout_constraintTop_toBottomOf="@+id/txt_result"/>

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

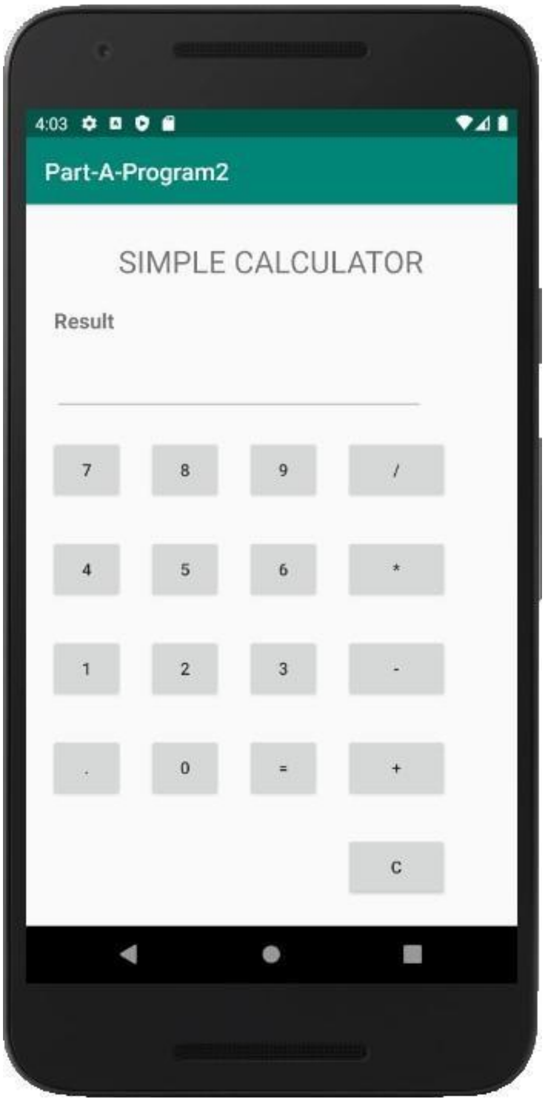
MainActivity.java

```
packagecom.example.partaprogram2;

importandroidx.appcompat.app.AppCompatActivity;
importandroid.os.Bundle;
importandroid.view.View;
importandroid.widget.Button;
importandroid.widget.EditText;
importandroid.widget.Toast;

importjava.util.regex.Pattern;

publicclassMainActivityextendsAppCompatActivityimplementsView.OnClickListener{
    ButtonbtnOne,btnTwo,btnThree,btnFour,btnFive,btnSix;
    Button  btnSeven,  btnEight,  btnNine,
            btnZero;ButtonbtnAdd,btnSub,btnMul,btnDi
    v;
    ButtonbtnClear,btnEqual,btnDot;

EditTexttxtResult;

@Override
protectedvoidonCreate(BundlesavedInstanceState){
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("");
}
```

```
publicvoidonClick(Viewv)
{
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(btnEqual))
{
try {

Stringdata=txtResult.getText().toString();
if(data.contains("/")){
String[]operands=data.split("/");
if(operands.length==2){
doubleoperand1=Double.parseDouble(operands[0]);
doubleoperand2=Double.parseDouble(operands[1]);
doubleresult=operand1/operand2;
txtResult.setText(String.valueOf(result));
                      }
else
{
Toast.makeText(getBaseContext(),"InvalidInput",
Toast.LENGTH_LONG).show();
}

}
elseif(data.contains("*")){
String[]operands=data.split(Pattern.quote("*"));
if(operands.length==2){
doubleoperand1=Double.parseDouble(operands[0]);
doubleoperand2=Double.parseDouble(operands[1]);
doubleresult=operand1*operand2;
txtResult.setText(String.valueOf(result));
}
else
{
Toast.makeText(getBaseContext(),"InvalidInput",
Toast.LENGTH_LONG).show();
}

}
elseif(data.contains("+")){
String[]operands=data.split(Pattern.quote("+"));
if(operands.length==2){
doubleoperand1=Double.parseDouble(operands[0]);
doubleoperand2=Double.parseDouble(operands[1]);
doubleresult=operand1+operand2;
txtResult.setText(String.valueOf(result));
}
else
{
Toast.makeText(getBaseContext(),"InvalidInput",
Toast.LENGTH_LONG).show();
}
}
elseif(data.contains("-")){
String[]operands=data.split("-");
if(operands.length==2){
doubleoperand1=Double.parseDouble(operands[0]);
doubleoperand2=Double.parseDouble(operands[1]);
doubleresult=operand1-operand2;
txtResult.setText(String.valueOf(result));
}
else
{
Toast.makeText(getBaseContext(),"InvalidInput",
Toast.LENGTH_LONG).show();
}
}
}

catch(Exceptione){
Toast.makeText(getBaseContext(),"InvalidInput",
Toast.LENGTH_LONG).show();
}
}
```

```
if(v.equals(btnAdd))
txtResult.append("+");
if(v.equals(btnSub))
txtResult.append("-");
if(v.equals(btnMul))
txtResult.append("*");
if(v.equals(btnDiv))
txtResult.append("/");
}
}
```

Sample Output

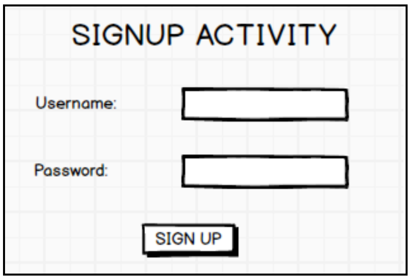


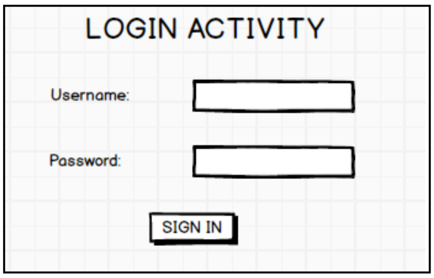
Program 3

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

- Password should contain uppercase and lowercaseletters.
- Password should contain letters andnumbers.
- Password should contain specialcharacters.
- Minimum length of the password (the default value is8).

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 naviAMCECe 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 fromone activity to another.

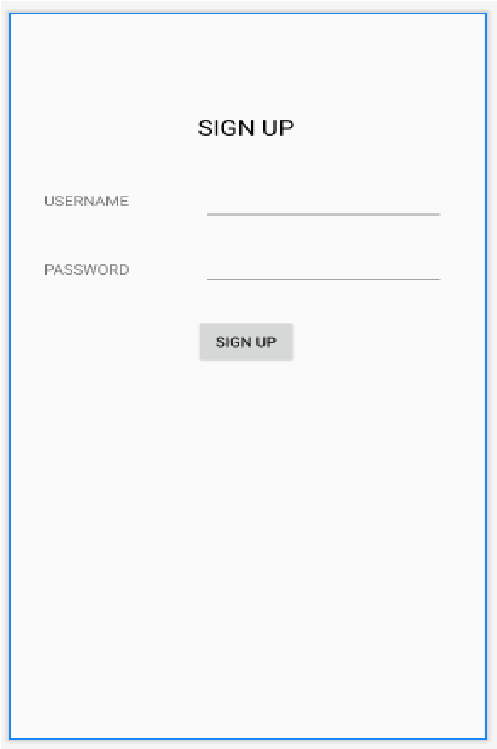


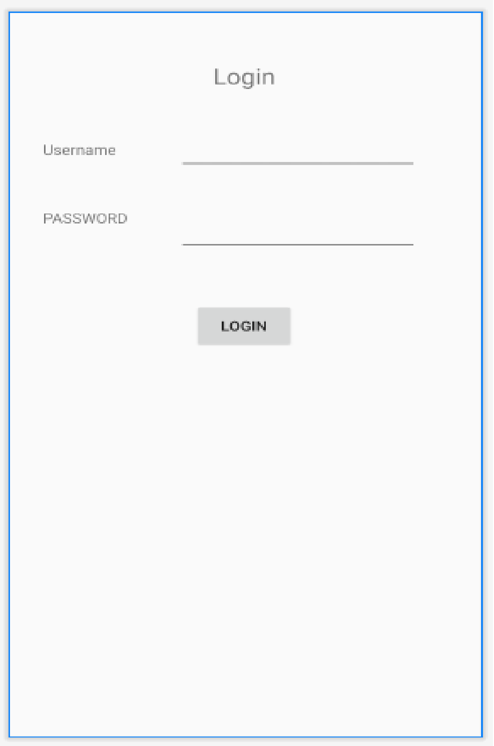


1. Create a New Android Project with EmptyActivity.
2. Open activity_main.xml filefromres ➔ layout folder, check/add ConstraintLayoutastherootview.
3. Create Signup Layout using Drag and Drop framework design thelayout.
4. Create One more Empty Activity LoginActivity using Android Studio Create Activity Flow (Refer Android StudioTutorial)
5. Openactivity_login.xmlfilefromres ➔ layoutfolder,check/addConstraintLayout asthe rootview.
6. Create Login Layout using Drag and Dropframework.
7. Add Listeners to Button ClickEvent:
   - Create a class which implements OnClickListenerinterface.
   - Override onClick() method of OnClickListenerInterface.
   - Register the button for click event by calling setOnClickListener() method of View class and pass the object of the class that implemented OnClickListenerInterface.

8. Use Regular Expression**"^(?=.*[A-Z])(?=.*[a-z])(?=.*\\d)(?=.*[@$!])[A-Za-z\\d@$!]{8,}$”** to validate thepassword.

Design





Activity_main.xml

```
<?xmlversion="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=".MainActivity">

<TextView android:id="@+id/textView2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="100dp"
android:text="SIGNUP"
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="SignUp"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/txt_password"/>
</androidx.constraintlayout.widget.ConstraintLayout>
```

Activity_login.xml

```
<?xmlversion="1.0"encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayoutxmlns:android="http://schemas.androi
d.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

```
packagecom.example.parta.program3;

importandroidx.appcompat.app.AppCompatActivity;

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

importjava.util.regex.Matcher;
importjava.util.regex.Pattern;

publicclassMainActivityextendsAppCompatActivityimplementsView.OnClickListener{

EditTexttxtUsername;
EditTexttxtPassword;

ButtonbtnSignup;

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

```
@Override
protectedvoidonCreate(BundlesavedInstanceState){
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);

}

publicvoidonClick(Viewv)
{
     Stringusername=txtUsername.getText().toString();
     Stringpassword=txtPassword.getText().toString();

      if(validatePassword(password)){
      Bundlebundle=newBundle();
      bundle.putString("user",username);
      bundle.putString("Lab@2018",password);

      Intentit=newIntent(this,LoginActivity.class);
      it.putExtra("data",bundle);

      startActivity(it);
      }
else
{
```

```
Toast.makeText(getBaseContext(),"InvalidPassword",
Toast.LENGTH_LONG).show();
}
}

publicbooleanvalidatePassword(Stringpassword)
    {
       Patternpattern=Pattern.compile(regularExpression);
        Matchermatcher=pattern.matcher(password);
returnmatcher.matches();

    }
}
```

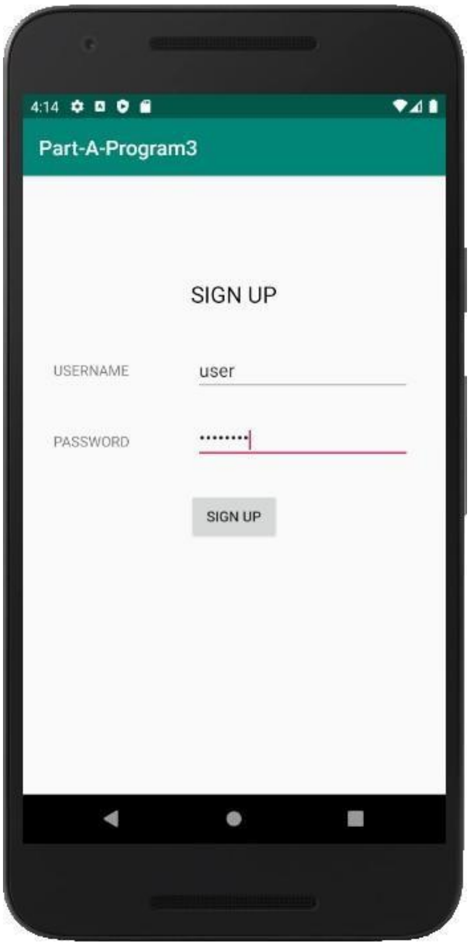
LoginActivity.java

```
packagecom.example.parta.program3;

importandroidx.appcompat.app.AppCompatActivity;
importandroid.os.Bundle;
importandroid.view.View;
importandroid.widget.Button;
importandroid.widget.EditText;
importandroid.widget.Toast;

publicclassLoginActivityextendsAppCompatActivityimplementsView.OnClickListener{

EditTexttxtLoginUsername;
EditTexttxtLoginPassword;
    ButtonbtnLogin;

    Stringuser,pass;

intcount=0;

@Override
protectedvoidonCreate(BundlesavedInstanceState){
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);

        Bundlebundle=getIntent().getBundleExtra("data");
user=bundle.getString("user");
pass=bundle.getString("Lab@2018");

   }

publicvoidonClick(Viewv)
    {
        Stringuser1=txtLoginUsername.getText().toString();
        Stringpass1=txtLoginPassword.getText().toString();

if(user.equals(user1)&&pass.equals(pass1))
        {
Toast.makeText(this,"LoginSuccessful"
,Toast.LENGTH_LONG).show();
        }
else
```

```
{count++; if(count==3)
            {
btnLogin.setEnabled(false);
Toast.makeText(this,
"FailedLoginAttempts"
,Toast.LENGTH_LONG).show();
            }
else
{
Toast.makeText(this,"LoginFailed"+count
,Toast.LENGTH_LONG).show();
            }
        }

    }
}
```

Sample Output

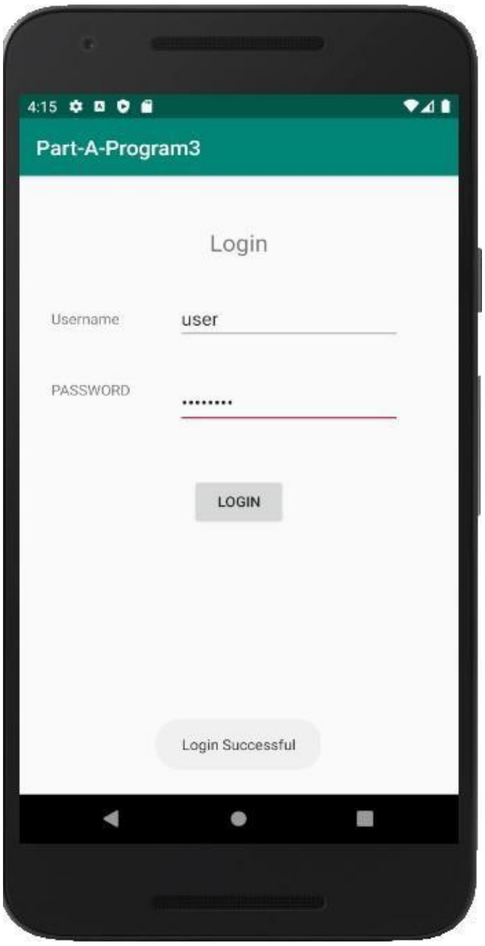


Part-A-Program3

SIGN UP

USERNAME user

PASSWORD ••••••••



Part-A-Program3

Login

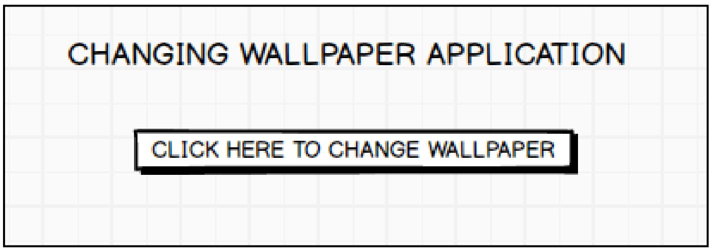
Username user

PASSWORD ••••••••

Login Successful

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
CLICK HERE TO CHANGE WALLPAPER

1. Create a New Android Project with EmptyActivity.
2. Open activity_main.xml filefromres ➔ layout folder,check/addLinearLayoutastherootview.
3. Create thelayout
4. Add 3 or Moreimagesto drawablefolder(res ➔ drawable)
5. Declare uses permission android.permission.SET_WALLPAPPER in the AndroidManifest.xmlfile
6. Schedule Timer task to change the wallpaper on every 30 secondsinterval.
7. Initialize and use WallpaperManager.setBitmap() method to change thewallpaper.

activity_main.xml

```
<?xmlversion="1.0"encoding="utf-8"?>
<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent" android:layout_height="match_parent"
android:orientation="vertical"
android:gravity="center"
tools:context=".MainActivity">

<Button
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="ClickheretoChangeWallpaper"
android:id="@+id/btn_start_change_wallpaper"/>

</LinearLayout>
```

MainActivity.java

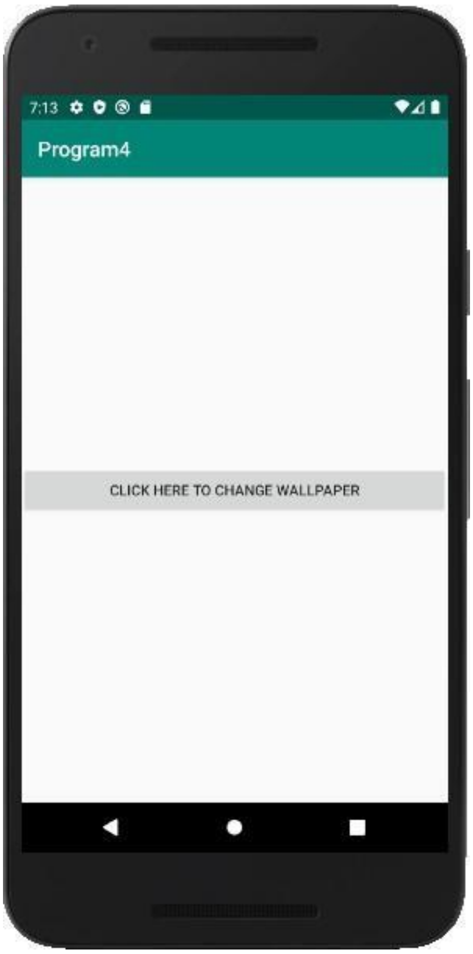
```
packagecom.example.program4;

importandroidx.appcompat.app.AppCompatActivity;

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

importjava.util.Timer;
importjava.util.TimerTask;

publicclassMainActivityextendsAppCompatActivityimplementsView.OnClickListener{

ButtonbtnChangeWallpaper;
booleanrunning;
int[]imagesArray=newint[]{

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,
R.drawable.img11,
R.drawable.img12
};

inti=0;

@Override
protectedvoidonCreate(BundlesavedInstanceState){
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);

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

}
```

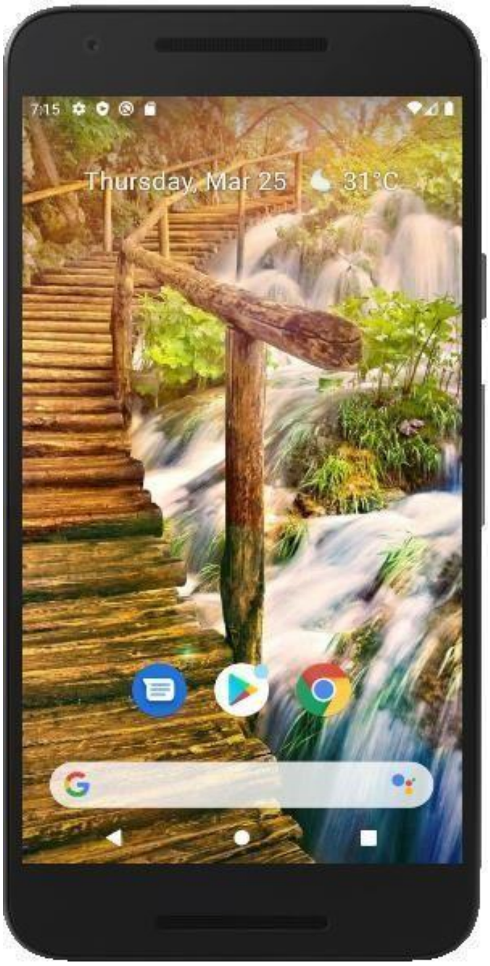
```
publicvoidonClick(Viewv)
{
   if(!running)
   {
     newTimer().schedule(newMyTimer(),0,3000);
     running=true;
    }
}

classMyTimerextendsTimerTask
{
publicvoidrun()
{
try
{
WallpaperManagerwallpaperManager=
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.setBitmap(BitmapFactory.decodeResource(getResources()
                                 ,imagesArray[i]));
i++;
            }
catch(Exceptione)
            {

            }

        }
    }

}
```

AndriodManifest.xml

```
<?xmlversion="1.0"encoding="utf-8"?>
<manifestxmlns:android="http://schemas.android.com/apk/res/android"
package="com.example.program4">

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

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

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

</manifest>
```

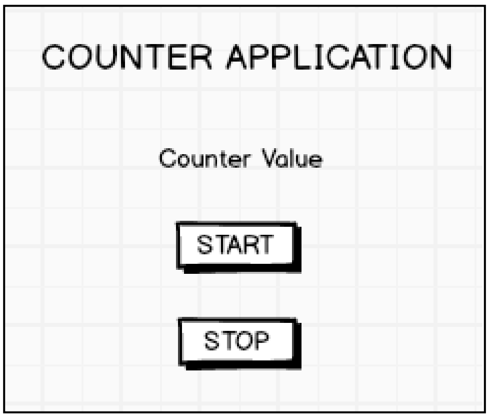
Sample Output





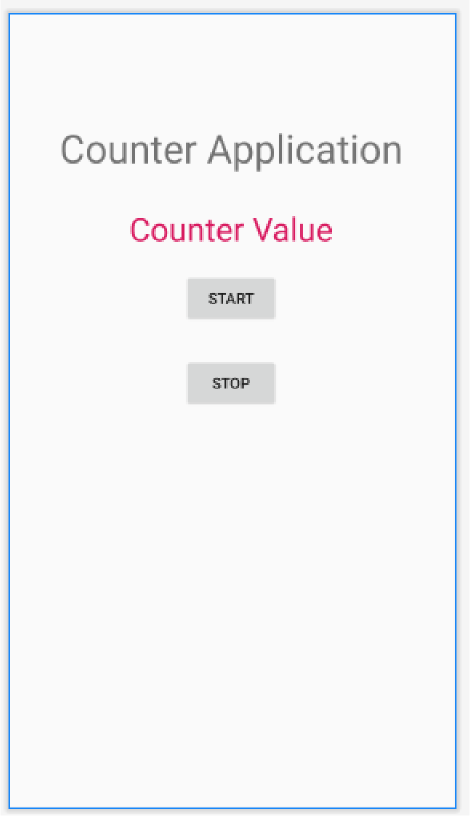
Program 5

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 EmptyActivity.
2. Openactivity_main.xmlfilefromres ➐ layoutfolder,check/addConstraintLayoutastherootview.
3. Create the layout design using Drag and Dropframework.
4. Add Listeners to Button ClickEvent:
   - Create a class which implements OnClickListenerinterface.
   - Override onClick() method of OnClickListenerInterface.
   - Register the button for click event by calling setOnClickListener() method of View class and pass the object of the class that implemented OnClickListenerInterface.
5. Create a Thread to start the counterlogic.
6. Steps to Create aThread
   - Create a class that extends ThreadClass.
   - Override run method of Thread Class.
   - Use start() method of thread class to start thethread.
7. Create Handler class to receive message from child thread, Handler executes in Main Thread.
8. Steps to CreateHandler
   - Create Object of typeHandler.
   - OverridhandleMessage() of handlerclass.
9. Pass the counter value to be displayed to thehandler.
10. Update the UI to display the counter value received fromthread.

Design



activity_main.xml

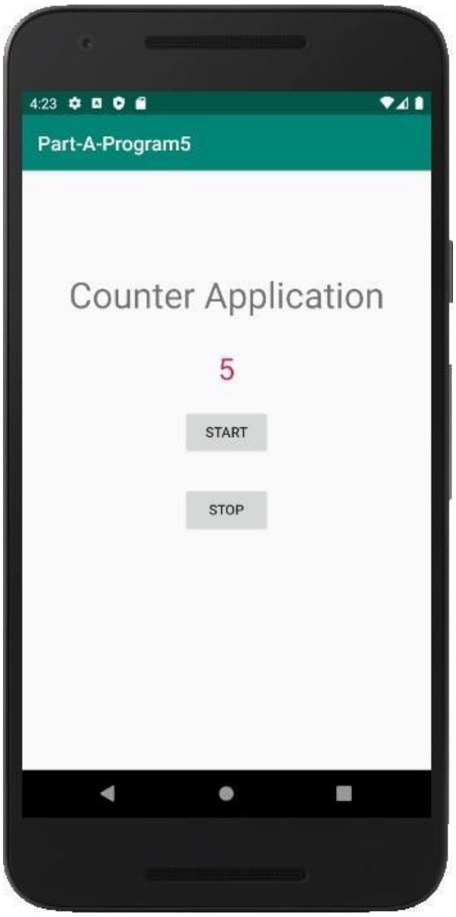
```
<?xmlversion="1.0"encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayoutxmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/lbl_counter"
android:layout_width="match_parent"
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="CounterApplication"
android:textSize="36sp"
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="CounterValue"
android:textColor="@color/colorAccent"
android:textSize="30sp"
```

```
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

```
packagecom.example.program5;

importandroidx.appcompat.app.AppCompatActivity;

importandroid.os.Bundle;
importandroid.os.Handler;
importandroid.os.Message;
importandroid.view.View;
importandroid.widget.Button;
importandroid.widget.TextView;

importorg.w3c.dom.Text;

publicclassMainActivityextendsAppCompatActivityimplementsView.OnClickListener{

TextViewlblCounter;
ButtonbtnStart,btnStop;

intcounter=0;
booleanrunning=false;

@Override
protectedvoidonCreate(BundlesavedInstanceState){
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);
}

publicvoidonClick(Viewv)
{
if(v.equals(btnStart))
{
   counter=0;
   running=true;
   newMyCounter().start();
  }
  elseif(v.equals(btnStop))
  {
    running=false;
  }
}

Handlerhandler=newHandler()
{
publicvoidhandleMessage(Messagem)
{
   lblCounter.setText(String.valueOf(m.what));
}

};


classMyCounterextendsThread
{
  publicvoidrun()
  {
    while(running)
    {
     counter++;
     handler.sendEmptyMessage(counter);

     try {
    Thread.sleep(1000);
    }
    catch(Exceptione){     }
      }
   }
 }
}
```

Sample Output



4:23

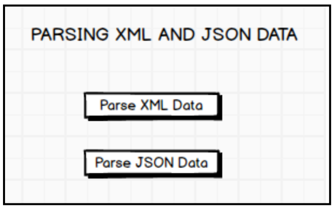
Part-A-Program5

Counter Application

5

Program 6

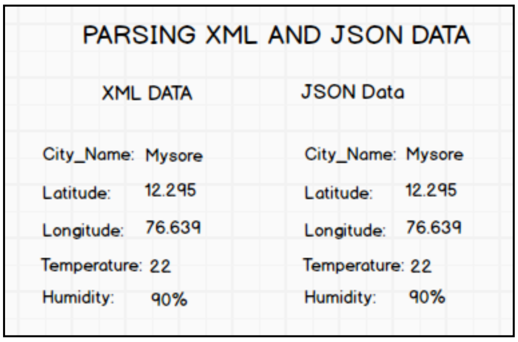
Create two files of XML and JSON type with values for City_Name, Latitude, Longitude, Temperature, and Humidity. Develop an application to create an activity with two buttons to parse the XML and JSON files which when clicked should display the data in their respective layouts side byside.



PARSING XML AND JSON DATA

Parse XML Data

Parse JSON Data



PARSING XML AND JSON DATA

XML DATA

City_Name: Mysore
Latitude: 12.295
Longitude: 76.639
Temperature: 22
Humidity: 90%

JSON Data

City_Name: Mysore
Latitude: 12.295
Longitude: 76.639
Temperature: 22
Humidity: 90%

1. Create a New Android Project with EmptyActivity.
2. Openactivity_main.xmlfilefromres ❼ layoutfolder,check/addConstraintLayoutasrootview.
3. Create the layout design using Drag and Dropframework.
4. Add Listeners to Button ClickEvent:
   - Create a class which implements OnClickListenerinterface.
   - 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 OnClickListenerInterface.
5. Create assets folder (Refer Section Android StudioTutorial)
6. Create **input.xml** file inside assets folder and paste the below XmlData

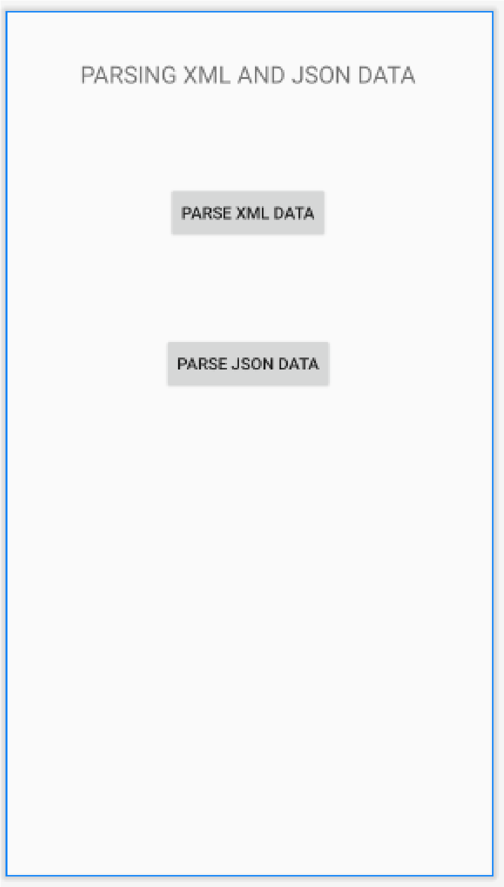
```
<?xml version="1.0"?>
<records>
<employee>

<city_name>Mysore</city_name>
<Latitude>12.295</Latitude>
<Longitude>76.639</Longitude>
<Temperature>22</Temperature>
<Humidity>90%</Humidity>
</employee>
</records>
```

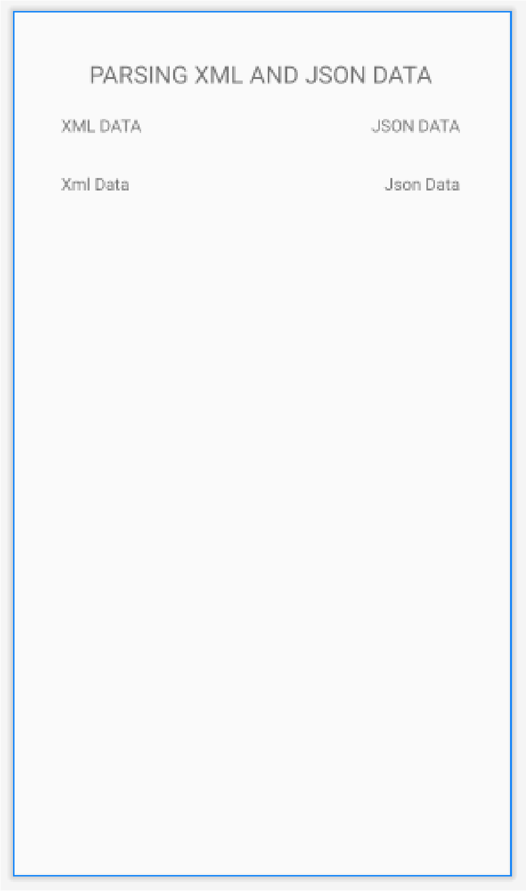
7. Create **input.json** file inside assets folder and paste the below JsonData

```
{
"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 onscreen

Design





activity_main.xml

```
<?xmlversion="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=".MainActivity">

<Button
android:id="@+id/btn_parsexml"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="80dp"
android:text="ParseXMLData"
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="ParseJsonData"
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="PARSINGXMLANDJSONDATA"
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

```
<?xmlversion="1.0"encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayoutxmlns:android="http://schemas.androi
d.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"android:layout
_width="match_parent" android:layout_height="match_parent"
tools:context=".ViewActivity">

<TextView
android:id="@+id/lbl_xml_data"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="30dp"
android:text="XmlData"
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="PARSINGXMLANDJSONDATA"
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="XMLDATA"
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="JSONDATA"
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="JsonData"
app:layout_constraintEnd_toEndOf="@+id/textView3"
app:layout_constraintTop_toBottomOf="@+id/textView3"/>
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

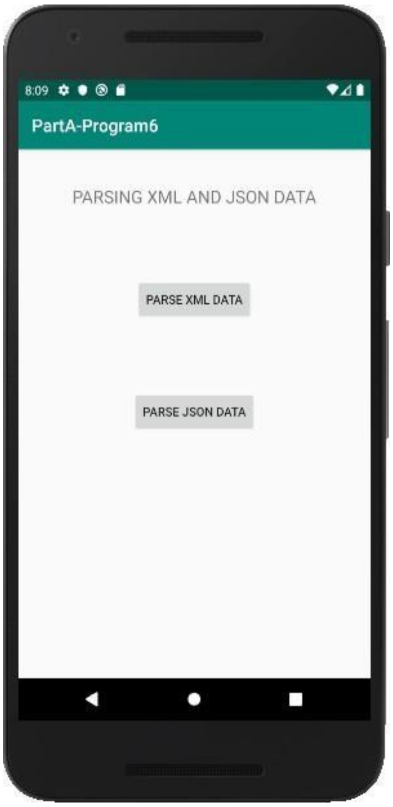
```
packagecom.example.parta_program6;

importandroidx.appcompat.app.AppCompatActivity;

importandroid.content.Intent;
importandroid.os.Bundle;
importandroid.view.View;
importandroid.widget.Button;

publicclassMainActivityextendsAppCompatActivityimplementsView.OnClickListener{

ButtonbtnParseXml,btnParseJson;

@Override
protectedvoidonCreate(BundlesavedInstanceState){
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
publicvoidonClick(Viewv){

if(v.equals(btnParseJson))
{
    Intentit=newIntent(this,ViewActivity.class);
    it.putExtra("mode",1);
    startActivity(it);

}
elseif(v.equals(btnParseXml))
{
     Intentit=newIntent(this,ViewActivity.class);
     it.putExtra("mode",2);
    startActivity(it);
}
}
}
```

ViewActivity.java

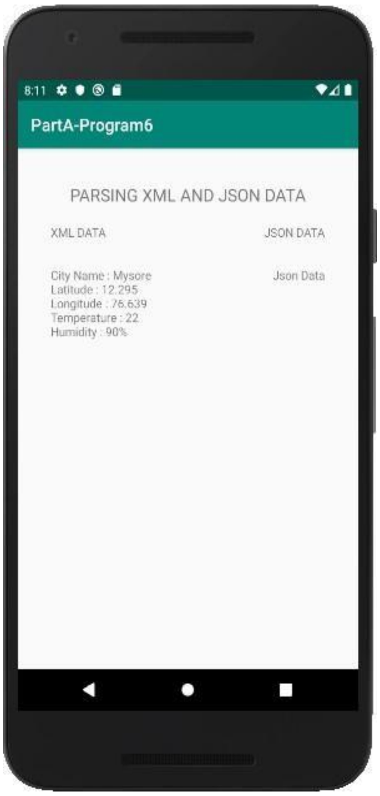
```
packagecom.example.parta_program6;

importandroidx.appcompat.app.AppCompatActivity;

importandroid.os.Bundle;
importandroid.widget.TextView;

importorg.json.JSONObject;
importorg.w3c.dom.Document;
importorg.w3c.dom.Element;
importorg.w3c.dom.Node;
importorg.w3c.dom.NodeList;

importjava.io.InputStream;

importjavax.xml.parsers.DocumentBuilder;
importjavax.xml.parsers.DocumentBuilderFactory;

publicclassViewActivityextendsAppCompatActivity{ TextViewlblXmlData,lblJsonData;

intmode=0;@O
verride
protectedvoidonCreate(BundlesavedInstanceState){
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();

}

publicStringparseXmlDocument()
{
try {

InputStreamis=getAssets().open("input.xml");

DocumentBuilderFactorydbFactory=DocumentBuilderFactory.newInstance();
DocumentBuilderdBuilder=dbFactory.newDocumentBuilder();
Documentdoc=dBuilder.parse(is);

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

NodeListnList=doc.getElementsByTagName("employee");

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

Nodenode=nList.item(i);
if(node.getNodeType()==Node.ELEMENT_NODE){
Elementelement2=(Element)node;
lblXmlData.setText("CityName:"+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(Exceptione){e.printStackTrace();}
return  null;
}

privatestaticStringgetValue(Stringtag,Elementelement){
NodeListnodeList=element.getElementsByTagName(tag).item(0).getChildNodes();
Nodenode=nodeList.item(0);
returnnode.getNodeValue();
}

publicvoidparseJson()
{
try {
InputStreaminputStream=getAssets().open("input.json");
byte[]data=newbyte[inputStream.available()];
inputStream.read(data);
StringreadData=newString(data);
JSONObjectjsonObject=newJSONObject(readData);
JSONObjectjsonObject1=jsonObject.getJSONObject("employee");
lblJsonData.setText("CityName:"+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(Exceptione){e.printStackTrace();}
 }
}
```

Sample Output



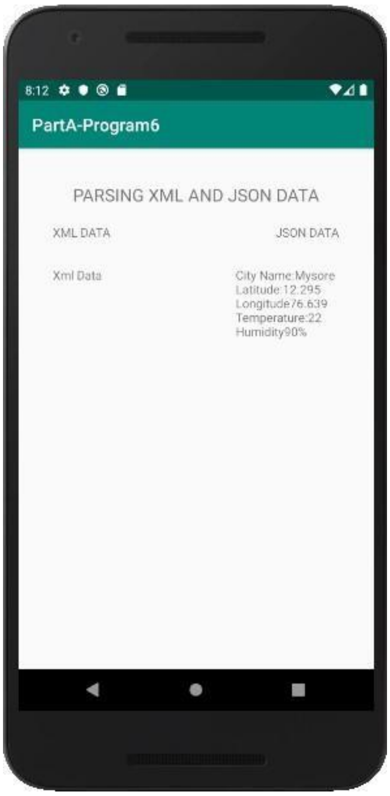
PARSING XML AND JSON DATA



PARSING XML AND JSON DATA

XML DATA JSON DATA

```
City Name : Mysore          Json Data
Latitude : 12.295
Longitude : 76.639
Temperature : 22
Humidity : 90%
```



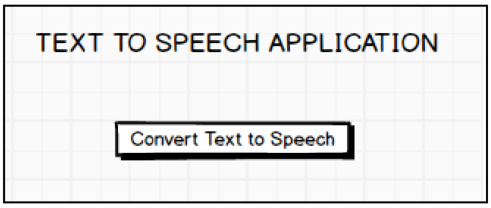
PARSING XML AND JSON DATA

XML DATA JSON DATA

```
Xml Data          City Name:Mysore
                  Latitude:12.295
                  Longitude76.639
                  Temperature:22
                  Humidity90%
```

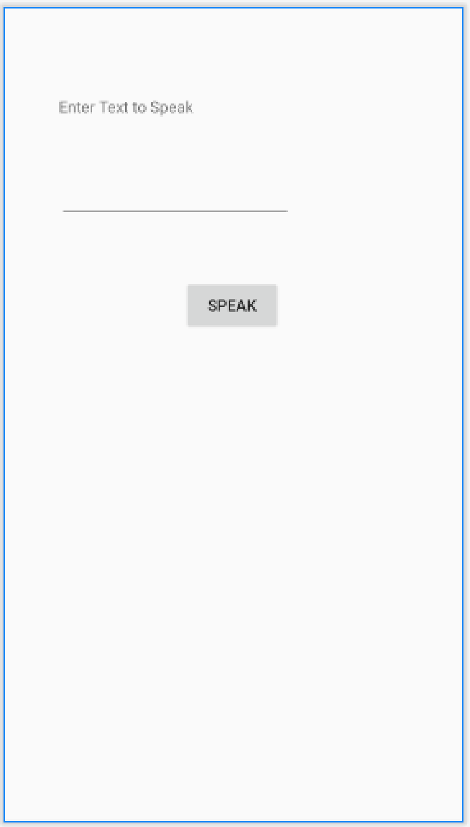
Program 7

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 EmptyActivity.
2. Openactivity_main.xmlfilefromres ❼ layoutfolder,check/addConstraintLayoutasrootview.
3. Create the layout design using Drag and Dropframework.
4. Add Listeners to Button ClickEvent:
   - Create a class which implments OnClickListenerinterface.
   - Override onClick() method of OnClickListenerInterface.
   - Register the button for click event by calling setOnClickListener() method of View class and pass the object of the class that implemented OnClickListenerInterface.
5. Initialize TextToSpeech Engine and the Language to Speak using setLanguage()method
6. Use Speak() method to speak the text passed toit.

Design



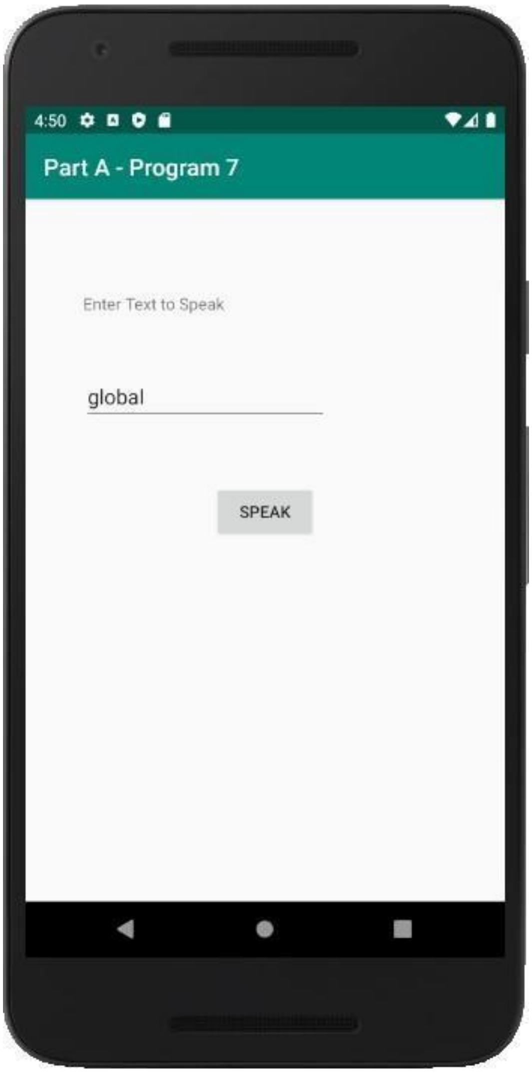
activity_main.xml

```
<?xmlversion="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: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="EnterTexttoSpeak"
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

```
packagecom.example.parta.parta_program7;

importandroidx.appcompat.app.AppCompatActivity;

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

importjava.util.Locale;

publicclassMainActivityextendsAppCompatActivityimplementsView.OnClickListener{

EditTexttxtSpeak;
ButtonbtnSpeak;
TextToSpeechtextToSpeech;

@Override
protectedvoidonCreate(BundlesavedInstanceState){
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=newTextToSpeech(getBaseContext(),
newTextToSpeech.OnInitListener(){
@Override
publicvoidonInit(intstatus){
if(status!=TextToSpeech.ERROR)
{
Toast.makeText(getBaseContext(),"Success",Toast.LENGTH_LONG).show();
 }
    }
 });
textToSpeech.setLanguage(Locale.UK);
}

publicvoidonClick(Viewv)
{
        Stringtext=txtSpeak.getText().toString();
        textToSpeech.speak(text,TextToSpeech.QUEUE_FLUSH,null);

 }
}
```

Sample Output



4:50

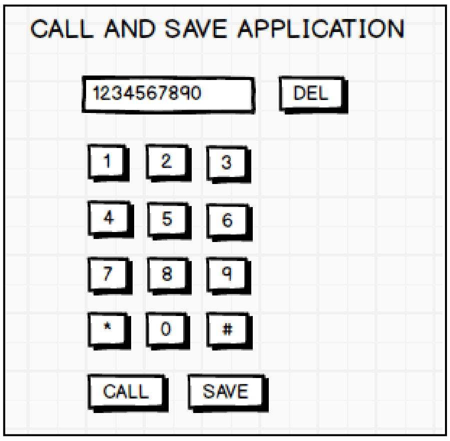
Part A - Program 7

Enter Text to Speak

global

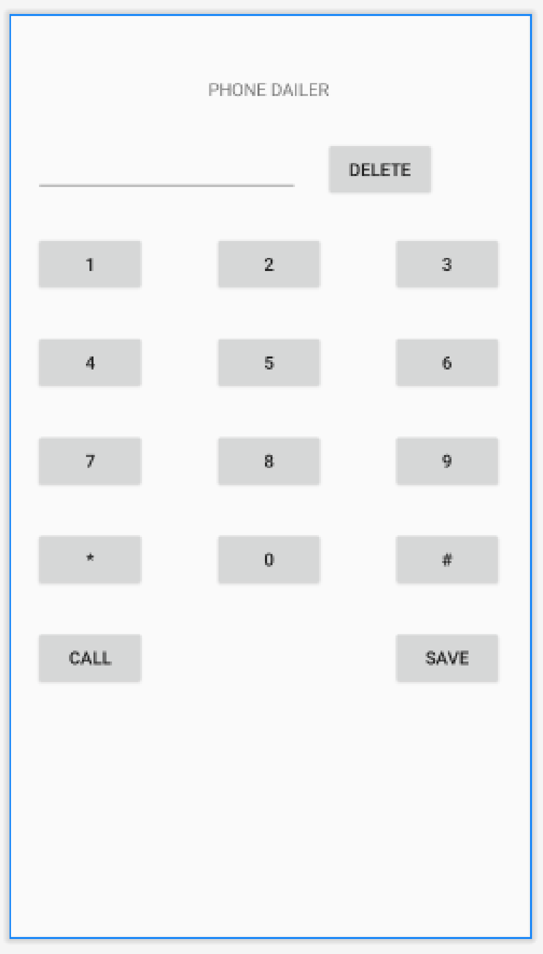
Program 8

CreateanactivitylikeaphonedialerwithCALLandSAVEbuttons.OnpressingtheCALLbutton,itmust callthephonenumberand onpressingtheSAVE buttonitmustsavethenumberto thephonecontacts.



1. Create a New Android Project with EmptyActivity.
2. Open activity_main.xml filefromres ❼ layout folder, check/add ConstraintLayout astherootview.
3. Create the layout design using Drag and Dropframework.
4. Add Listeners to Button ClickEvent:
   - Create a class which implments OnClickListenerinterface.
   - Override onClick() method of OnClickListenerInterface.
   - Register the button for click event by calling setOnClickListener() method of View class and pass the object of the class that implemented OnClickListenerInterface.
5. Declare uses permission android.permission.CALL_PHONE in the manifestfile.
6. Use ACTION_CALL intent name and pass the “tel:<phone-number> as URI in intent data and start the callactivity.
7. Use intent name and pass the “Telephone Number” and “unknown” as name as intent data call Contacts SaveActivity.

Design



activity_main.xml

```
<?xmlversion="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=".MainActivity">

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

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

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

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

<Button
android:id="@+id/btn_two"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="30dp"
android:text="2"
```

```
app:layout_constraintEnd_toStartOf="@+id/btn_three"
app:layout_constraintStart_toEndOf="@+id/btn_one"
app:layout_constraintTop_toBottomOf="@+id/txt_phonenumber"/>

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

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

<Button
android:id="@+id/btn_five"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="30dp"
android:text="5"
app:layout_constraintEnd_toStartOf="@+id/btn_six"
app:layout_constraintStart_toEndOf="@+id/btn_four"
app:layout_constraintTop_toBottomOf="@+id/btn_two"/>

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

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

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

```
android:text="8"
app:layout_constraintEnd_toStartOf="@+id/btn_nine"
app:layout_constraintStart_toEndOf="@+id/btn_seven"
app:layout_constraintTop_toBottomOf="@+id/btn_five"/>

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

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

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

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

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

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

```
android:layout_marginTop="30dp"
android:layout_marginEnd="20dp"
android:text="#"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintTop_toBottomOf="@+id/btn_nine"/>
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```
packagecom.example.part_a_program_8;

importandroidx.appcompat.app.AppCompatActivity;
importandroid.content.Intent;
importandroid.net.Uri;
importandroid.os.Bundle;
importandroid.provider.ContactsContract;
importandroid.view.View;
importandroid.widget.Button;
importandroid.widget.EditText;

publicclassMainActivityextendsAppCompatActivityimplementsView.OnClickListener{

ButtonbtnOne,btnTwo,btnThree,btnFour,btnFive;
ButtonbtnSix,btnSeven,btnEight,btnNine,btnZero;
ButtonbtnDel,btnStar,btnHash,btnCall,btnSave;

EditTexttxtPhonenumber;

@Override
protectedvoidonCreate(BundlesavedInstanceState){
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);
```

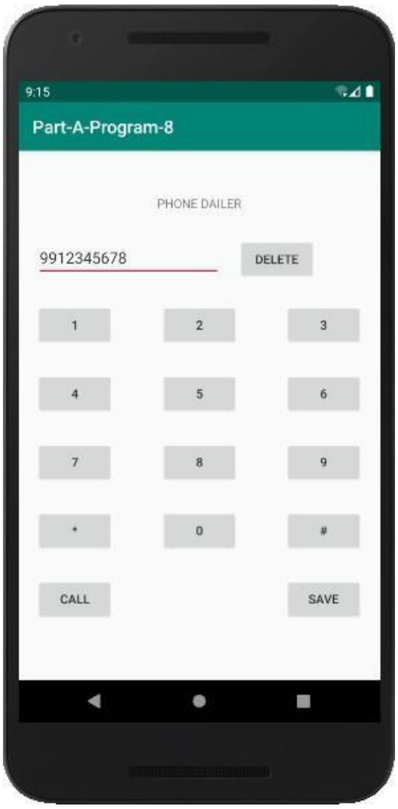
```
btnNine=(Button)findViewById(R.id.btn_nine);
btnNine.setOnClickListener(this);

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("");
}

publicvoidonClick(Viewv)
{
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");
```

```
elseif(v.equals(btnStar))
txtPhonenumber.append("*");

else  if(v.equals(btnHash))
txtPhonenumber.append("#");

else  if(v.equals(btnSave))
{
IntentcontactIntent=newIntent
                    (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))
{
Stringdata=txtPhonenumber.getText().toString();
if(data.length()>0)
{
txtPhonenumber.setText
                        (data.substring(0,data.length()-1));

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

btnCall.setOnClickListener(newView.OnClickListener()
        {
@Override
publicvoidonClick(Viewv){

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

    }

}
```

AndriodManifest.xml

```
<?xmlversion="1.0"encoding="utf-8"?>
<manifestxmlns: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">
<activityandroid:name=".MainActivity">
<intent-filter>
<actionandroid:name="android.intent.action.MAIN"/>

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

</manifest>
```

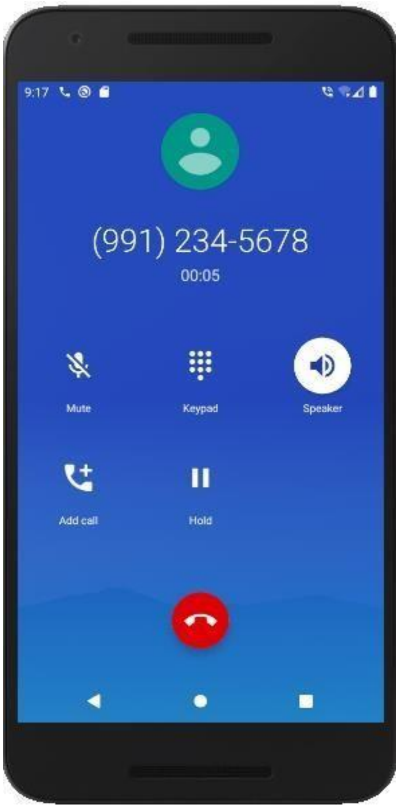
Sample Output



Part-A-Program-8

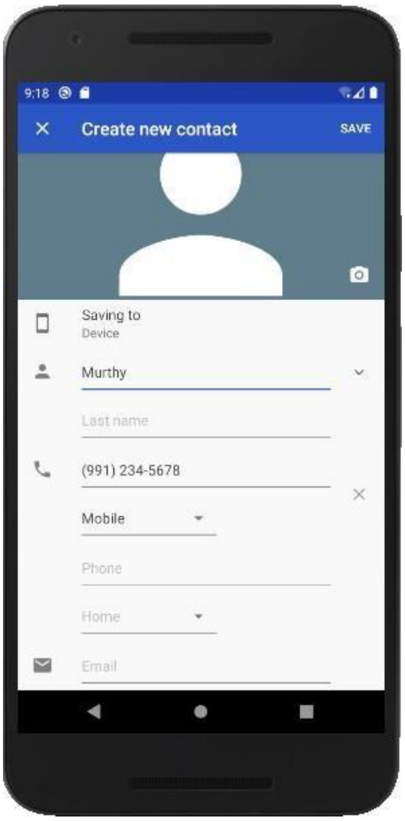
PHONE DAILER

9912345678



(991) 234-5678

00:05



9:18

Create new contact

SAVE

Saving to
Device

Murthy

Last name

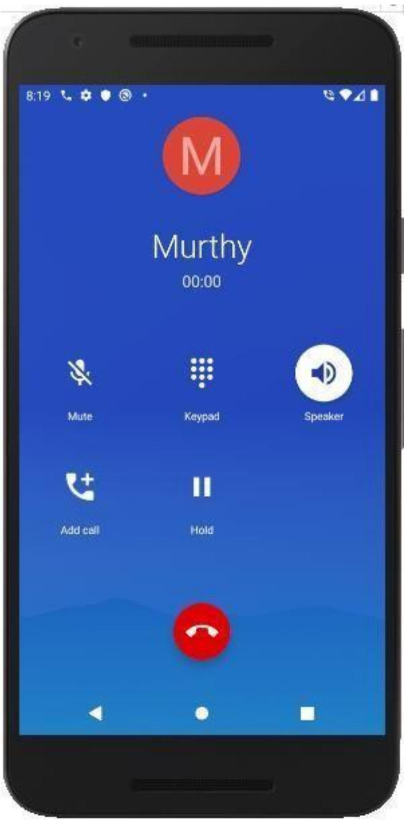
(991) 234-5678

Mobile

Phone

Home

Email



8:19

M

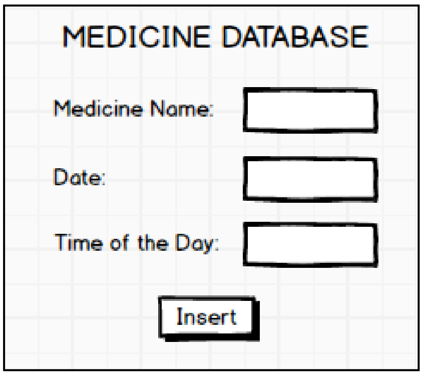
Murthy

00:00

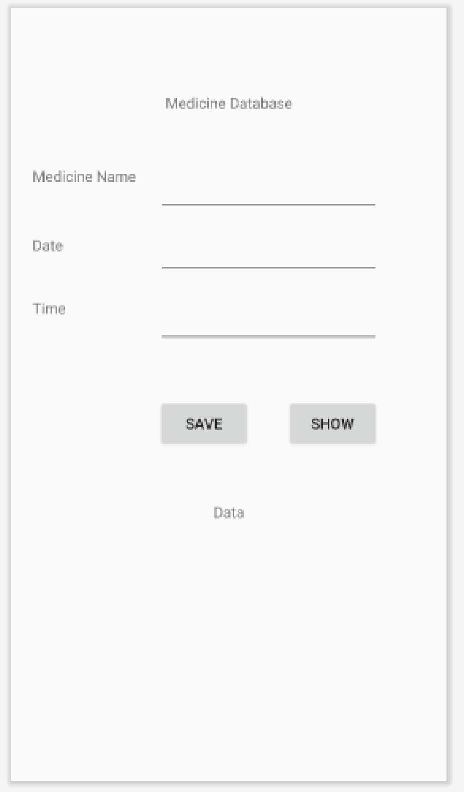
Additional Experiments

Program 1

Write a program to enter Medicine Name, Date and Time of the Day as input from the user and store it in the SQLite database. Input for Time of the Day should be either Morning or Afternoon or Evening or Night. Trigger an alarm based on the Date and Time of the Day and display the Medicine Name.



Design



activity_main.xml

```
<?xmlversion="1.0"encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayoutxmlns:android="http://schemas.androi
d.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="80dp"
android:text="MedicineDatabase"
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="20dp"
android:text="MedicineName"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="@+id/txt_medicine_name"/>


<TextView android:id="@+id/textView4"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="20dp"
android:text="Date"
app:layout_constraintBottom_toBottomOf="@+id/txt_date"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/txt_medicine_name"/>

<TextView android:id="@+id/textView5"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="20dp"
android:text="Time"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="@+id/txt_time"/>

<EditText
android:id="@+id/txt_medicine_name"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="20dp"
android:layout_marginTop="50dp"
android:ems="10"
android:inputType="textPersonName"
app:layout_constraintStart_toEndOf="@+id/textView3"
app:layout_constraintTop_toBottomOf="@+id/textView2"/>

<EditText
android:id="@+id/txt_date"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="15dp"
android:ems="10"
android:inputType="textPersonName"
app:layout_constraintStart_toStartOf="@+id/txt_medicine_name"
app:layout_constraintTop_toBottomOf="@+id/txt_medicine_name"/>
```

```
<EditText
android:id="@+id/txt_time"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="20dp"
android:ems="10"
android:inputType="textPersonName"
app:layout_constraintStart_toStartOf="@+id/txt_date"
app:layout_constraintTop_toBottomOf="@+id/txt_date"/>

<Button
android:id="@+id/btn_save"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="50dp"
android:text="Save"
app:layout_constraintStart_toStartOf="@+id/txt_time"
app:layout_constraintTop_toBottomOf="@+id/txt_time"/>


<Button
ndroid:id="@+id/btn_show"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="50dp"
android:text="Show"
app:layout_constraintEnd_toEndOf="@+id/txt_time"
app:layout_constraintTop_toBottomOf="@+id/txt_time"/>

<TextView
android:id="@+id/lbl_data"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="50dp"
android:text="Data"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/btn_save"/>
</androidx.constraintlayout.widget.ConstraintLayout>
```

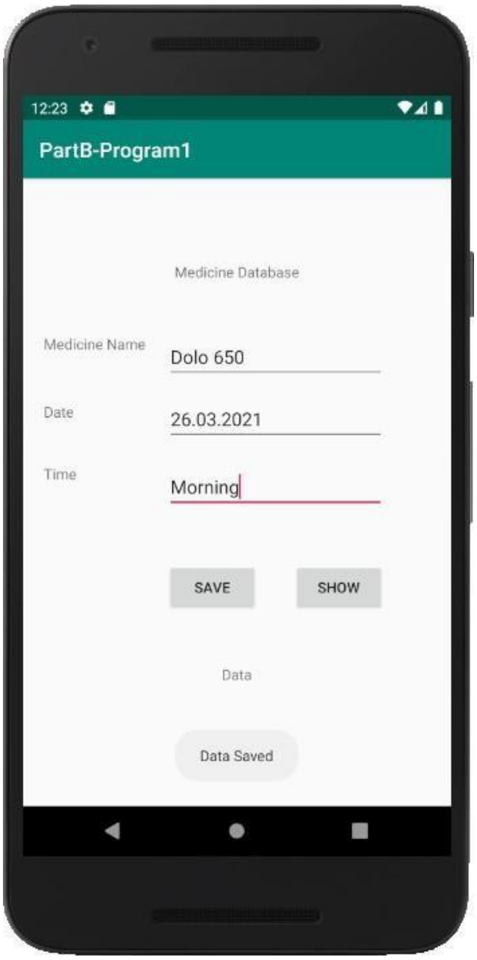
MyDatabase.java

```
packagecom.example.partb_program1;

importandroid.content.Context;
importandroid.database.sqlite.SQLiteDatabase;
importandroid.database.sqlite.SQLiteOpenHelper;

importandroidx.annotation.Nullable;

publicclassMyDatabaseextendsSQLiteOpenHelper{

publicstaticStringDATABASE_NAME="medicine.db";

publicMyDatabase(@NullableContextcontext,@NullableStringname,@Nullable
SQLiteDatabase.CursorFactoryfactory,intversion){
super(context,name,factory,version);
    }
@Override
publicvoidonCreate(SQLiteDatabasedb){

db.execSQL("CREATETABLEMEDICINE_NAMES(NAMETEXT,MDATETEXT,MTIMETEXT)");

    }

@Override
publicvoidonUpgrade(SQLiteDatabasedb,intoldVersion,intnewVersion){

    }
}
```

MainActivity.java

```
packagecom.example.partb_program1;

importandroidx.appcompat.app.AppCompatActivity;

importandroid.content.ContentValues;
importandroid.database.Cursor;
importandroid.database.sqlite.SQLiteDatabase;
importandroid.os.Bundle;
importandroid.view.View;
importandroid.widget.Button;
importandroid.widget.EditText;
importandroid.widget.TextView;
importandroid.widget.Toast;

importorg.w3c.dom.Text;

publicclassMainActivityextendsAppCompatActivityimplementsView.OnClickListener{

EditTexttxtMedicineName,txtDate,txtTime;
    ButtonbtnSave,btnShow;
TextViewlblData;

MyDatabasemyDatabase;

@Override
protectedvoidonCreate(BundlesavedInstanceState){
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
txtMedicineName=(EditText)findViewById(R.id.txt_medicine_name);
txtDate=(EditText)findViewById(R.id.txt_date);
txtTime=(EditText)findViewById(R.id.txt_time);

btnSave=(Button)findViewById(R.id.btn_save);
btnSave.setOnClickListener(this);
btnShow=(Button)findViewById(R.id.btn_show);
btnShow.setOnClickListener(this);
lblData=(TextView)findViewById(R.id.lbl_data);

myDatabase=newMyDatabase(getBaseContext(),
MyDatabase.DATABASE_NAME,null,1);

}

publicvoidonClick(Viewv)
{
if(v.equals(btnSave))
{
Stringmedicinename=txtMedicineName.getText().toString();
Stringdate=txtDate.getText().toString();
Stringtime=txtTime.getText().toString();

SQLiteDatabasedatabase=myDatabase.getWritableDatabase();
ContentValuescv=newContentValues();
cv.put("NAME",medicineName);
cv.put("MDATE",date);
cv.put("MTIME",time);



database.insert("MEDICINE_NAMES",null,cv);
Toast.makeText(getBaseContext(),"DataSaved",Toast.LENGTH_LONG).show();
}
elseif(v.equals(btnShow))
{

SQLiteDatabasedatabase=myDatabase.getReadableDatabase();
Cursorcursor=database.query("MEDICINE_NAMES",
newString[]{"NAME","MDATE","MTIME"},null,null,null,null,null);
lblData.setText("NAME\tDATE\tTIME\n");
while(cursor.moveToNext())
{
lblData.append(cursor.getString(0)+"\t");
lblData.append(cursor.getString(1)+"\t");
lblData.append(cursor.getString(2)+"\n");
        }
     }
  }
}
```

Sample Output



Medicine Database

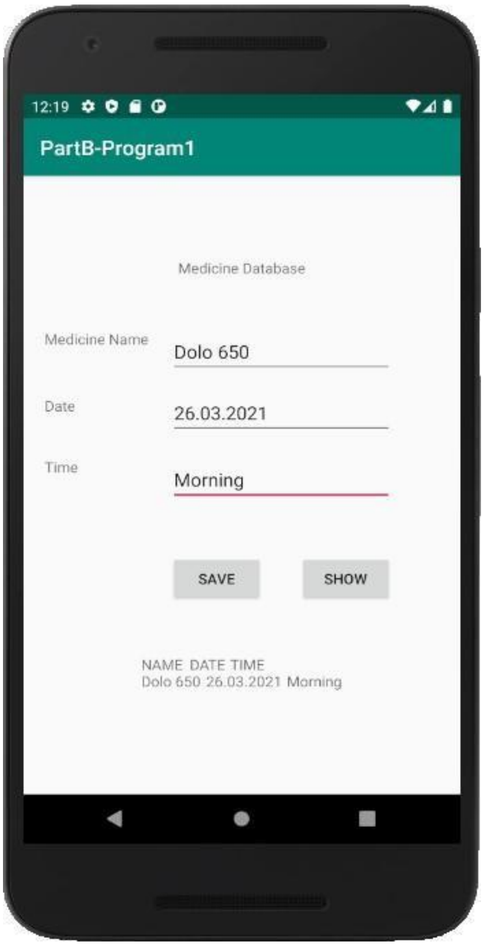
Medicine Name Dolo 650

Date 26.03.2021

Time Morning

Data

Data Saved



Medicine Database

Medicine Name Dolo 650

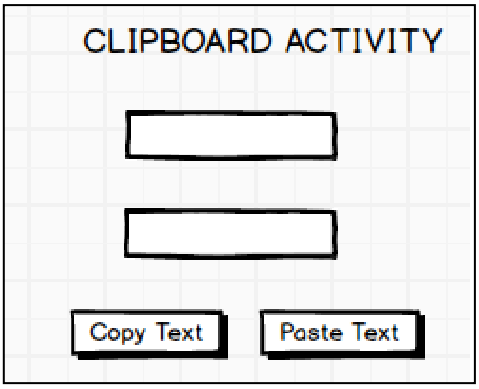
Date 26.03.2021

Time Morning

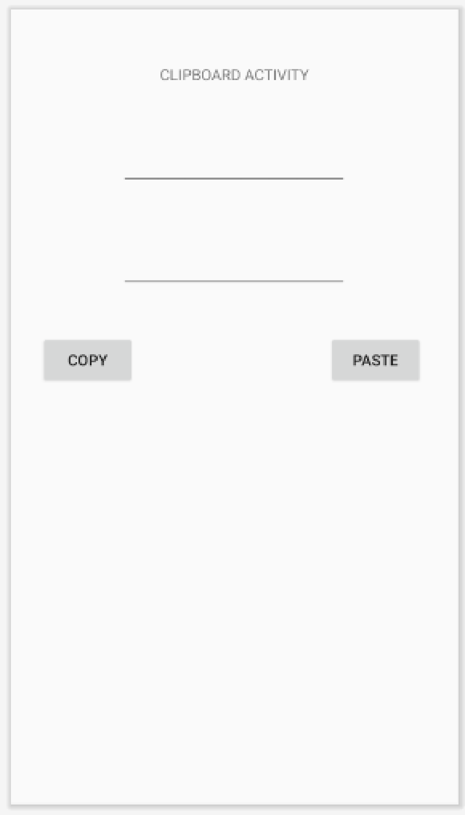
```
NAME  DATE TIME
Dolo 650 26.03.2021 Morning
```

Program 2

Develop an application that makes use of the clipboard framework for copying and pasting of thetext. The activity consists of two EditText controls and two Buttons to trigger the copy and pastefunctionality.



Design



activity_main.xml

```
<?xmlversion="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:id="@+id/layout"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">


<Button
android:id="@+id/btn_create"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="10dp"
android:layout_marginTop="40dp"
android:text="Create"
app:layout_constraintEnd_toStartOf="@+id/textView2"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/textView2"/>

<Button
android:id="@+id/btn_open"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="40dp"
android:layout_marginEnd="10dp"
android:text="Open"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toEndOf="@+id/textView2"
app:layout_constraintTop_toBottomOf="@+id/textView2"/>

<TextView android:id="@+id/textView2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="50dp"
android:text="FileApplication"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"/>

<EditText
android:id="@+id/txt_content"
android:layout_width="272dp"
android:layout_height="138dp"
android:layout_marginTop="50dp"
android:ems="10"
android:inputType="textPersonName"
app:layout_constraintTop_toBottomOf="@+id/btn_create"
tools:layout_editor_absoluteX="65dp"/>

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

```
android:layout_height="wrap_content"
android:layout_marginTop="50dp"
android:text="Save"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/txt_content"/>
</androidx.constraintlayout.widget.ConstraintLayout>
```

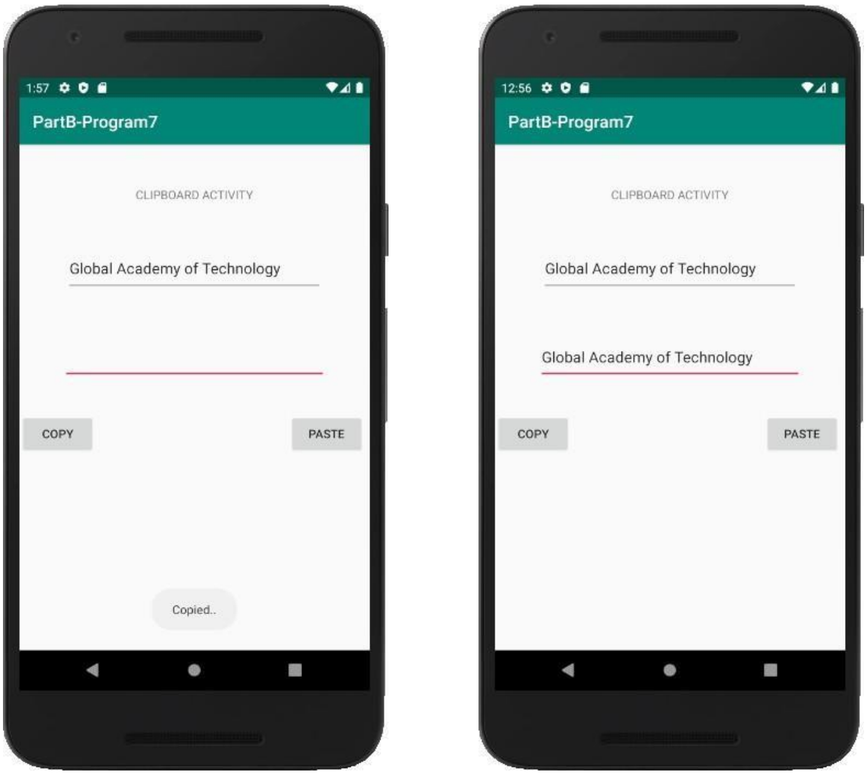
MainActivity.java

```
packagecom.example.partbprogram7;

importandroidx.appcompat.app.AppCompatActivity;
importandroid.content.ClipData;
importandroid.content.ClipboardManager;
importandroid.os.Bundle;
importandroid.view.View;
importandroid.widget.Button;
importandroid.widget.EditText;
importandroid.widget.Toast;

publicclassMainActivityextendsAppCompatActivityimplementsView.OnClickListener{
EditTexttxtCopy,txtPaste;
ButtonbtnCopy,btnPaste;

ClipboardManagermyClipboard;

@Override
protectedvoidonCreate(BundlesavedInstanceState){
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
txtCopy=(EditText)findViewById(R.id.txt_copy);
txtPaste=(EditText)findViewById(R.id.txt_paste);

btnCopy=(Button)findViewById(R.id.btn_copy);
btnCopy.setOnClickListener(this);

btnPaste=(Button)findViewById(R.id.btn_paste);
btnPaste.setOnClickListener(this);

myClipboard=(ClipboardManager)getSystemService(CLIPBOARD_SERVICE);
}

@Override
publicvoidonClick(Viewv){
if(v.equals(btnCopy))
{
ClipDatamyClip;
Stringdata=txtCopy.getText().toString();
myClip=ClipData.newPlainText("text",data);
myClipboard.setPrimaryClip(myClip);
Toast.makeText(getBaseContext(),"Copied..",Toast.LENGTH_LONG).show();
}
elseif(v.equals(btnPaste))
{
ClipDataabc=myClipboard.getPrimaryClip();
ClipData.Itemitem=abc.getItemAt(0);
txtPaste.setText(item.getText().toString());
     }
  }
}
```

AndroidManifest.xml

```
<?xmlversion="1.0"encoding="utf-8"?>
<manifestxmlns:android="http://schemas.android.com/apk/res/android"
package="com.example.partbprogram7">

<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">
<activityandroid:name=".MainActivity">
<intent-filter>
<actionandroid:name="android.intent.action.MAIN"/>
<categoryandroid:name="android.intent.category.LAUNCHER"/>
</intent-filter>
</activity>
</application>
</manifest>
```

Sample Output



PartB-Program7

CLIPBOARD ACTIVITY

Global Academy of Technology

Copied..

PartB-Program7

CLIPBOARD ACTIVITY

Global Academy of Technology

Global Academy of Technology

Program 3

Create an AIDL service that calculates Car Loan EMI. The formula to calculate EMI is

$$\mathbf{E = P * (r(1+r)^n)/((1+r)^n-1)}$$

where

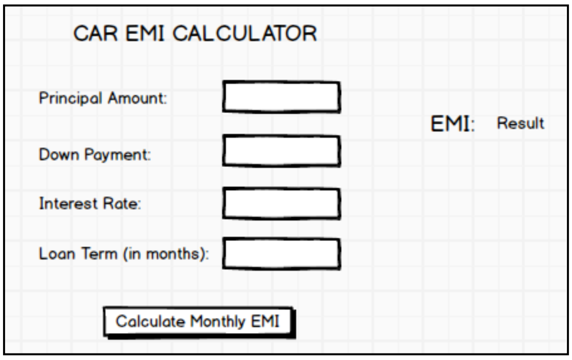
E = The EMI payable on the car loan amount

P = The Car loan Principal Amount

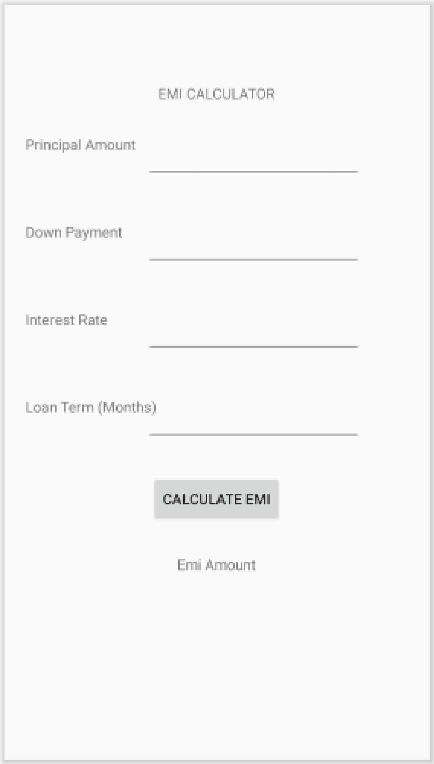
r = The interest rate value computed on a monthly basis

n = The loan tenure in the form of months

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



Design



activity_main.xml

```
<?xmlversion="1.0"encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayoutxmlns:android="http://schemas.androi
d.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/lblpayment"
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:text="EMICALCULATOR"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
tools:layout_editor_absoluteY="76dp"/>

<TextView android:id="@+id/textView2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="20dp"
android:layout_marginTop="30dp"
android:text="PrincipalAmount"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/textView"/>

<EditText
android:id="@+id/txt_principal"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="10dp"
android:layout_marginTop="30dp"
android:ems="10"
android:inputType="textPersonName"
app:layout_constraintStart_toEndOf="@+id/textView2"
app:layout_constraintTop_toBottomOf="@+id/textView"/>

<TextView
android:id="@+id/downpayment"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="DownPayment"
app:layout_constraintStart_toStartOf="@+id/textView2"
app:layout_constraintTop_toTopOf="@+id/txt_downnpayment"/>

<EditText
android:id="@+id/txt_downnpayment"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="40dp"
android:ems="10"
android:inputType="textPersonName"
app:layout_constraintStart_toStartOf="@+id/txt_principal"
```

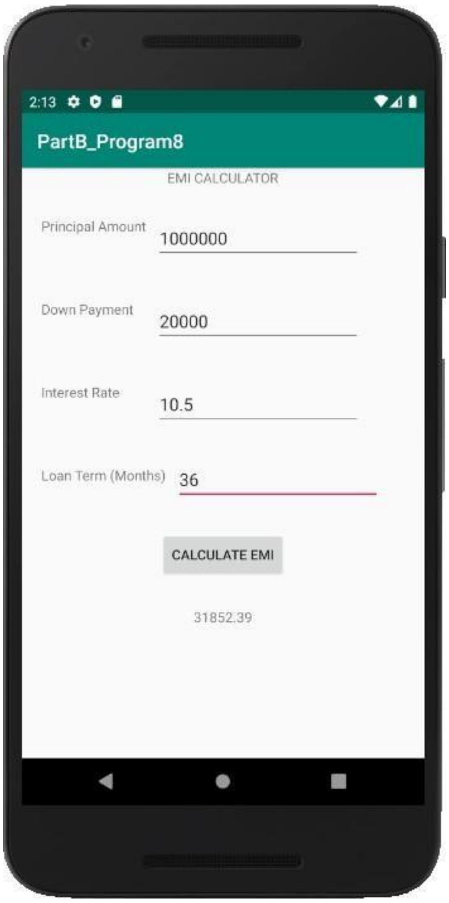
```
app:layout_constraintTop_toBottomOf="@+id/txt_principal"/>

<TextView android:id="@+id/textView4"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="InterestRate"
app:layout_constraintStart_toStartOf="@+id/downpayment"
app:layout_constraintTop_toTopOf="@+id/txt_interestrate"/>


<EditText
android:id="@+id/txt_interestrate"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="40dp"
android:ems="10"
android:inputType="textPersonName"
app:layout_constraintStart_toStartOf="@+id/txt_downnpayment"
app:layout_constraintTop_toBottomOf="@+id/txt_downnpayment"/>

<TextView
android:id="@+id/textView5"
android:layout_width="130dp"
android:layout_height="33dp"
android:layout_marginTop="8dp"
android:text="LoanTerm(Months)"
app:layout_constraintStart_toStartOf="@+id/textView4"
app:layout_constraintTop_toTopOf="@+id/txt_termmonths"/>

<EditText
android:id="@+id/txt_termmonths"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="20dp"
android:layout_marginTop="32dp"
android:ems="10"
android:inputType="textPersonName"
app:layout_constraintStart_toStartOf="@+id/txt_interestrate"
app:layout_constraintTop_toBottomOf="@+id/txt_interestrate"/>

<Button
android:id="@+id/btn_calculate"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="30dp"
android:text="CalculateEMI"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/txt_termmonths"/>

<TextView
android:id="@+id/lbl_emiamount"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="30dp"
android:text="EmiAmount"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
```

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

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

MainActivity.java

```
packagecom.example.partb_program8;

importandroidx.appcompat.app.AppCompatActivity;
importandroid.os.Bundle;
importandroid.view.View;
importandroid.widget.Button;
importandroid.widget.EditText;
importandroid.widget.TextView;
importandroid.widget.Toast;

importjava.text.DecimalFormat;
importjava.util.logging.SimpleFormatter;

publicclassMainActivityextendsAppCompatActivityimplementsView.OnClickListener{

EditTexttxtPrinicple,txtDownPayment,txtInterestRate,txtLoanTerm;

ButtonbtnCalculate;

TextViewlblResult;

@Override
protectedvoidonCreate(BundlesavedInstanceState){
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
txtPrinicple=(EditText)findViewById(R.id.txt_principal);
txtDownPayment=(EditText)findViewById(R.id.txt_downnpayment);
txtInterestRate=(EditText)findViewById(R.id.txt_interestrate);
txtLoanTerm=(EditText)findViewById(R.id.txt_termmonths);

btnCalculate=(Button)findViewById(R.id.btn_calculate);
btnCalculate.setOnClickListener(this);

lblResult=(TextView)findViewById(R.id.lbl_emiamount);

}

publicvoidonClick(Viewv)
{
try
{

DecimalFormatformatter=new
DecimalFormat("#0.00");

doubleprinicipleAmount=
Double.parseDouble(txtPrinicple.
getText().toString());
doubledownPayment=Double.parseDouble(txtDownPayment.getText().toString());
principleAmount=principleAmount-downPayment;
doubleinterestRate=Double.parseDouble(txtInterestRate.getText().toString());
interestRate=interestRate/(12*100);
doubleloanTerm=Double.parseDouble(txtLoanTerm.getText().toString());

doubleemi=principleAmount*
                    (interestRate*Math.pow((1+interestRate),loanTerm))
                    /(Math.pow((1+interestRate),loanTerm)-1);
lblResult.setText(String.valueOf(formatter.format(emi)));
}

catch(Exceptione)
{
Toast.makeText(getBaseContext(),"InvalidInput",Toast.LENGTH_LONG).show();
}
}
}
```

Sample Output



2:13

PartB_Program8

EMI CALCULATOR

Principal Amount: 1000000

Down Payment: 20000

Interest Rate: 10.5

Loan Term (Months): 36

31852.39

VIVA QUESTIONS

1. What is Android?
It is an open-sourced operating system that is used primarily on mobile devices, such as cell phones and tablets. It is a Linux kernel-based system that’s been equipped with rich components that allows developers to create and run apps that can perform both basic and advanced functions.</p>

2. What Is the Google Android SDK?
The Google Android SDK is a toolset that developers need in order to write apps on Android enabled devices. It contains a graphical interface that emulates an Android driven handheld environment, allowing them to test and debug their codes.

3. What is the Android Architecture?
Android Architecture is made up of 4 key components:

4. Describe the Android Framework.
The Android Framework is an important aspect of the Android Architecture. Here you can find all the classes and methods that developers would need in order to write applications on the Android environment.

5. What is AAPT?
AAPT is short for Android Asset Packaging Tool. This tool provides developers with the ability to deal with zip-compatible archives, which includes creating, extracting as well as viewing its contents.

6. What is the importance of having an emulator within the Android environment?
The emulator lets developers “play” around an interface that acts as if it were an actual mobile device. They can write and test codes, and even debug. Emulators are a safe place for testing codes especially if it is in the early design phase.

7. What is the use of an activityCreator?
An activity Creator is the first step towards the creation of a new Android project. It is made up of a shell script that will be used to create new file system structure necessary for writing codes within the Android IDE.

8 . Describe Activities.
Activities are what you refer to as the window to a user interface. Just as you create windows in order to display output or to ask for an input in the form of dialog boxes, activities play the same role, though it may not always be in the form of a user interface.

9. What are Intents?
Intents displays notification messages to the user from within the Android enabled device. It can be used to alert the user of a particular state that occurred. Users can be made to respond to intents.

10. Differentiate Activities from Services.
Activities can be closed, or terminated anytime the user wishes. On the other hand, services are designed to run behind the scenes, and can act independently. Most services run continuously, regardless of whether there are certain or no activities being executed.

11. What items are important in every Android project?
These are the essential items that are present each time an Android project is created:

Android Manifest.xml
build.xml
bin/
src/
res/
assets/

12. What is the importance of XML-based layouts?
The use of XML-based layouts provides a consistent and somewhat standard means of setting GUI definition format. In common practice, layout details are placed in XML files while other items are placed in source files.

13. What are containers?
Containers, as the name itself implies, holds objects and widgets together, depending on which specific items are needed and in what particular arrangement that is wanted. Containers may hold labels, fields, buttons, or even child containers, as examples.

14. What is Orientation?
Orientation, which can be set using set Orientation(), dictates if the Linear Layout is represented as a row or as a column. Values are set as either HORIZONTAL or VERTICAL.

15. What is the importance of Android in the mobile market?
Developers can write and register apps that will specifically run under the Android environment. This means that every mobile device that is Android enabled will be able to support and run these apps. With the growing popularity of Android mobile devices, developers can take advantage of thistrend by creating and uploading their apps on the Android Market for distribution to anyone who wants to download it.

16. What do you think are some disadvantages of Android?
Given that Android is an open-source platform, and the fact that different Android operatingsystems have been released on different mobile devices, there’s no clear cut policy to how applications can adapt with various OS versions and upgrades.
–One app that runs on this particular version of Android OS may or may not run on another version.
–Another disadvantage is that since mobile devices such as phones and tabs come in different sizes and forms, it poses a challenge for developers to create apps that can adjust correctly to the right screen size and other varying features and specs.

17. What is adb?
Adb is short for “Android Debug Bridge”. It allows developers the power to execute remote shell commands. Its basic function is to allow and control communication towards and from the emulator port.

18. What are the four essential states of an activity?
Active – if the activity is at the foreground
Paused – if the activity is at the background and still visible
Stopped – if the activity is not visible and therefore is hidden or obscured by another activity
Destroyed – when the activity process is killed or completed terminated

19. What is ANR?
ANR is short for Application Not Responding. This is actually a dialog that appears to the user whenever an application have been unresponsive for a long period of time.

20. Which elements can occur only once and must be present?
Among the different elements, the and elements must be present and can occur only once. The rest are optional, and can occur as many times as needed.

21. How are escape characters used as attribute?
Escape characters are preceded by double backslashes. For example, a newline character is created using

22. What is the importance of settings permissions in app development?
Permissions allow certain restrictions to be imposed primarily to protect data and code. Without these, codes could be compromised, resulting to defects in functionality.

23. What is the function of an intent filter?
Because every component needs to indicate which intents they can respond to, intent filters are used to filter out intents that these components are willing to receive. One or more intent filters are possible, depending on the services and activities that is going to make use of it

24. Enumerate the three key loops when monitoring an activity?
Entire lifetime – activity happens between on Create and on Destroy
Visible lifetime – activity happens between on Start and on Stop
Foreground lifetime – activity happens between on Resume and on Pause

25. When is the on Stop(. method invoked?
A call to on Stop method happens when an activity is no longer visible to the user, either because another activity has taken over or if in front of that activity.

26. Is there a case wherein other qualifiers in multiple resources take precedence over locale?
Yes, there are actually instances wherein some qualifiers can take precedence over locale. There are two known exceptions, which are the MCC (mobile country code. and MNC (mobile network code. qualifiers.

Reference Books

1. Google Developer Training, "Android Developer Fundamentals Course – Concept Reference", Google Developer Training Team, 2017. https://www.gitbook.com/book/google-developer-training/android-developer- fundamentals-course-concepts/details
   (Download pdf file from the above link)
2. Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, Wiley IndiaPvt Ltd, 2014. ISBN-13:978-8126547197
3. Dawn Griffiths and David Griffiths, "Head First Android Development", 1st Edition, O"Reilly SPD Publishers, 2015. ISBN-13:978-9352131341
4. BillPhillips,ChrisStewartandKristinMarsicano, "AndroidProgramming:TheBigNerd Ranch Guide", 3rd Edition, Big Nerd Ranch Guides, 2017. ISBN-13:978-0134706054