

**MOTTO**

*“Hand In Hand We Learn”*

**MISSION**

*To nurture and transform the raw students to a highly skilled, knowledgeable, technically resourceful engineer with grace and character to take on the forth coming global challenges.*

**VISION**

*To evolve skilled & value based resource professionals, to provide opportunities to the students without any discrimination, to find space for exponential growth of personality & character, education & empowerment and to face the challenges of tomorrow successfully through quality education..*

**MOBILE APPLICATION AND DEVELOPMENT**

**(18CSMP68)**

(As per Visvesvaraya Technological University Syllabus)

Compiled By:

Asst. Professor

Dept. of CSE

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

2022-2023

**Course Name: Mobile Application Development**

**Course Code: 18CSMP68**

**Course prerequisite: Core Java**

* **Laboratory Objectives:**

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

* **Laboratory Outcomes:**   
  After studying these laboratory programs, students will be able to  
  **CO1**: Create, test and debug Android application by setting up Android development  
  environment.  
  **CO2**: Implement adaptive, responsive user interfaces that work across a wide range of devices.  
  **CO3:** Infer long running tasks and background work in Android applications.  
  **CO4**: Demonstrate methods in storing, sharing and retrieving data in Android applications.

**CO5**: Infer the role of permissions and security for Android applications

* **PROGRAM SPECIFIC OUTCOMES (PSO’S)**  
  **PSO1**: Nurture critical and reflective thinking on fundamental issues and problems related to  
  information.  
  **PSO2:** Understand the processes to support the delivery and management of information systems  
  with an appropriate application environment

### CO-PO-PSO Mapping

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO**  **No.** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| **CO1** | 3 | 3 | 3 |  | 3 |  |  | 2 | 3 | 2 | 2 | 1 | 2 | 3 |
| **CO2** | 3 | 3 | 3 |  | 3 |  |  | 2 | 3 | 2 | 2 | 1 | 2 | 3 |
| **CO3** | 3 | 3 | 3 |  | 3 |  |  | 2 | 3 | 2 | 2 | 1 | 2 | 3 |
| **CO4** | 3 | 3 | 3 |  | 3 |  |  | 2 | 3 | 2 | 2 | 1 | 2 | 3 |
| **CO5** | 3 | 3 | 3 |  | 3 |  |  | 2 | 3 | 2 | 2 | 1 | 2 | 3 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MOBILE APPLICATION DEVELOPMENT**  **(Effective from the academic year 2018 -2019) SEMESTER – VI** | | | | |
| **Course Code** | | **18CSMP68** | **IA Marks** | 40 |
| **Number of Contact Hours/Week** | | 0:0:2 | **Exam Marks** | 60 |
| **Total Number of Contact Hours** | | 3 Hours/Week | **Exam Hours** | 03 |
| **CREDITS – 02** | | | | |
| **Laboratory Objectives:**Thislaboratory (18CSMP68) will enable students to | | | | |
| * Learn and acquire the art of Android Programming. * ConfigureAndroid studio to run the applications. * Understand and implement Android's User interface functions. * Create, modify and query on SQlite database. * Inspect different methods of sharing data using services. | | | | |
| **Descriptions (if any):** | | | | |
| 1. The installation procedure of the Android Studio/Java software must be demonstrated and carried out in groups. 2. Students should use the latest version of Android Studio/Java/ Kotlin to execute these programs. Diagrams given are for representational purposes only, students are expected to improvise on them. 3. **Part B programs should be developed as an application and are to be demonstrated as a mini project in a group by adding extra features or the students can also develop their application**   **and demonstrate it as a mini-project. (Projects/programs are not limited to the list given in Part B).** | | | | |
| **Programs List:** | | | | |
| **PART – A** | | | | |
| **1** | Create an application to design aVisiting Card. The Visiting card should havea companylogoatthe 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 isto be displayed. Insert a horizontal line between the job title and the phone number. | | | |
| **2** | Develop an Android application usingcontrols like Button, TextView, EditText for designing a  calculatorhaving basic functionality like Addition, Subtraction, Multiplication,andDivision. | | | |

|  |  |
| --- | --- |
|  |  |
| **3** | Create a SIGN Up activity with Username and Password. Validation of password should happen based on the following rules:   * Password should contain uppercase and lowercase letters. * Password should contain letters and numbers. * Password should contain special characters. * Minimum length of the password (the default value is 8).   On successful **SIGN UP** proceed to the next Login activity. Here the user should **SIGN IN** using the Username and Password created during signup activity. If the Username and Password are matched then navigate to the next activity whichdisplays a message saying “Successful Login” or else display a toast message saying “Login Failed”.The user is given only two attempts and after thatdisplay a toast message saying “Failed Login Attempts” and disable the SIGN IN button. Use Bundle to transfer information from one activity to another. |
| **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. |
| **5** | Write a program to create an activity with two buttons START and STOP. On  pressingoftheSTART 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. |
| **6** | Create two files of XML and JSON type with values for City\_Name, Latitude, Longitude, Temperature,andHumidity. Develop an application to create an activity with two buttons to parse the XML and JSON files which when clicked should display the data in their respective layouts side by side. |
| **7** | Develop a simple application withoneEditTextso 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. |
| **8** | Create an activity like a phone dialer withCALLand 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** | |
| **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 Eveningor Night. Trigger an alarm based on the Date and Time of the Day and display the Medicine Name. |
| **2** | Develop a content provider application with an activity called “Meeting Schedule” which takes Date, Time and Meeting Agenda as input from the user and store this information into the SQLite database. Create another application with an activity called “Meeting Info” having DatePicker  control, which on the selection of a date should display the Meeting Agenda information for that particular date, else it should display a toast message saying “No Meeting on this Date”. |
|  |  |

|  |  |
| --- | --- |
| **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. |
| **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 “First Create a File”. |
| **5** | Create an application to demonstrate a basic media playerthat 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. |

|  |  |
| --- | --- |
|  |  |
| **6 6** | Develop an application to demonstrate the use of Asynchronous tasks in android. The asynchronous task should implement the functionality of a simple moving banner. On pressing the **Start Task** button, the banner message should scrollfrom right to left. On pressing the **Stop Task** button, the banner message should stop.Let the banner message be “Demonstration of Asynchronous Task”. |
| **77 7** | Develop an application that makes use of the clipboard framework for copying and pasting of the text. The activity consists of two EditText controls and two Buttons to trigger the copy and paste functionality. |
| **88 8** | Create an AIDL service that calculates Car Loan EMI. The formula to calculate EMI is  **E = P \* (r(1+r)n)/((1+r)n-1)**  where  E = The EMI payable on the car loan amount P = The Car loan 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 the Car. Develop an application that makes use of this AIDL service to calculate the EMI. This application should have four EditText to read the 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 and interest Rate values. |

|  |  |
| --- | --- |
|  |  |
| **Laboratory Outcomes:**After studying theselaboratory programs, students will be able to | |
| * Create, test and debug Android application by setting up Android development environment. * Implement adaptive, responsive user interfaces that work across a wide range of devices. * Infer long running tasks and background work in Android applications. * Demonstrate methods in storing, sharing and retrieving data in Android applications. * Infer the role of permissions and security for Android applications. | |
| **Procedure to Conduct Practical Examination** | |
| * Experiment distribution   + For laboratories having only one part: Students are allowed to pick oneexperiment from the lot with equal opportunity.   + For laboratories having PART A and PART B: Students are allowed to pick oneexperiment from PART A and one experiment from PART B, with equalopportunity. * Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only. * Marks Distribution (Courseed to change in accordance with university regulations)   + For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15= 100 Marks   + For laboratories having PART A and PART B     1. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks     2. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks | |
| **Text 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-](https://www.gitbook.com/book/google-developer-training/android-developer-fundamentals-course-concepts/details) [course-concepts/details](https://www.gitbook.com/book/google-developer-training/android-developer-fundamentals-course-concepts/details)  (Download pdf file from the above link) | |
| **Reference Books:** | |
| 1. Erik Hellman, **“Android Programming – Pushing the Limits”,** 1st Edition, Wiley India Pvt Ltd, 2014. ISBN-13: 978-8126547197 2. Dawn Griffiths and David Griffiths, **“Head First Android Development”,** 1st Edition, O‟Reilly SPD Publishers, 2015. ISBN-13: 978-9352131341 3. Bill Phillips, Chris Stewart and Kristin Marsicano, **“Android Programming: The Big Nerd**   **Ranch Guide”,** 3rd Edition, Big Nerd Ranch Guides, 2017. ISBN-13: 978-0134706054 | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SI. NO** | **PROGRAM** | **CO** | **RBT** | **PAGE NO.** |
| 1 | Installing Android Studio and Packages |  | L2 | 1 |
| 2 | 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. | CO1  to CO5 | L3 |  |
| 3 | 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. | CO1  to CO5 | L3 |  |
| 4 | Program 3:  Create a SIGN Up activity with Username and Password. Validation of password should happen based on the following rules:   * Password should contain uppercase and lowercase letters. * Password should contain letters and numbers. * Password should contain special characters. * Minimum length of the password (the default value is 8). | CO1  to CO5 | L3 |  |
| 5 | 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 | CO1  to CO5 | L3 |  |
| 6 | 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 | CO1  to CO5 | L3 |  |

**INDEX**

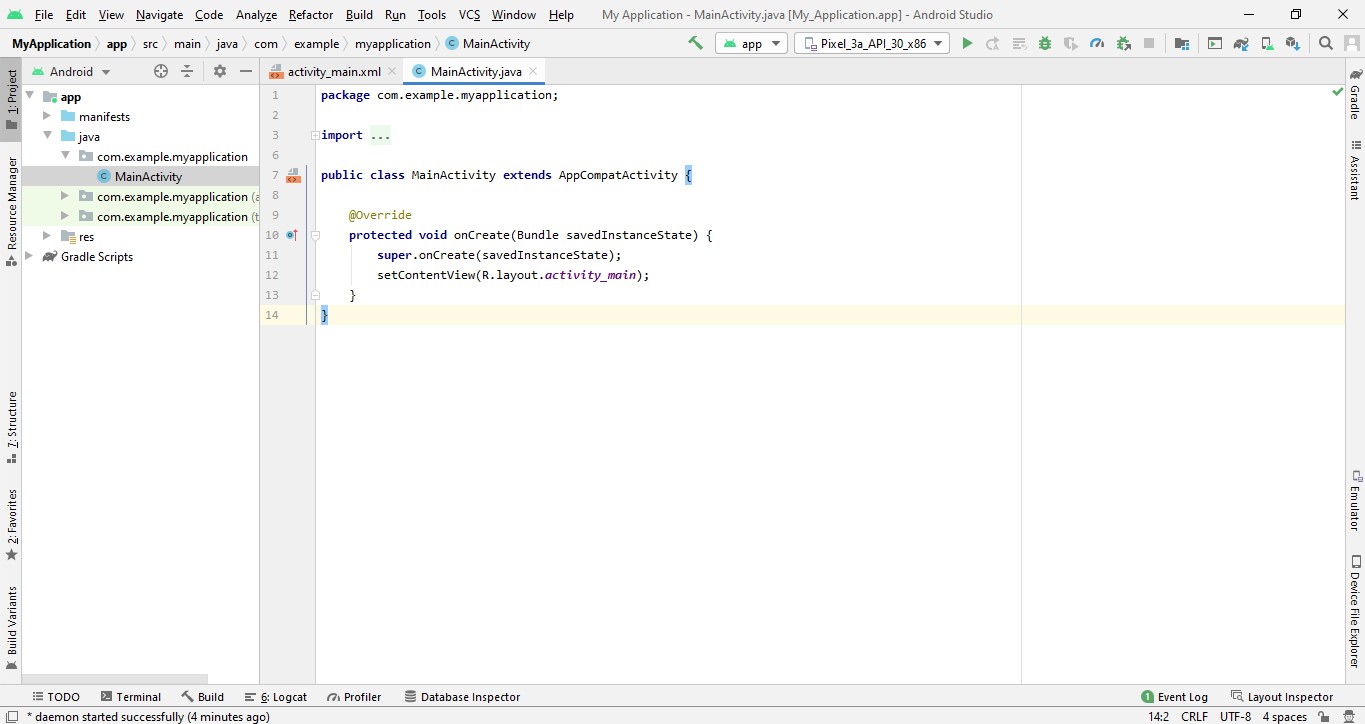
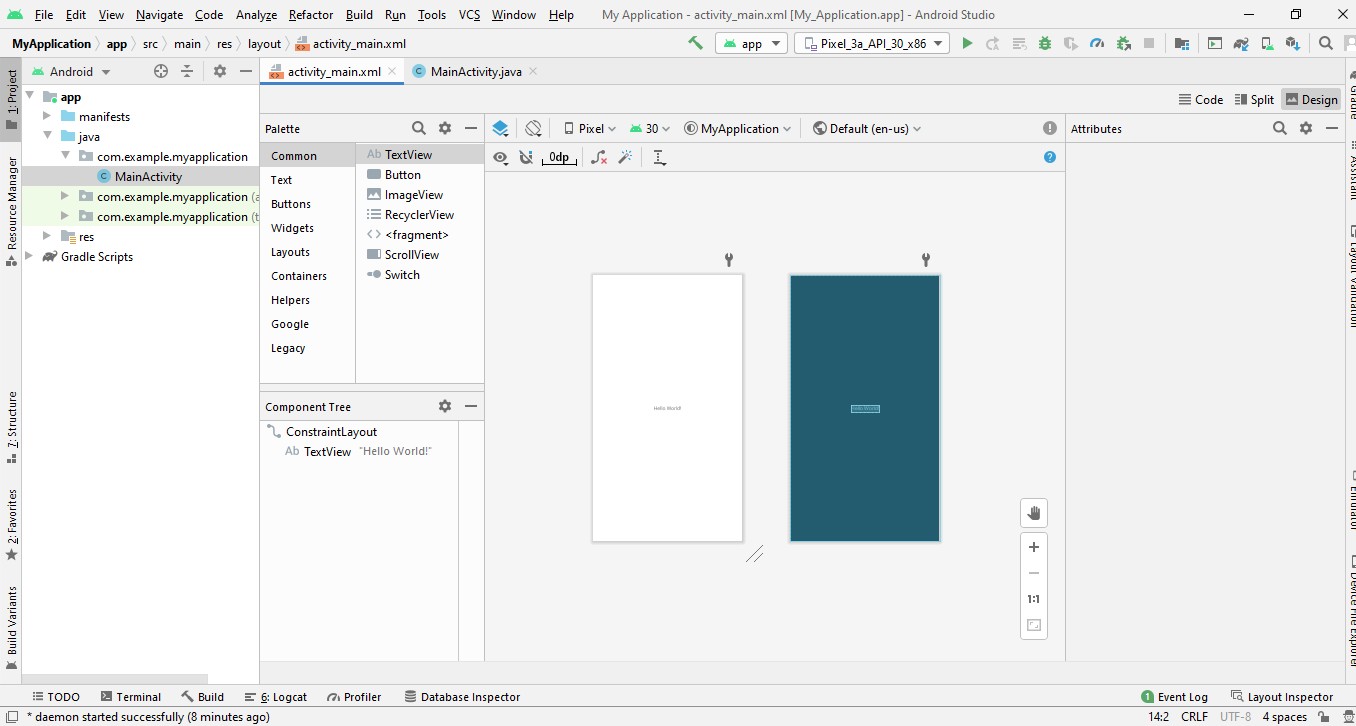
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | until the STOP button is pressed. Display the counter L3 |  |  |  |
| 7 | Program 6:  Create two files of XML and JSON type with values for City\_Name, Latitude, Longitude, Temperature and Humidity. Develop an application to create an activity with two buttons to parse the XML and JSON files which when clicked should display the data in their respective  layouts side by side. | CO1  to CO5 | L3 |  |
| 8 | 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. | CO1  to CO5 | L3 |  |
| 9 | 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. | CO1  to CO5 | L3 |  |
| 10 | Viva questions | - | L2 |  |

## Installing Android Studio and Packages

### Download Android Version 4.0.2 from the below link

[https://redirector.gvt1.com/edgedl/android/studio/install/4.0.2.0/android-studio-ide- 193.6821437-](https://redirector.gvt1.com/edgedl/android/studio/install/4.0.2.0/android-studio-ide-193.6821437-windows.exe) [windows.exe](https://redirector.gvt1.com/edgedl/android/studio/install/4.0.2.0/android-studio-ide-193.6821437-windows.exe)

* **Once installed successfully will be able to see following screen**



### Explore the project structure

In this practical, you will explore how the project files are organized in Android Studio.

These steps assume that your Hello World project starts out as shown in screenshots in page 1.

### Explore the project structure and layout

In the **Project > Android** view of your previous task, there are three top-level folders below your **app** folder:

**manifests**, **java**, and **res**.

1. Expand the **manifests** folder.

This folder contains **AndroidManifest.xml.** This file describes all of the components of your Android app and is read by the Android run-time system when your program is executed.

1. Expand the **java** folder.

All your Java language files are organized in this folder. The **java** folder contains three subfolders:

* + **com.example.hello.helloworld (or the domain name you have specified):** All the files for a package are in a folder named after the package. For your Hello World application, there is one package and it only contains MainActivity.java (the file extension may be omitted in the Project view).
  + **com.example.hello.helloworld(androidTest):** This folder is for your instrumented tests, and starts out with a skeleton test file.
  + **com.example.hello.helloworld(test):** This folder is for your unit tests and starts out with an automatically created skeleton unit test file.

1. Expand the **res** folder. This folder contains all the resources for your app, including images, layout files, strings, icons, and styling. It includes these subfolders:
   * **drawable**. Store all your app's images in this folder.
   * **layout**. Every activity has at least one layout file that describes the UI in XML. For Hello World, this folder contains activity\_main.xml.
   * **mipmap**. Store your launcher icons in this folder. There is a sub-folder for each supported screen density. Android uses the screen density, that is, the number of pixels per inch to determine the required image resolution. Android groups all actual screen densities into generalized densities, such as medium (mdpi), high (hdpi), or extra-extraextra-high (xxxhdpi). The ic\_launcher.png folder contains the default launcher icons for all the densities supported by your app.
   * **values**. Instead of hardcoding values like strings, dimensions, and colors in your XML and Java files, it is best practice to define them in their respective values file. This makes it easier to change and be consistent across your app.
2. Expand the **values** subfolder within the res folder. It includes these subfolders:
   * **colors.xml**. Shows the default colors for your chosen theme, and you can add your own colors or change them based on your app's requirements.
   * **dimens.xml**. Store the sizes of views and objects for different resolutions.
   * **strings.xml**. Create resources for all your strings. This makes it easy to translate them to other languages.
   * **styles.xml**. All the styles for your app and theme go here. Styles help give your app a consistent look for all UI elements.

### Create a virtual device (emulator)

* In this task, you will use the Android Virtual Device (AVD) manager to create a virtual device or emulator that simulates the configuration for a particular type of Android device.
* Using the AVD Manager, you define the hardware characteristics of a device and its API level, and save it as a virtual device configuration.
* When you start the Android emulator, it reads a specified configuration and creates an emulated device that behaves exactly like a physical version of that device , but it resides on your computer .
* **Why:** With virtual devices, you can test your apps on different devices (tablets, phones) with different API levels to make sure it looks good and works for most users. You do not need to depend on having a physical device available for app development.

### Create a virtual device

In order to run an emulator on your computer, you have to create a configuration that describes the virtual device.

1. In Android Studio, select **Tools > Android > AVD Manager**, or click the **AVD Manager ** icon in the toolbar.
2. If you have created a virtual device before, the window shows all of your existing devices and the button is at the bottom. Otherwise follow step 3
3. Click the **+Create Virtual Device….**

The Select Hardware screen appears showing a list of preconfigured hardware devices. For each device, the table shows its diagonal display size (Size), screen resolution in pixels (Resolution), and pixel density (Density). For example the Nexus 5 device, the pixel density is xxhdpi, which means your app uses the launcher icons in the xxhdpi folder of the mipmap folder. Likewise, your app will use layouts and drawables from folders defined for that density as well.

1. Choose the for example Nexus 5 hardware device and click **Next**.
2. On the **System Image** screen, from the **Recommended** tab, choose which version of the Android system to run on the virtual device. You can select the latest system image. There are many more versions available than shown in the **Recommended** tab. Look at the **x86 Images** and **Other**

**Images** tabs to see them.

1. If a **Download** link is visible next to a system image version, it is not installed yet, and you need to download it. Ifbnecessary, click the link to start the download, and click **Finish** when it's done.

**7**. On **System Image** screen, choose a system image and click **Next**.

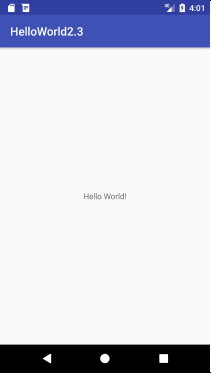
**8**. Verify your configuration, and click **Finish**. (If the **Your Android Devices** AVD Manager window stays open, you can go ahead and close it.)

### Run your app on an emulator

1. In Android Studio, select **Run > Run app** or click the **Run icon **in the toolbar.
2. In the **Select Deployment Target** window, under **Available Emulators**, select **Nexus 5 API 23** and click

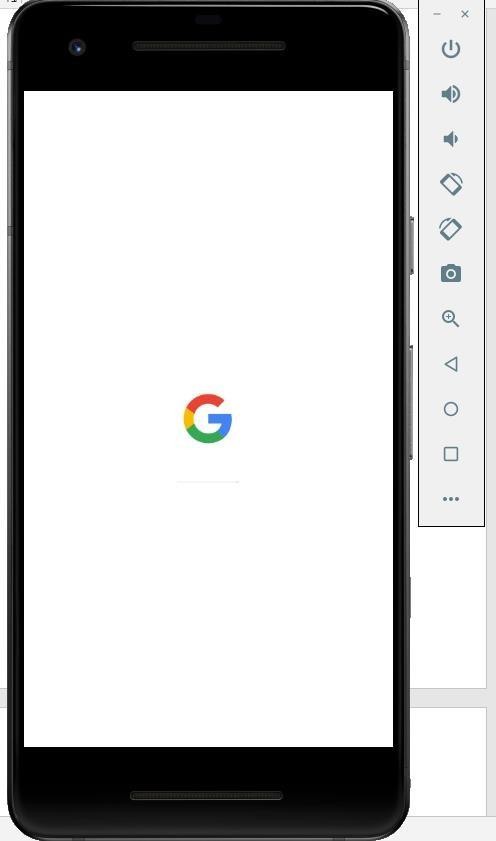
**OK**.

* + The emulator starts and boots just like a physical device. Depending on the speed of your computer, this may take a while.
  + Your app builds, and once the emulator is ready, Android Studio will upload the app to the emulator and run it.
  + You should see the Hello World app as shown in the following screenshot.



**Nexus 5 API 23** Emulator showing output for hello world program.

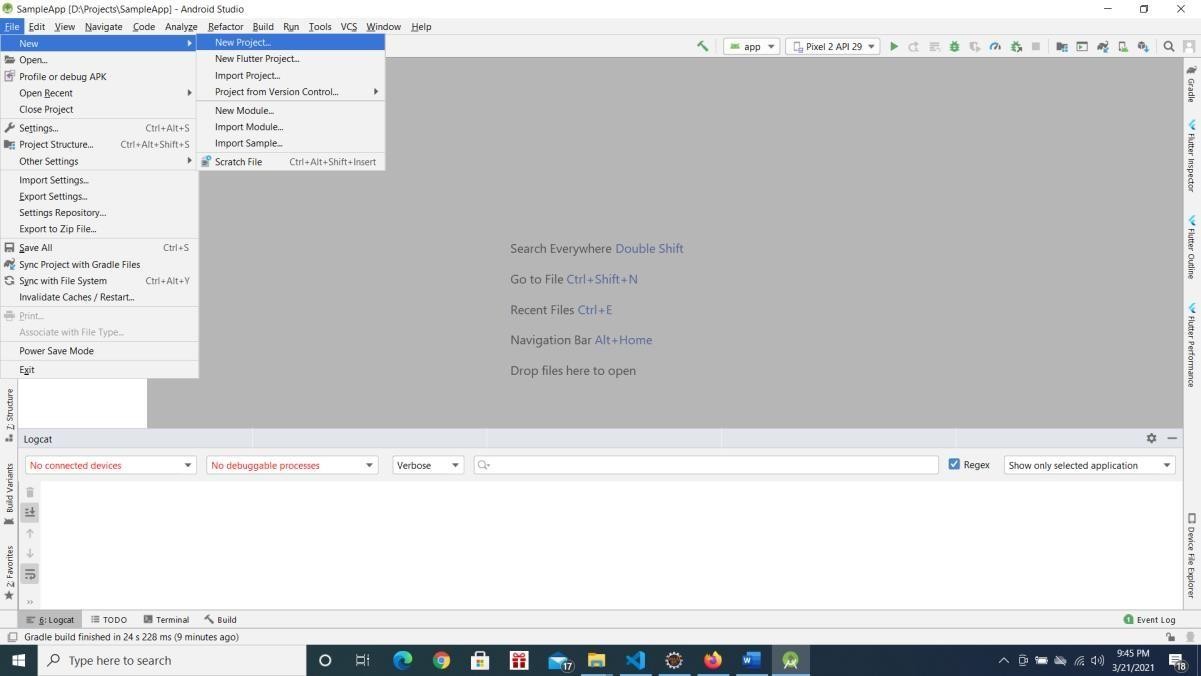
* If Pixel\_3a\_Api\_30\_x86 emulator is installed , then the emulator would look as shown below,



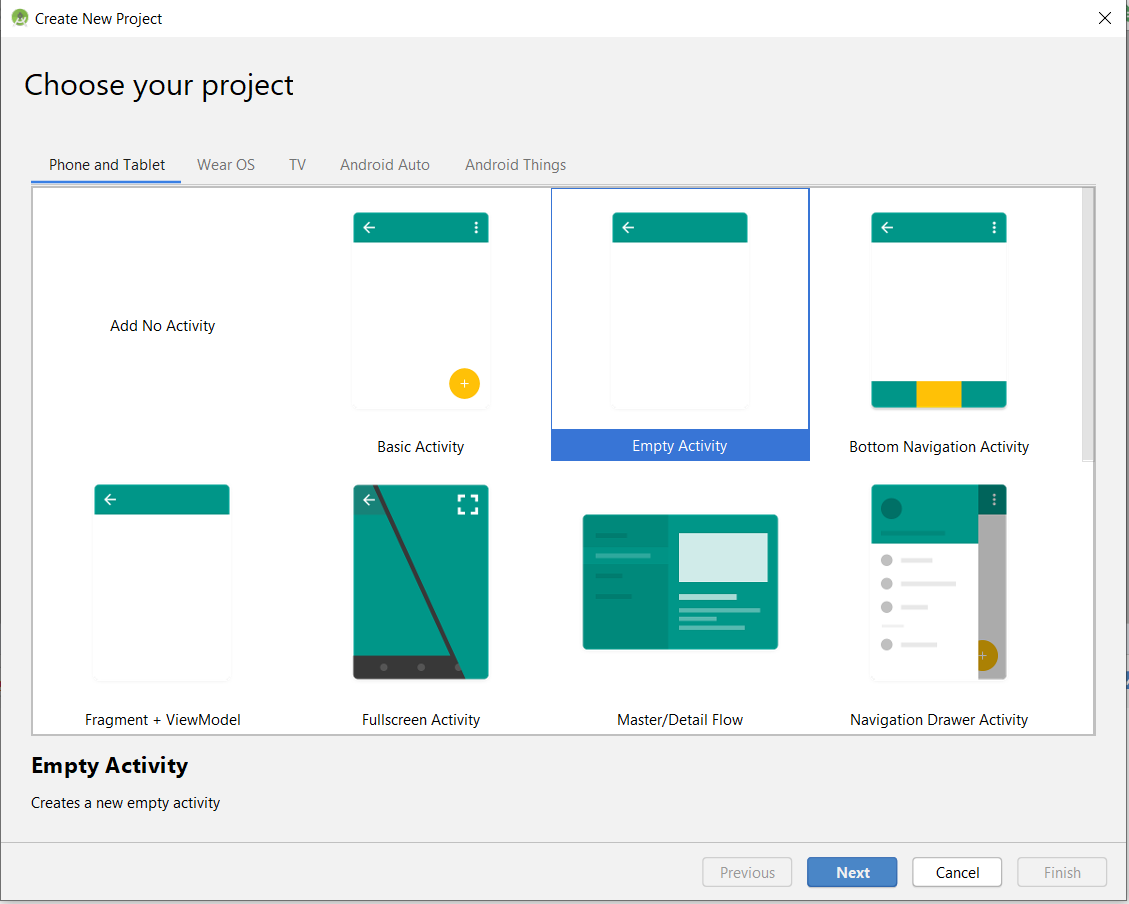
### 1.5 Creating a New Project in Android

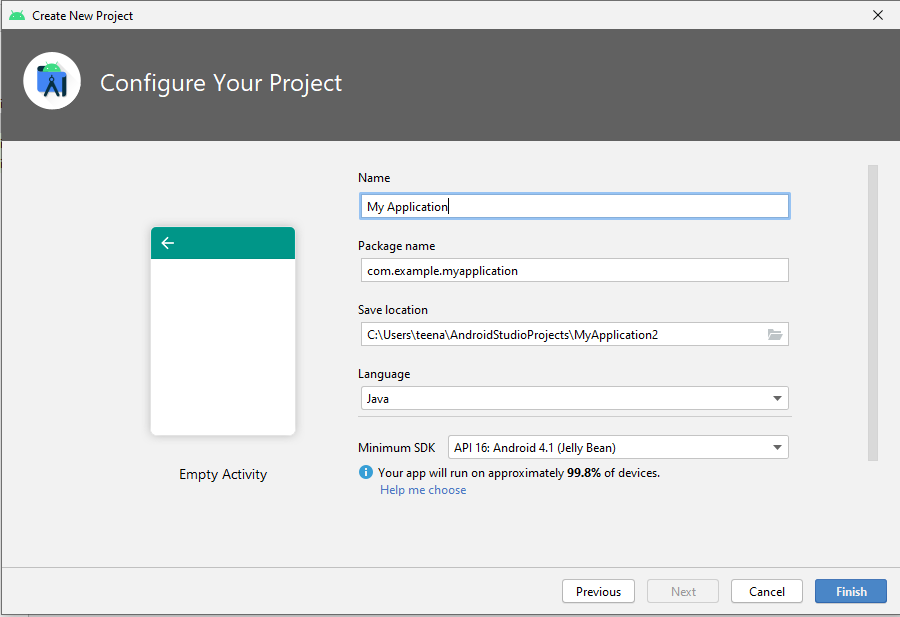
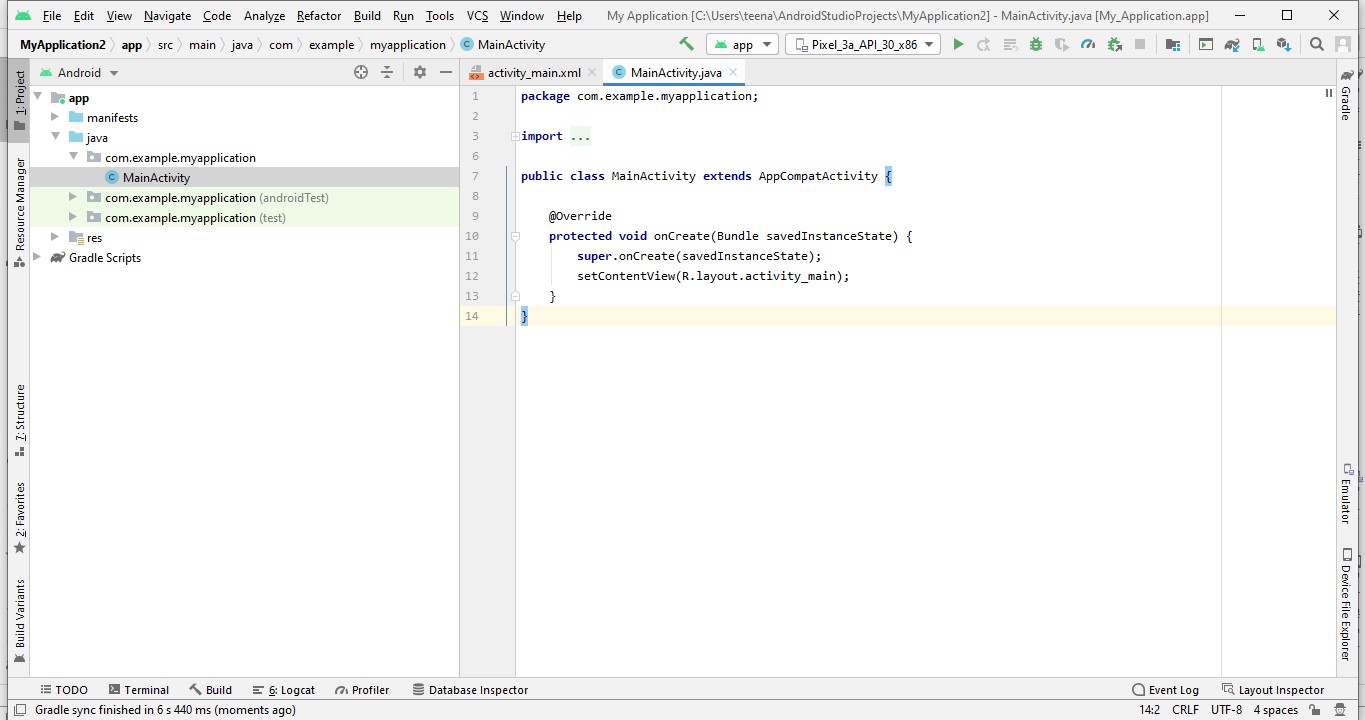
While creating a New Project for First Time, make sure you have connected to internet, It downloads the required packages from internet.

1. Go to File next click **New** then **New Project,** you will get the following screen



2 Choose **Phone and Tablet** tab and select **Empty Activity** and press **Next** as show in below screenshot



1. **Configure your Project** window will open, enter below details and press **Finish** button.
2. . After pressing **Finish** button you will get the following screen, and you can start developing the app.

**“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 centre. Information like the name of the employee, job title, phone number, address, email, fax and the website address are to be displayed. Insert a horizontal line between the job title and the 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.

* **XML –CODE:**

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity2">

<TextView

android:id="@+id/textView2"

android:layout\_width="216dp"

android:layout\_height="64dp"

android:layout\_marginStart="24dp"

android:layout\_marginTop="16dp"

android:gravity="center"

android:text="Seacollege"

android:textAllCaps="true"

android:textSize="24sp"

android:textStyle="bold"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<ImageView

android:id="@+id/imageView"

android:layout\_width="160dp"

android:layout\_height="117dp"

android:layout\_marginStart="5dp"

android:layout\_marginTop="16dp"

android:layout\_marginEnd="6dp"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toEndOf="@+id/textView2"

app:layout\_constraintTop\_toTopOf="parent"

app:srcCompat="@mipmap/sea" />

<View

android:id="@+id/view"

android:layout\_width="402dp"

android:layout\_height="2dp"

android:layout\_marginStart="13dp"

android:layout\_marginTop="53dp"

android:layout\_marginEnd="16dp"

android:background="#6C3939"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.496"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/imageView" />

<TextView

android:id="@+id/textView3"

android:layout\_width="257dp"

android:layout\_height="35dp"

android:layout\_marginStart="76dp"

android:layout\_marginTop="45dp"

android:layout\_marginEnd="78dp"

android:gravity="center"

android:text="Name"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/view" />

<TextView

android:id="@+id/textView4"

android:layout\_width="251dp"

android:layout\_height="35dp"

android:layout\_marginStart="80dp"

android:layout\_marginTop="21dp"

android:layout\_marginEnd="80dp"

android:gravity="center"

android:text="Jobtitle"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/textView3" />

<TextView

android:id="@+id/textView5"

android:layout\_width="259dp"

android:layout\_height="39dp"

android:layout\_marginStart="80dp"

android:layout\_marginTop="25dp"

android:layout\_marginEnd="72dp"

android:gravity="center"

android:text="Address"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/textView4" />

<View

android:id="@+id/view2"

android:layout\_width="298dp"

android:layout\_height="2dp"

android:layout\_marginTop="29dp"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.672"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/textView6" />

<TextView

android:id="@+id/textView6"

android:layout\_width="261dp"

android:layout\_height="44dp"

android:layout\_marginTop="30dp"

android:gravity="center"

android:text="Phone number"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.533"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/textView5" />

</androidx.constraintlayout.widget.ConstraintLayout>

* JAVA.CODE :

package com.example.vcardq;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

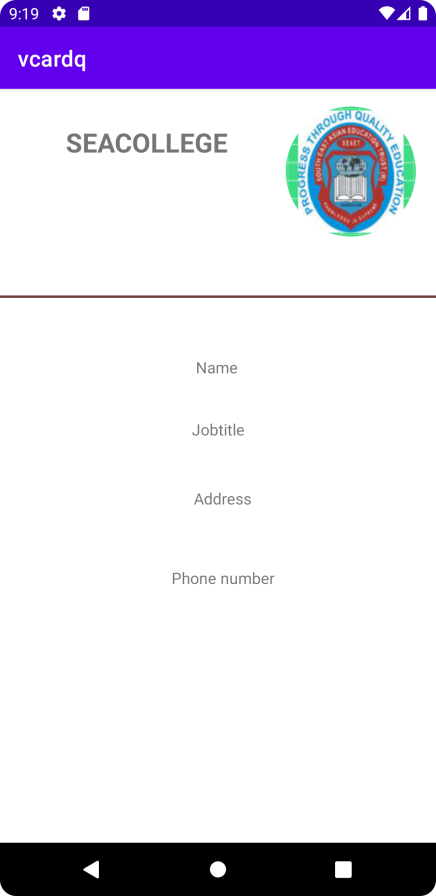
@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);}

**OUTPUT :**



**Program-2:**

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

* **XML –CODE:**

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

tools:context=".MainActivity">

<TextView

android:id="@+id/txt"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center\_horizontal"

android:text="@string/simple\_calculator"

android:textAlignment="center"

android:textAllCaps="true"

android:textSize="34sp"

android:textStyle="bold" />

<TextView

android:id="@+id/txtv3"

android:layout\_width="match\_parent"

android:layout\_height="59dp"

android:textAlignment="center"

android:textColor="#000000"

android:textSize="34sp"

android:textStyle="bold"

android:visibility="visible" />

<EditText

android:id="@+id/edt1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:ems="10"

android:hint="@string/enter\_number"

android:inputType="textPersonName"

android:minHeight="48dp" />

<TableLayout

android:layout\_width="410dp"

android:layout\_height="wrap\_content">

<TableRow

android:layout\_width="400dp"

android:layout\_height="wrap\_content">

<Button

android:id="@+id/btn1"

android:layout\_width="100dp"

android:layout\_height="wrap\_content"

android:baselineAligned="false"

android:text="1"

android:textAlignment="gravity"

android:textColor="#000000"

android:textSize="16sp"

android:textStyle="bold" />

<Button

android:id="@+id/btn2"

android:layout\_width="100dp"

android:layout\_height="wrap\_content"

android:text=”2"

android:textAlignment="gravity"

android:textColor="#000000"

android:textSize="16sp"

android:textStyle="bold" />

<Button

android:id="@+id/btn3"

android:layout\_width="100dp"

android:layout\_height="match\_parent"

android:text="3"

android:textAlignment="gravity"

android:textColor="#000000"

android:textSize="16sp"

android:textStyle="bold" />

<Button

android:id="@+id/btn4"

android:layout\_width="wrap\_content"

android:layout\_height="match\_parent"

android:text="/"

android:textColor="#000000"

android:textSize="16sp"

android:textStyle="bold" />

</TableRow>

<TableRow

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<Button

android:id="@+id/btn5"

android:text="4"

android:textColor="#000000"

android:textSize="16sp"

android:textStyle="bold" />

<Button

android:id="@+id/btn6"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="5"

android:textColor="#000000"

android:textSize="16sp"

android:textStyle="bold" />

<Button

android:id="@+id/btn7"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="6"

android:textColor="#000000"

android:textSize="16sp"

android:textStyle="bold" />

<Button

android:id="@+id/btn8"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="\*"

android:textColor="#000000"

android:textSize="16sp"

android:textStyle="bold" />

</TableRow>

<TableRow

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<Button

android:id="@+id/btn9"

android:text="7"

android:textColor="#000000"

android:textSize="16sp"

android:textStyle="bold">

</Button>

<Button

android:id="@+id/btn10"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="8"

android:textColor="#000000"

android:textSize="16sp" />

<Button

android:id="@+id/btn11"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="9"

android:textColor="#000000" />

<Button

android:id="@+id/btn12"

android:layout\_width="105dp"

android:layout\_height="wrap\_content"

android:text="+"

android:textColor="#000000"

android:textStyle="bold" />

</TableRow>

<TableRow

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<Button

android:id="@+id/btn13"

android:textColor="#000000"

android:textSize="16sp"

android:textStyle="bold"

tools:text="0"/>

<Button

android:id="@+id/btn14"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="c"

android:textColor="#000000" />

<Button

android:id="@+id/btn15"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="-"

android:textColor="#000000" />

<Button

android:id="@+id/btn16"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="="

android:textColor="#000000"

android:textSize="16sp"

android:textStyle="bold" />

</TableRow>

<TableRow

android:layout\_width="match\_parent"

android:layout\_height="match\_parent" >

<Button

android:id="@+id/button2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="." />

</TableRow>

</TableLayout>

*  JAVA-CODE :

package com.example.cv;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Switch;

import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

Button one, two, three, four, five, six, seven, eight, nine,zero,add,mul,div,sub,equal,

point,clear;

TextView disp;

String result;

String tmp;

String operator;

EditText resultTextView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

numbers();

initControlListener();

}

public void initControlListener() {

zero.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onNumberButtonClicked("0");

}

});

one.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onNumberButtonClicked("1");

}

});

two.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onNumberButtonClicked("2");

}

});

four.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onNumberButtonClicked("4");

}

});

five.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onNumberButtonClicked("5");

}

});

six.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onNumberButtonClicked("6");

}

});

seven.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onNumberButtonClicked("7");

}

});

eight.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onNumberButtonClicked("8");

}

});

nine.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onNumberButtonClicked("9");

}

});

three.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onNumberButtonClicked("3");

}

});

clear.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onClearButtonClicked();

}

});

div.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onOperatorButtonClicked("/");

}

});

mul.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onOperatorButtonClicked("\*");

}

});

add.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onOperatorButtonClicked("+");

}

});

sub.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onOperatorButtonClicked("-");

}

});

equal.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { onEqualButtonClicked();

}

});

point.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view){ onNumberButtonClicked(".");

}

});

}

public void onEqualButtonClicked() {

double res = 0;

try {

double number = Double.parseDouble(tmp);

doublenumber2=

Double.parseDouble(resultTextView.getText().toString());

switch (operator) {

case "+":

res = (number + number2);

break;

case "/":

res = (number / number2);

break;

case "-":

res = (number - number2);

break;

case "\*":

res = (number \* number2);

break;

}

result=res+"";

disp.setText(result);

resultTextView.setText("");

}

catch (Exception e) {

e.printStackTrace();

}

}

public void onOperatorButtonClicked(String op) {

tmp = resultTextView.getText().toString();

onClearButtonClicked();

operator = op;

}

public void onClearButtonClicked() {

result = "";

resultTextView.setText("");

disp.setText("");

}

private void onNumberButtonClicked(String pos) {

disp.setText("");

result = resultTextView.getText().toString() ;

result = result + pos;

resultTextView.setText(result);

}

private void numbers() {

one=findViewById(R.id.btn1);

two=findViewById(R.id.btn2);

three=findViewById(R.id.btn3);

div=findViewById(R.id.btn4);

four=findViewById(R.id.btn5);

five=findViewById(R.id.btn6);

six=findViewById(R.id.btn7);

mul=findViewById(R.id.btn8);

seven=findViewById(R.id.btn9);

eight=findViewById(R.id.btn10);

nine=findViewById(R.id.btn11);

add=findViewById(R.id.btn12);

zero=findViewById(R.id.btn13);

clear=findViewById(R.id.btn14);

sub=findViewById(R.id.btn15);

equal=findViewById(R.id.btn16);

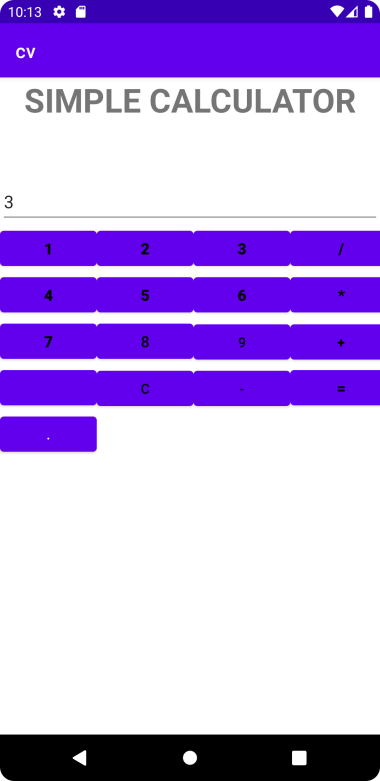
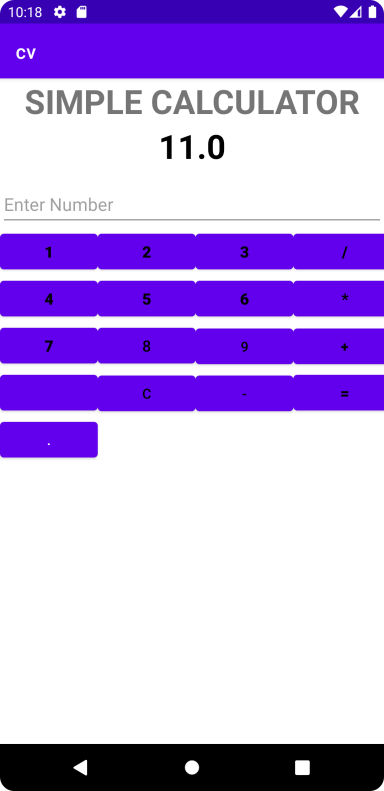
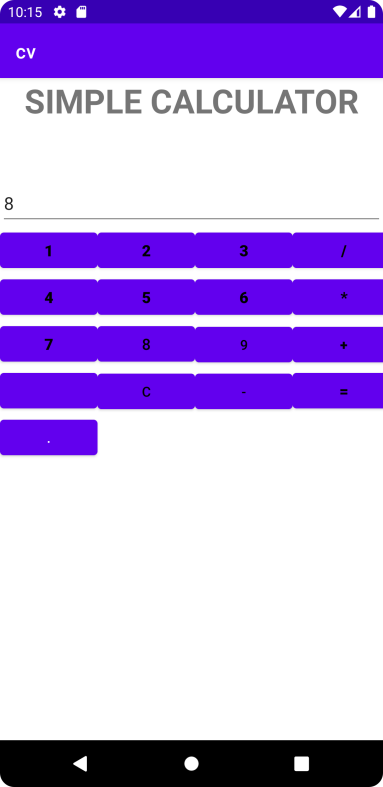
point=findViewById(R.id.button2);

disp=findViewById(R.id.txtv3);

resultTextView = findViewById(R.id.edt1);

}

 OUTPUT :

**Program-3:**

Create a SIGN Up activity with Username and Password. Validation of password should happen based on the following rules: • Password should contain uppercase and lowercase letters. • Password should contain letters and numbers. • Password should contain special characters. • Minimum length of the password (the default value is 8). On successful SIGN UP proceed to the next Login activity. Here the user should SIGN IN using the Username and Password created during signup activity. If the Username and Password are matched then navigate to the next activity which displays a message saying “Successful Login” or else display a toast message saying “Login Failed” .The user is given only two attempts and after that display a toast message saying “Failed Login Attempts” and disable the SIGN IN button. Use Bundle to transfer information from one activity to another

**XML –CODE:**

//Creating signup activity

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity">

<TextView

android:id="@+id/txtview1"

android:layout\_width="230dp"

android:layout\_height="24dp"

android:layout\_marginTop="144dp"

android:text="SignupActivity"

android:textAlignment="center"

android:textAllCaps="true"

android:textColor="#9C27B0"

android:textSize="20sp"

android:textStyle="bold"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.58"

app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<EditText

android:id="@+id/editpwd"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginEnd="12dp"

android:ems="10"

android:inputType="textPersonName"

app:layout\_constraintBottom\_toBottomOf="@+id/textView3"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="@+id/edittxt"

app:layout\_constraintTop\_toTopOf="@+id/textView3"

tools:ignore="TouchTargetSizeCheck,SpeakableTextPresentCheck" />

<TextView

android:id="@+id/textView2"

android:layout\_width="132dp"

android:layout\_height="35dp"

android:layout\_marginStart="16dp"

android:layout\_marginTop="64dp"

android:text="@string/username"

android:textSize="24sp"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/txtview1" />

<EditText

android:id="@+id/edittxt"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="36dp"

android:layout\_marginEnd="8dp"

android:ems="10"

android:inputType="textPersonName"

android:minHeight="48dp"

app:layout\_constraintBottom\_toBottomOf="@+id/textView2"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toEndOf="@+id/textView2"

app:layout\_constraintTop\_toTopOf="@+id/textView2"

tools:ignore="SpeakableTextPresentCheck" />

<TextView

android:id="@+id/textView3"

android:layout\_width="135dp"

android:layout\_height="46dp"

android:layout\_marginTop="40dp"

android:text="@string/password"

android:textSize="24sp"

app:layout\_constraintStart\_toStartOf="@+id/textView2"

app:layout\_constraintTop\_toBottomOf="@+id/textView2" />

<Button

android:id="@+id/but1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="136dp"

android:layout\_marginEnd="184dp"

android:text="SignUp"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/textView3"

/>

</androidx.constraintlayout.widget.ConstraintLayout>

//LoginActivity

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity2">

<TextView

android:id="@+id/txtview1"

android:layout\_width="230dp"

android:layout\_height="24dp"

android:layout\_marginTop="24dp"

android:text="LoginActivity"

android:textAlignment="center"

android:textAllCaps="true"

android:textColor="#9C27B0"

android:textSize="20sp"

android:textStyle="bold"

app:layout\_constraintHorizontal\_bias="0.43"

app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<EditText

android:id="@+id/editpwdnb"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="195dp"

android:layout\_marginEnd="6dp"

android:layout\_marginBottom="120dp"

android:ems="10"

android:inputType="textPersonName"

android:minHeight="48dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.98"

app:layout\_constraintStart\_toEndOf="@+id/textView3"

app:layout\_constraintStart\_toStartOf="parent"

tools:ignore="SpeakableTextPresentCheck" />

<TextView

android:id="@+id/textView2"

android:layout\_width="128dp"

android:layout\_height="43dp"

android:layout\_marginBottom="180dp"

android:text="@string/username"

android:textSize="24sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toStartOf="@+id/editusername"

app:layout\_constraintHorizontal\_bias="0.644"

app:layout\_constraintStart\_toStartOf="parent" />

<TextView

android:id="@+id/textView3"

android:layout\_width="135dp"

android:layout\_height="46dp"

android:layout\_marginStart="56dp"

android:layout\_marginBottom="112dp"

android:text="@string/password"

android:textSize="24sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintStart\_toStartOf="parent" />

<Button

android:id="@+id/but2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="160dp"

android:layout\_marginTop="120dp"

android:layout\_marginBottom="8dp"

android:text="@string/login"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.0" />

<EditText

android:id="@+id/editusername"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="186dp"

android:layout\_marginEnd="16dp"

android:layout\_marginBottom="180dp"

android:ems="10"

android:inputType="textPersonName"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

tools:ignore="SpeakableTextPresentCheck,TouchTargetSizeCheck" />

</androidx.constraintlayout.widget.ConstraintLayout>

//Welcome Activity

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:textAlignment="center"

tools:context=".MainActivity3">

<TextView

android:id="@+id/tve"

android:layout\_width="139dp"

android:layout\_height="88dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

*  JAVA.CODE : 
* MainActivity.Java.Code :

package com.example.signupactivity;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import java.util.regex.Pattern;

public class MainActivity extends AppCompatActivity {

EditText eunametxt, eupasstxt;

Button subtn;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

eunametxt = findViewById(R.id.edittxt);

eupasstxt = findViewById(R.id.editpwd);

subtn = findViewById(R.id.but1);

subtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String uname = eunametxt.getText().toString();

String pwd = eupasstxt.getText().toString();

if (!pwdval(pwd)) {

Toast.makeText(MainActivity.this, "Password does not match the rules", Toast.LENGTH\_LONG).show();

}

else {

Intent intent = new Intent( MainActivity.this, MainActivity2.class);

intent.putExtra("username", uname);

intent.putExtra("password", pwd);

startActivity(intent);

}

}

});

}

Pattern lowercase = Pattern.compile("^.\*[a-z].\*$");

Pattern uppercase = Pattern.compile("^.\*[A-Z].\*$");

Pattern number = Pattern.compile("^.\*[0-9].\*$");

Pattern specialCharacter = Pattern.compile("^.\*[^a-zA-Z0-9].\*$");

private boolean pwdval(String password) {

if (password.length() < 8) {

return false;

}

if (!lowercase.matcher(password).matches()) {

return false;

}

if (!uppercase.matcher(password).matches()) {

return false;

}

if (!number.matcher(password).matches()) {

return false;

}

if (!specialCharacter.matcher(password).matches()) {

return false;

} else {

return true;

}

}

}

* MainActivity2.Java.Code :

package com.example.signupactivity;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import java.util.regex.Pattern;

public class MainActivity2 extends AppCompatActivity {

EditText b,j;

Button l;

int count=3;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main2);

b = findViewById(R.id.editusername);

j = findViewById(R.id.editpwdnb);

l = findViewById(R.id.but2);

l.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String name=getIntent().getStringExtra("username");

String nm= getIntent().getStringExtra("password");

// Intent intent = getIntent();

// String name = intent.getStringExtra("username");

//String nm = intent.getStringExtra("password");

String usdname = b.getText().toString();

String pd = j.getText().toString();

Intent intent1 = new Intent(MainActivity2.this, MainActivity3.class);

if (name.equals(usdname) && nm.equals(pd)) {

intent1.putExtra("mes", "Successful Login");

intent1.putExtra("Username", usdname);

startActivity(intent1);

} else {

Toast.makeText(getApplicationContext(), "Invalid Credentials", Toast.LENGTH\_LONG).show();

}

count--;

if (count == 0) {

Toast.makeText(getApplicationContext(), "FAILED LOGIN ATTEMPTS", Toast.LENGTH\_LONG).show();

l.setEnabled(false);

}

}

});

}

MainActivity3.Java.Code:

package com.example.signupactivity;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.widget.TextView;

import android.widget.Toast;

public class MainActivity3 extends AppCompatActivity {

TextView name;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main3);

name=findViewById(R.id.tve);

Intent b=getIntent();

String mesa=b.getStringExtra("mes");

Toast.makeText(getApplicationContext(),mesa,Toast.LENGTH\_LONG).show();

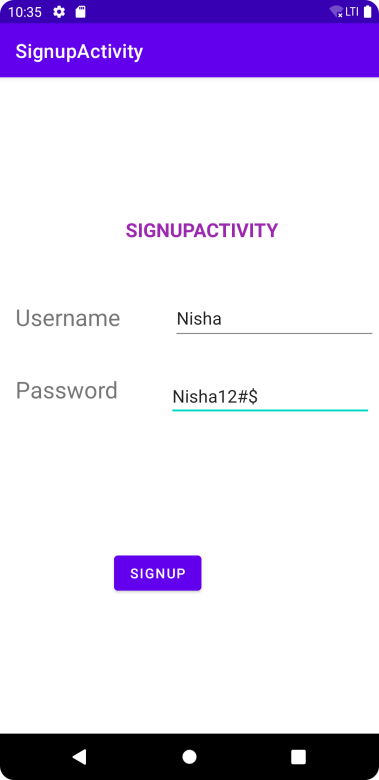
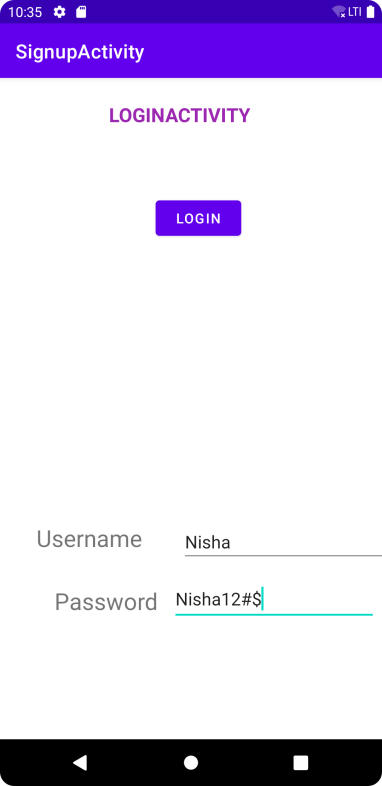
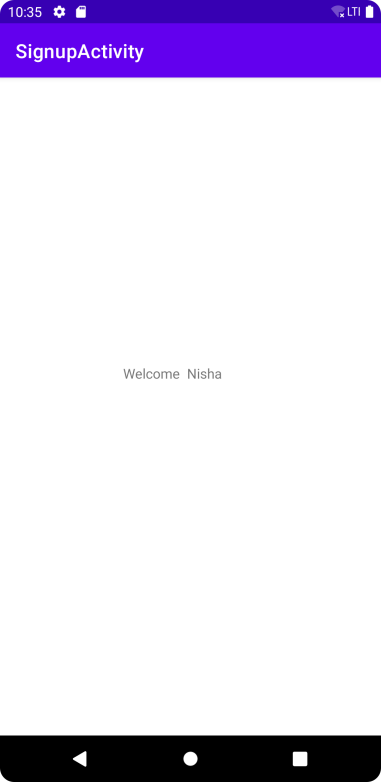
String uname=b.getStringExtra("Username");

name.setText("Welcome "+"" +uname);

}

}

**OUTPUT :**

****

Program-4:

Develop an application to set an image as wallpaper. On click of a button, the wallpaper image

should start to change randomly every 30 seconds.

* **XML –CODE:**

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

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity">

<TextView

android:id="@+id/text1"

android:layout\_width="219dp"

android:layout\_height="39dp"

android:text="Wallpaper\_changer"

android:textSize="25sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.069" />

<Button

android:id="@+id/button"

android:layout\_width="266dp"

android:layout\_height="58dp"

android:text="click\_to\_change\_wallpaper"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.222" />

</androidx.constraintlayout.widget.ConstraintLayout>

❖**JAVA CODE:**

package com.example.seawallpaper;

import androidx.appcompat.app.AppCompatActivity;

import androidx.core.content.res.ResourcesCompat;

import android.app.WallpaperManager;

import android.graphics.Bitmap;

import android.graphics.drawable.BitmapDrawable;

import android.graphics.drawable.Drawable;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import java.io.IOException;

import java.util.Timer;

import java.util.TimerTask;

public class MainActivity extends AppCompatActivity {

Button wall;

Timer tim;

Drawable drawable;

WallpaperManager wlp;

int prev = 1;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

tim = new Timer();

wlp = WallpaperManager.getInstance(this);

wall = findViewById(R.id.button);

wall.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

setWallpaper();

}

});

}

private void setWallpaper() {

tim.schedule(new TimerTask() {

@Override

public void run() {

if (prev == 1) {

drawable = ResourcesCompat.getDrawable(getResources(), R.drawable.bb, null);

prev = 2;

} else if (prev == 2) {

drawable = ResourcesCompat.getDrawable(getResources(), R.drawable.e, null);

prev = 3;

} else if (prev == 3) {

drawable = ResourcesCompat.getDrawable(getResources(), R.drawable.c, null);

prev = 4;

} else if (prev == 4) {

drawable = ResourcesCompat.getDrawable(getResources(), R.drawable.a, null);

prev = 5;

} else if (prev == 5) {

drawable = ResourcesCompat.getDrawable(getResources(), R.drawable.download, null);

prev = 1;

}

/\*A bitmap is an image file format which is used to store the digital images.

The word bitmap means map of bits. They are used to create realistic graphics and images.

\*/

Bitmap wallpaper = ((BitmapDrawable) drawable).getBitmap();

try {

wlp.setBitmap(wallpaper);

} catch (IOException e) {

e.printStackTrace();

}

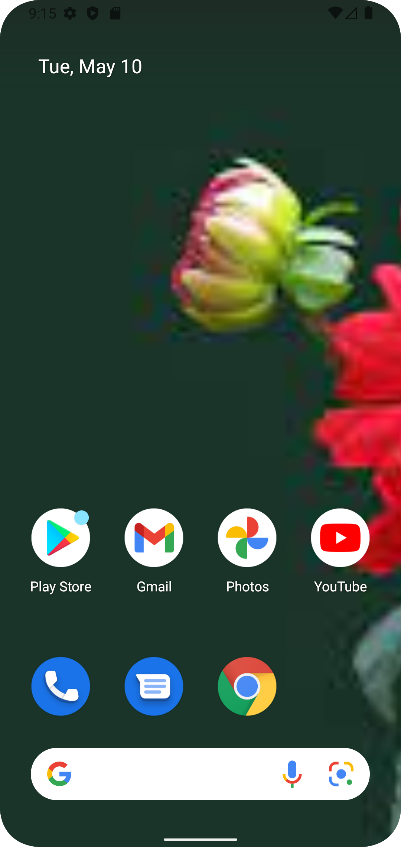
}

}, 0, 30000);

}

}

**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 TextView control.

* **XML –CODE:**

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

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity">

<TextView

android:id="@+id/view1"

android:layout\_width="176dp"

android:layout\_height="45dp"

android:text="counter\_app"

android:textSize="30sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.069" />

<Button

android:id="@+id/startbut"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="start"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.498"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.183" />

<TextView

android:id="@+id/out"

android:layout\_width="65dp"

android:layout\_height="59dp"

android:textSize="25sp"

android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.531"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.303" />

<Button

android:id="@+id/stopbut"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="stop"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.507"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.404" />

</androidx.constraintlayout.widget.ConstraintLayout>

* JAVA.CODE :

package com.example.seacounter;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.os.Handler;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

Button start, stop;

TextView counter;

int i;

Handler handler;

//handler is used to delay ,some task for certain interval of time.

Runnable runnable;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

start = findViewById(R.id.startbut);

stop = findViewById(R.id.stopbut);

counter = findViewById(R.id.out);

handler = new Handler();

start.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Counter();

start.setEnabled(false);

stop.setEnabled(true);

}

});

stop.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

handler.removeCallbacks(runnable);

i = 0;

start.setEnabled(true);

stop.setEnabled(false);

}

});

}

public void Counter() {

runnable = new Runnable() {

@Override

public void run() {

i++;

counter.setText(String.valueOf(i));

//To delay ,run method of runnable we use postdelayed method of handler class

// which having two parameters

// context and delay time.

handler.postDelayed(this, 100);

}

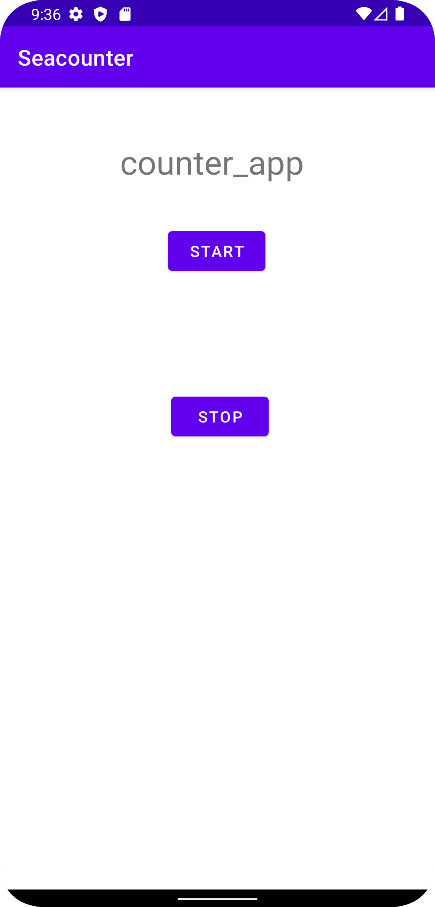
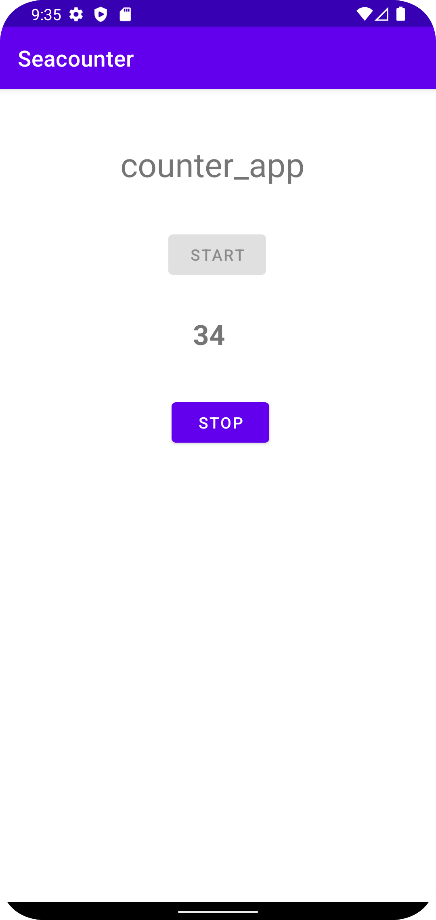
};

handler.post(runnable);

}

}

 OUTPUT :

Program-6:

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

* **XML –CODE:**

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

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_gravity="center"

android:textAlignment="center"

tools:context=".MainActivity">

<Button

android:id="@+id/button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="36dp"

android:layout\_marginTop="96dp"

android:text="JSON DATA"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<Button

android:id="@+id/button2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="96dp"

android:text="XMLDATA"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toEndOf="@+id/button"

app:layout\_constraintTop\_toTopOf="parent" />

<EditText

android:id="@+id/editjson"

android:layout\_width="184dp"

android:layout\_height="428dp"

android:ems="10"

android:gravity="start|top"

android:inputType="textMultiLine"

android:textColor="#000000"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/button"

app:layout\_constraintVertical\_bias="0.238"

tools:ignore="SpeakableTextPresentCheck" />

<EditText

android:id="@+id/editxml"

android:layout\_width="171dp"

android:layout\_height="417dp"

android:ems="10"

android:gravity="start|top"

android:inputType="textMultiLine"

android:textColor="#000000"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toEndOf="@+id/editjson"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.579"

tools:ignore="SpeakableTextPresentCheck" />

<TextView

android:id="@+id/textView"

android:layout\_width="347dp"

android:layout\_height="44dp"

android:layout\_marginTop="4dp"

android:orientation="horizontal"

android:text="DATA FROM JSON AND XML"

android:textAlignment="gravity"

android:textSize="16sp"

app:layout\_constraintBottom\_toTopOf="@+id/button2"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.25" />

</androidx.constraintlayout.widget.ConstraintLayout>

* **JAVA CODE:**

package com.example.xmljsondata;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

import android.widget.Toast;

import org.json.JSONArray;

import org.json.JSONObject;

import org.w3c.dom.Document;

import org.w3c.dom.Element;

import org.w3c.dom.Node;

import org.w3c.dom.NodeList;

import java.io.InputStream;

import java.nio.charset.StandardCharsets;

import javax.xml.parsers.DocumentBuilder;

import javax.xml.parsers.DocumentBuilderFactory;

public class MainActivity extends AppCompatActivity {

TextView js, xm;

Button jso, xml;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

js = findViewById(R.id.editjson);

xm = findViewById(R.id.editxml);

jso = findViewById(R.id.button);

xml = findViewById(R.id.button2);

jso.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view)

{

String json;

StringBuilder stringBuilder = new StringBuilder();

try {

InputStream is = getAssets().open("city.json");

int size = is.available();

byte[] buffer = new byte[size];

is.read(buffer);

json = new String(buffer, StandardCharsets.UTF\_8);

JSONObject jsonObject = new JSONObject(json);

JSONArray jsonArray = jsonObject.getJSONArray("cities");

stringBuilder.append("JSON DATA");

stringBuilder.append("\n--------");

for (int i = 0; i < jsonArray.length(); i++) {

JSONObject jsonObject1 = jsonArray.getJSONObject(i);

stringBuilder.append("\nName: ").append(jsonObject1.getString("City\_name"));

stringBuilder.append("\nLatitude: ").append(jsonObject1.getString("Latitude"));

stringBuilder.append("\nLongitude: ").append(jsonObject1.getString("Longitude"));

stringBuilder.append("\nTemperature: ").append(jsonObject1.getString("Temperature"));

stringBuilder.append("\nHumidity: ").append(jsonObject1.getString("Humidity"));

stringBuilder.append("\n----------");

}

js.setText(stringBuilder.toString());

Toast.makeText(MainActivity.this, stringBuilder.toString(), Toast.LENGTH\_LONG).show();

is.close();

} catch (Exception e) {

e.printStackTrace();

Toast.makeText(MainActivity.this, "Error in reading" + e.toString(), Toast.LENGTH\_LONG).show();

}

}

});

xml.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

try {

InputStream is = getAssets().open("city.xml");

DocumentBuilderFactory documentBuilderFactory = DocumentBuilderFactory.newInstance();

DocumentBuilder documentBuilder = documentBuilderFactory.newDocumentBuilder();

Document document = documentBuilder.parse(is);

StringBuilder stringBuilder = new StringBuilder();

stringBuilder.append("XML DATA");

stringBuilder.append("\n---------");

NodeList nodeList = document.getElementsByTagName("city");

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

Node node = nodeList.item(i);

if (node.getNodeType() == Node.ELEMENT\_NODE) {

Element element = (Element) node;

stringBuilder.append("\nName: ").append(getValue("City\_name", element));

stringBuilder.append("\nLatitude: ").append(getValue("Latitude", element));

stringBuilder.append("\nLongitude: ").append(getValue("Longitude", element));

stringBuilder.append("\nTemperature: ").append(getValue("Temperature", element));

stringBuilder.append("\nHumidity: ").append(getValue("Humidity", element));

stringBuilder.append("\n----------");

}

}

xm.setText(stringBuilder.toString());

} catch (Exception e) {

e.printStackTrace();

Toast.makeText(MainActivity.this, "Error Parsing XML", Toast.LENGTH\_LONG).show();

}

}

private String getValue(String tag, Element element) {

return element.getElementsByTagName(tag).item(0).getChildNodes().item(0).getNodeValue();

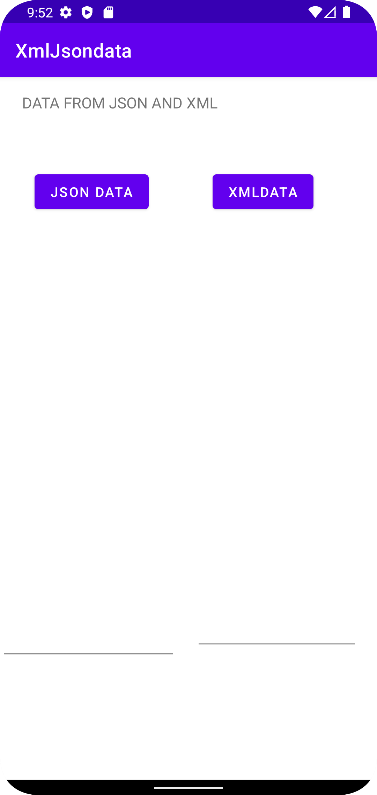
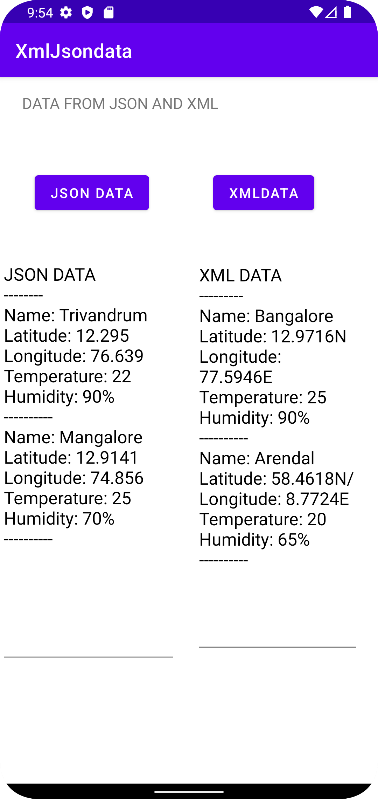
}

});

}

}

 OUTPUT :

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.

* XML –CODE:

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

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="<http://schemas.android.com/apk/res/android>"

xmlns:app="[http://schemas.android.com/apk/res-auto"](http://schemas.android.com/apk/res-auto) xmlns:tools="[http://schemas.android.com/tools"](http://schemas.android.com/tools) android:id="@+id/hint"

android:layout\_width="match\_parent" android:layout\_height="match\_parent"

tools:context=".MainActivity">

<TextView

android:id="@+id/textView2"

android:layout\_width="387dp"

android:layout\_height="50dp"

android:text="TEXT TO SPEECH APPLICATION"

android:textColor="@color/black"

android:textSize="26sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintLeft\_toLeftOf="parent" app:layout\_constraintRight\_toRightOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.192" />

<TextView android:id="@+id/textView"

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content"

android:text="Enter the text:"

android:textColor="@color/black"

android:textSize="24sp"

android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.134" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.32" />

<EditText android:id="@+id/texttoconv"

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content"

android:ems="10"

android:hint="Enterthetexhere..." android:inputType="textPersonName" android:textColor="@color/black"

android:textColorHighlight="@color/black"

android:textColorLink="@color/white"

android:textSize="24sp" android:textStyle="italic"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.593" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.423" />

<Button android:id="@+id/convert"

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Convert Text to Speech" android:textColor="#FFFFFF" android:textSize="18sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.614" />

</androidx.constraintlayout.widget.ConstraintLayout>

* **JAVA CODE:**

package com.example.texttospeech;

import static android.speech.tts.TextToSpeech.ERROR;

importstaticandroid.speech.tts.TextToSpeech.QUEUE\_FLUSH; import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.speech.tts.TextToSpeech; import android.view.View;

import android.widget.Button; import android.widget.EditText; import android.widget.Toast; import java.util.Locale;

public class MainActivity extends AppCompatActivity { EditText b;

Button n; TextToSpeech k;

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

b = findViewById(R.id.edittext); n = findViewById(R.id.button);

k = new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener() { @Override

public void onInit(int status) {

if (status != ERROR) { k.setLanguage(Locale.UK);

}

}

});

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

public void onClick(View view) { String h = b.getText().toString();

k.speak(h, TextToSpeech.QUEUE\_FLUSH, null, null); Toast.makeText(MainActivity.this, h, Toast.LENGTH\_SHORT).show();

}

});

}

/\* ctrl+o to override the lifecycle event\*/ @Override

protected void onPause() { if (k != null) {

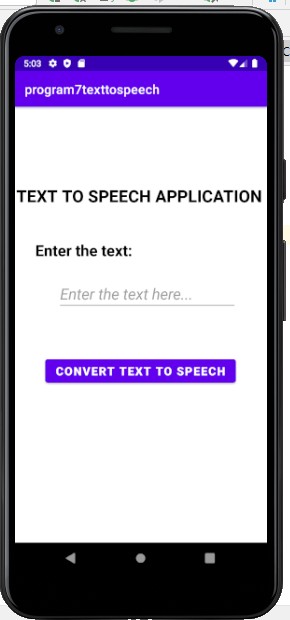
k.stop();

k.shutdown();

}

super.onPause();

}

}

**OUTPUT :**

Program 8:

### Create an activity like a phone dialer with CALLand 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.

* **XML –CODE:**

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

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="<http://schemas.android.com/apk/res/android>"

xmlns:app="[http://schemas.android.com/apk/res-auto"](http://schemas.android.com/apk/res-auto%22) xmlns:tools="[http://schemas.android.com/tools"](http://schemas.android.com/tools) android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:onClick="inputNumber" tools:context=".MainActivity">

<TextView android:id="@+id/text1"

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content"

android:background="#03A9F4" android:text="CALLANDSAVEAPPLICATION"

android:textColor="@color/black" android:textSize="26sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintHorizontal\_bias="0.405" app:layout\_constraintLeft\_toLeftOf="parent" app:layout\_constraintRight\_toRightOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.05" />

<EditText android:id="@+id/phnum" android:layout\_width="257dp" android:layout\_height="49dp" android:ems="10"

android:inputType="textPersonName" android:textColor="@color/black" android:textSize="24sp" app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.103" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="@+id/text1" app:layout\_constraintVertical\_bias="0.108" />

<Button android:id="@+id/butndel" android:layout\_width="80dp" android:layout\_height="45dp" android:text="DEL"

android:textColor="@color/white"

android:textSize="20sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.915" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.153" />

<Button android:id="@+id/butn1" android:layout\_width="63dp" android:layout\_height="60dp"

android:onClick="inputNumber" android:text="1" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.071" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.369" />

<Button android:id="@+id/butn2" android:layout\_width="63dp" android:layout\_height="60dp"

android:onClick="inputNumber" android:text="2" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.37" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.369" />

<Button android:id="@+id/butn3" android:layout\_width="63dp" android:layout\_height="60dp"

android:onClick="inputNumber" android:text="3" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.66" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.369" />

<Button android:id="@+id/butn4" android:layout\_width="63dp" android:layout\_height="60dp"

android:onClick="inputNumber" android:text="4" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.071" app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.499" />

<Button android:id="@+id/butn5" android:layout\_width="63dp" android:layout\_height="60dp"

android:onClick="inputNumber" android:text="5" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.37" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.499" />

<Button android:id="@+id/butn6" android:layout\_width="63dp" android:layout\_height="60dp"

android:onClick="inputNumber" android:text="6" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.66" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.499" />

<Button android:id="@+id/butn7" android:layout\_width="63dp" android:layout\_height="60dp"

android:onClick="inputNumber"

android:text="7" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.071" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.63" />

<Button android:id="@+id/butn8" android:layout\_width="63dp" android:layout\_height="60dp"

android:onClick="inputNumber" android:text="8" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.37" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.63" />

<Button android:id="@+id/butn9" android:layout\_width="63dp"

android:layout\_height="60dp" android:onClick="inputNumber" android:text="9" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.66" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.63" />

<Button android:id="@+id/butnstar" android:layout\_width="63dp" android:layout\_height="60dp" android:onClick="inputNumber" android:text="\*" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.071" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.773" />

<Button android:id="@+id/butn0" android:layout\_width="63dp" android:layout\_height="60dp"

android:onClick="inputNumber" android:text="0"

android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.37" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.773" />

<Button android:id="@+id/butnhash" android:layout\_width="63dp" android:layout\_height="60dp" android:onClick="inputNumber" android:text="#" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.66" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.773" />

<Button android:id="@+id/butncall" android:layout\_width="140dp" android:layout\_height="62dp" android:text="CALL" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.066" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.922" />

<Button android:id="@+id/butnsave" android:layout\_width="140dp" android:layout\_height="62dp" android:text="SAVE" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintHorizontal\_bias="0.699" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" app:layout\_constraintVertical\_bias="0.922" />

</androidx.constraintlayout.widget.ConstraintLayout>

* **JAVA CODE:**

package com.example.callsave\_091;

import androidx.annotation.NonNull;

import androidx.appcompat.app.AppCompatActivity;

import androidx.core.app.ActivityCompat;

import androidx.core.content.ContextCompat;

import android.Manifest;

import android.content.Intent;

import android.content.pm.PackageManager;

import android.net.Uri;

import android.os.Bundle;import android.provider.ContactsContract;

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

import android.widget.EditText;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity

{ private static final int WRITE\_CONTACTS=1;

private static final int REQUEST\_CALL=1;

EditText display;

Button btn1,btn2, btn3,btn4, btn5,btn6, btn7,btn8, btn9,btn0;

Button btnCall,btnSave,btnStar,btnHash,btnRemove;

@Override

protected void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

btn0 = findViewById(R.id.btn0);btn1 = findViewById(R.id.btn1);

btn2 = findViewById(R.id.btn2);btn3 = findViewById(R.id.btn3);

btn4 = findViewById(R.id.btn4);btn5 = findViewById(R.id.btn5);

btn6 = findViewById(R.id.btn6);btn7 = findViewById(R.id.btn7);

btn8 = findViewById(R.id.btn8);btn9 = findViewById(R.id.btn9);

btnCall = findViewById(R.id.btnCall);btnSave = findViewById(R.id.btnSave);

btnRemove = findViewById(R.id.btnRemove);btnStar = findViewById(R.id.btnStar);

btnHash = findViewById(R.id.btnHash);

display = findViewById(R.id.display);

btn0.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

display.append("0");

}

});

btn1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

display.append("1");

}

});

btn2.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

display.append("2");

}

});

btn3.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

display.append("3");

}

});

btn4.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

display.append("4");

}

});

btn5.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

display.append("5");

}

});

btn6.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

display.append("6");

}

});

btn7.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

display.append("7");

}

});

btn8.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

display.append("8");

}

});

btn9.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

display.append("9");

}

});

btnStar.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

display.append("\*");

}

});

btnHash.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

display.append("#");

}

});

btnCall.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

makephonecall();

}

});

btnSave.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String data = display.getText().toString();

if (data.trim().length() > 0) {

Intent intent = new Intent(ContactsContract.Intents.Insert.ACTION);

intent.setType(ContactsContract.RawContacts.CONTENT\_TYPE);

intent.putExtra(ContactsContract.Intents.Insert.NAME, "Unknown");

intent.putExtra(ContactsContract.Intents.Insert.PHONE, data);

startActivity(intent);

}

else {

Toast.makeText(MainActivity.this, "Enter the data", Toast.LENGTH\_SHORT).show();

}

}

});

btnRemove.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String data = display.getText().toString();

if(data.length()>0){

String val = data.substring(0,data.length()-1);

display.setText(val);

}

else{

display.setText("");

}

}

});

}

private void makephonecall() {

String data = display.getText().toString();

if(data.trim().length()>0)

{

if(ContextCompat.checkSelfPermission(MainActivity.this, Manifest.permission.CALL\_PHONE )!= PackageManager.PERMISSION\_GRANTED)

{

ActivityCompat.requestPermissions(MainActivity.this,new String[]{Manifest.permission.CALL\_PHONE},REQUEST\_CALL);

}

else {

String dial="tel:"+data;

startActivity(new Intent(Intent.ACTION\_CALL,Uri.parse(dial)));

}

}

else {

Toast.makeText(this, "Enter phone number", Toast.LENGTH\_SHORT).show();

}

}

@Override

public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {

super.onRequestPermissionsResult(requestCode, permissions, grantResults);

if (requestCode == REQUEST\_CALL) {

if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED) {

makephonecall();

} else {

Toast.makeText(this, "Permission denied", Toast.LENGTH\_SHORT).show();

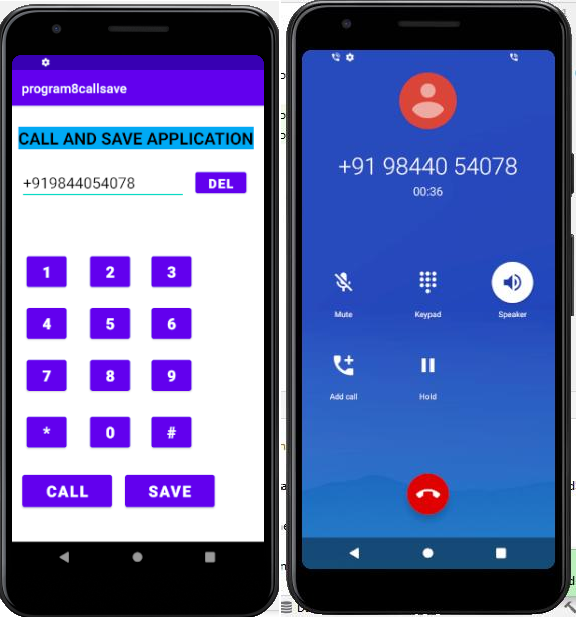
}

}

}

}

**OUTPUT :**



**Sample viva questions:**

**Q1. Android Is Developed By**

1. Apple
2. Microsoft
3. Google
4. **Android Inc**

**Q2. Android Web Browser Is Based On**

1. Chrome
2. **Open-source Webkit**
3. Safari
4. Firefox

**Q3. Android Is Based On Which Kernal**

1. **Linux**
2. Windows
3. Mac
4. Redhat

**Q4. Which Media Format Is Not Supported By Android**

1. MP4
2. **AVI**
3. MIDI
4. MPEG

**Q5. In Which Directory XML Layout Files Are Stored**

1. /assets
2. /src
3. /res/values
4. **/res/layout**

**Q6. Which Code Used By Android Is Not A Open Source.**

1. Video Driver
2. **WiFi Driver**
3. Device Driver
4. Bluetooth Driver

**Q7. How Many Levels Of Securities Are In Android?**

Android Level Security

**App And Kernel Level Security**

**Q8. Which Of The Following Does Not Belong To Transitions?**

1. ViewFlipper
2. ViewAnimator
3. ViewSwitcher
4. **ViewSlider**

**Q9. What Are The Functionalities In AsyncTask In Android?**

1. OnPreExecution()
2. **OnPostExecution()**
3. DoInBackground()
4. OnProgressUpdate()

**Q10. What Does AAPT Stands For?**

1. Android Asset Processing Tool.
2. Android Asset Providing Tool.
3. **Android Asset Packaging Tool.**
4. Android Asset Packaging Technique

**Q11. Which Broadcast In Android Includes Information About Battery State Level**

1. **Android.intent.action.BATTERY\_CHANGED**
2. Android.intent.action.BATTERY\_LOW
3. Android.intent.action.BATTERY\_OKAY
4. Android.intent.action.CALL\_BUTTON

**Q12. View Pager Is Used For**

1. Swiping Activities
2. **Swiping Fragments**
3. Paging Down List Items
4. View Pager Is Not Supported By Android SDK

**Q13. What Is JNI In Android?**

1. Java Interface
2. **Java Native Interface**
3. Java Network Interface
4. Image Editable Tool

**Q14. Adb Stands For**

1. **Android Debug Bridge.**
2. Android Drive Bridge.
3. Android Delete Bridge
4. Android Destroy Bridge.

**Q15. What Is LastKnownLocation In Android?**

1. To Find The Last Location Of A Phone
2. To Find Known Location Of A Phone
3. To Find The Last Known Location Of A Phone.
4. **None Of The Above**

**Q16. Which Programming Language Is Used For Android Application Development?**

1. NodeJs
2. PHP
3. JSX
4. **Java**

**Q17. Action Bar Can Be Associated To**

1. Only Fragments
2. **Only Activities**
3. Both Activities And Fragments
4. None Of The Above

**Q18. What is Manifest.xml in android?**

1. It has information about layout in an application
2. **It has all the information about an application**
3. It has the information about activities in an application
4. None of the above

**Q19. In which technique, we can refresh the dynamic content in android?**

1. **Ajax**
2. Java
3. Android
4. None of These

**Q20. What is an interface in android?**

1. Interface is a class.
2. **Interface acts as a bridge between class and the outside world.**
3. Interface is a layout file
4. All of the above

**What is APK in android?**

* 1. Android pack
  2. Android packages
  3. **Android Package Kit**
  4. None of above

**Q22. What is the name of the program that converts Java byte code into Dalvik byte code?**

1. **Dex compiler**
2. Dalvik Converter
3. Android Interpretive Compiler (AIC)
4. Mobile Interpretive Compiler (MIC)

**Q23. Jelly Bean is an incremental update, with the primary aim of improving the .............**

1. functionality
2. user interface
3. performance
4. All of the Above

**Q24. What does API stand for?**

1. **Application programming interface**
2. Algorithmic Protocol Interface
3. Accelerated Programming Interface
4. None of above

**Q25. What is the latest version of android?**

1. Android 8
2. Android 9
3. Android 10
4. **Android 11**

****

****