FACULTY OF ENGINEERING AND TECHNOLOGY

BACHELOR OF TECHNOLOGY

MOBILE APLLICATION DEVELOPMENT LABORATORY

(303105380)

VISEMESTER

COMPUTERSCIENCE&ENGINEERINGDEPARTMENT

LaboratoryManual

Session:2024-25

CERTIFICATE

Thisistocertifythat

Mr./Ms.……………………………………………………………………

with Enrollment No …………………………………. has successfully completed his/her laboratory experiments in the **MOBILE APLLICATION DEVELOPMENT LABORATORY(303105380)**from the department of **COMPUTER SCIENCE &ENGINEERING** during the academic year 2024-2025.

DateofSubmission:.........................

StaffIncharge:...........................

HeadOfDepartment:..

Practical No: 1

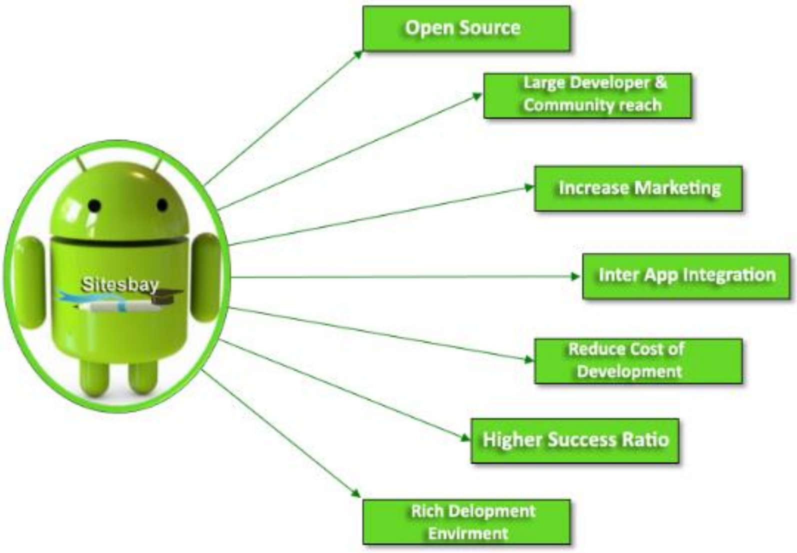
Aim:

ComparevariousoperatingsystemswithAndroidOS.

Android:

- Android is a software package and Linux based operating system for mobile devices such as tablet computers and smartphones.
- It is developed by Google and later the OHA (Open Handset Alliance). Java language is mainly used to write the android code even though other languages can be used.
- Thegoal of androidproject is to create asuccessful real-worldproduct that improves themobile experience for end users.
- Android offers a unified approach to application development for mobile devices which means developers need only develop for Android, and their applications should be able to run on different devices powered by Android.
- The first beta version of the Android Software Development Kit (SDK) was released by Google in 2007 where as the first commercial version, Android 1.0, was released in September 2008.
- The source code for Android is available under free and open-source software licenses. Google publishes most of the code under the Apache License version 2.0 and the rest, Linux kernel changes, under the GNU General Public License version 2.
- There are many code names of android such as Lollipop, KitKat, Jelly Bean, Ice cream Sandwich, Froyo, Eclair, Donut etc.

Why Android?



Features of Android OS

- → Android is an open-source operating system, allowing developers and manufacturers to customize and modify it.
- → It is built on the Linux kernel, providing security, stability, and efficient memory management.
- → Millions of apps are available on the Google Play Store, covering various categories like gaming, productivity, and social media.
- → Users can customize themes, widgets, wallpapers, and launchers to personalize their experience.
- → Allows running multiple applications simultaneously with features like split-screen mode and picture-in-picture mode.
- → Provides an interactive notification system, allowing users to reply to messages, control media, and take actions directly from notifications.
- → Comes with Google Assistant for voice commands, automation, and AI-driven features.
- → Receives frequent updates with new features, security patches, and performance enhancements.
- → Supports multiple connectivity options such as Wi-Fi, Bluetooth, NFC, 5G, and USB OTG.
- → Offers support for more than 100 languages, making it globally accessible.
- → Includes features like Google Play Protect, biometric authentication (fingerprint, face unlock), app sandboxing, and encryption for enhanced security.

Categories of Android Applications

1. **Social Media Apps** – Apps for communication and social networking.
2. **Entertainment & Multimedia Apps** – Apps for music, video streaming, and gaming.
3. **Productivity Apps** – Apps for office work, task management, and note-taking.
4. **Gaming Apps** – Apps for casual and high-end gaming.
5. **Education & Learning Apps** – Apps for online learning and skill development.
6. **E-Commerce & Shopping Apps** – Apps for online shopping and transactions.
7. **Finance & Banking Apps** – Apps for digital payments, banking, and investments.
8. **Health & Fitness Apps** – Apps for workout tracking, meditation, and medical services.
9. **Navigation & Travel Apps** – Apps for maps, transportation, and travel booking.
10. **Utility & Tools Apps** – Apps for enhancing device functionality.

Comparison Between Various Operating Systems

Aspect	**Android OS**	**Windows OS**	**Linux OS**	**iOS**
Developer	Google (Open Handset Alliance)	Microsoft Corporation	Community-driven (Initially by Linus Torvalds)	Apple Inc.

Initial Release	2008	1983	1991	2007
Kernel	Linux-based	Windows NT	Linux	Unix-based
Source Code	Open-source (AOSP)	Closed-source	Open-source	Closed-source (Partially open under Apple Public Source License)
Primary Devices	Smartphones, Tablets, Smart TVs, Wearables	PCs, Laptops, Tablets	Servers, Desktops, Embedded Systems	iPhones, iPads, Apple Watch, Apple TV
User Interface (UI)	Touch-based (Material Design)	Graphical UI (Start Menu, Taskbar)	CLI + GUI (Varies by distribution)	Touch-based (Flat UI)
App Store	Google Play Store, Third-party Stores	Microsoft Store, EXE files	Package Managers (apt, yum, snap)	Apple App Store
Customization	Highly customizable	Limited	Highly customizable	Very limited
Security	Google Play Protect, App sandboxing	Requires antivirus, Windows Defender	Highly secure with user control	Very secure, strict App Store policies
Performance	Optimized for mobile devices	Requires high resources	Varies (Lightweight & Heavy distros)	Optimized for Apple devices
Software Availability	Mobile apps	Wide range of applications	Extensive open-source software	Mobile apps only
Updates	Frequent security and feature updates	Regular updates with major feature releases	Rolling updates (some distributions)	Annual major updates
Multitasking	Yes, supports split-screen & background apps	Yes, supports multiple windows	Yes, supports multiple processes	Limited multitasking (background apps restricted)

Practical No:2

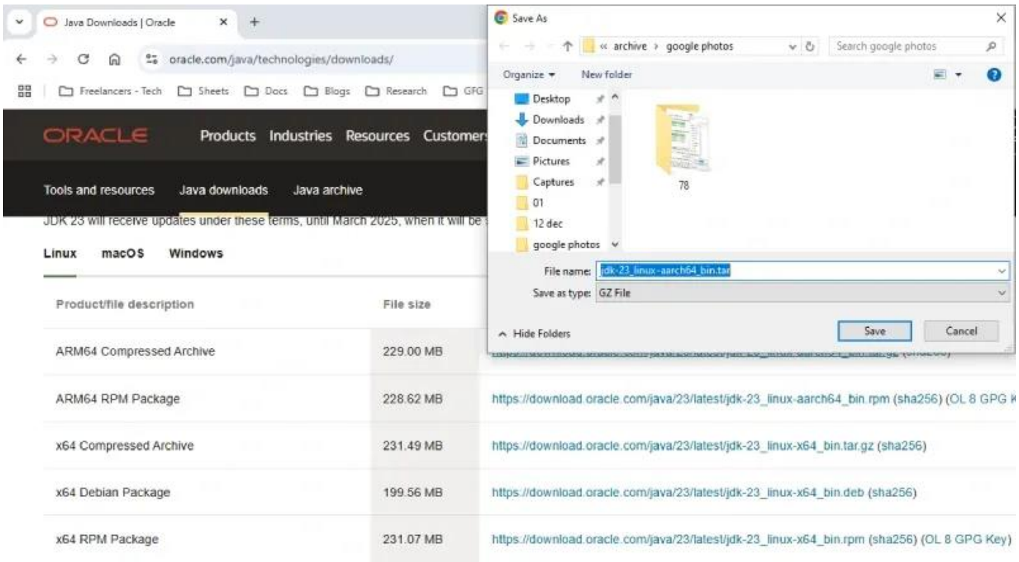
Aim:

Installandconfigurejavadevelopmentkit(JDK),androidstudioand android SDK.

Download JDK (Java Development Kit)

Step 1: Visit the Official Website

Go to the Official JDK website to download the file.



JDK 23 will receive updates under these terms, until March 2025, when it will be

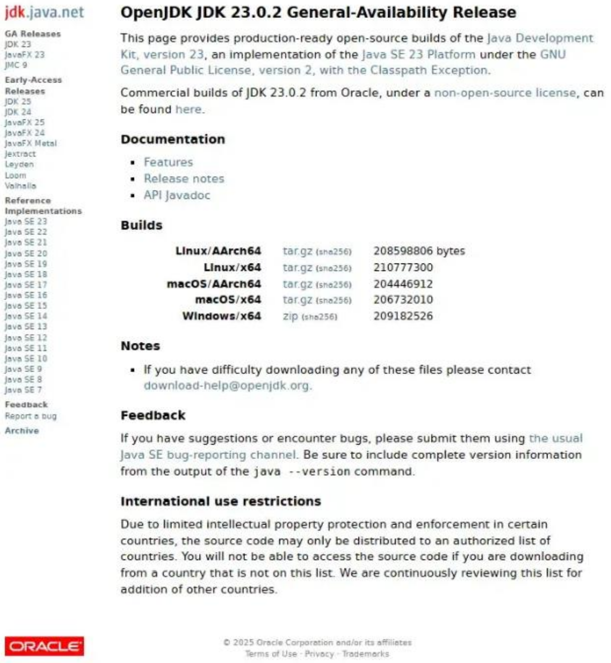
Linux macOS Windows

File name: jdk-23_linux-aarch64_bin.tar

Save as type: GZ File

Product/file description	File size	
ARM64 Compressed Archive	229.00 MB	[illegible]
ARM64 RPM Package	228.62 MB	https://download.oracle.com/java/23/latest/jdk-23_linux-aarch64_bin.rpm (sha256) (OL 8 GPG K
x64 Compressed Archive	231.49 MB	https://download.oracle.com/java/23/latest/jdk-23_linux-x64_bin.tar.gz (sha256)
x64 Debian Package	199.56 MB	https://download.oracle.com/java/23/latest/jdk-23_linux-x64_bin.deb (sha256)
x64 RPM Package	231.07 MB	https://download.oracle.com/java/23/latest/jdk-23_linux-x64_bin.rpm (sha256) (OL 8 GPG Key)

Since JDK is an open-source, so you can get it from OpenJDK in a few clicks.



OpenJDK JDK 23.0.2 General-Availability Release

This page provides production-ready open-source builds of the Java Development Kit, version 23, an implementation of the Java SE 23 Platform under the GNU General Public License, version 2, with the Classpath Exception.

Commercial builds of JDK 23.0.2 from Oracle, under a non-open-source license, can be found here.

Documentation

- Features
- Release notes
- API Javadoc

Builds

Linux/AArch64	tar.gz (sha256)	208598806 bytes
Linux/x64	tar.gz (sha256)	210777300
macOS/AArch64	tar.gz (sha256)	204446912
macOS/x64	tar.gz (sha256)	206732010
Windows/x64	zip (sha256)	209182526

Notes

- If you have difficulty downloading any of these files please contact download-help@openjdk.org.

Feedback

If you have suggestions or encounter bugs, please submit them using the usual Java SE bug-reporting channel. Be sure to include complete version information from the output of the `java --version` command.

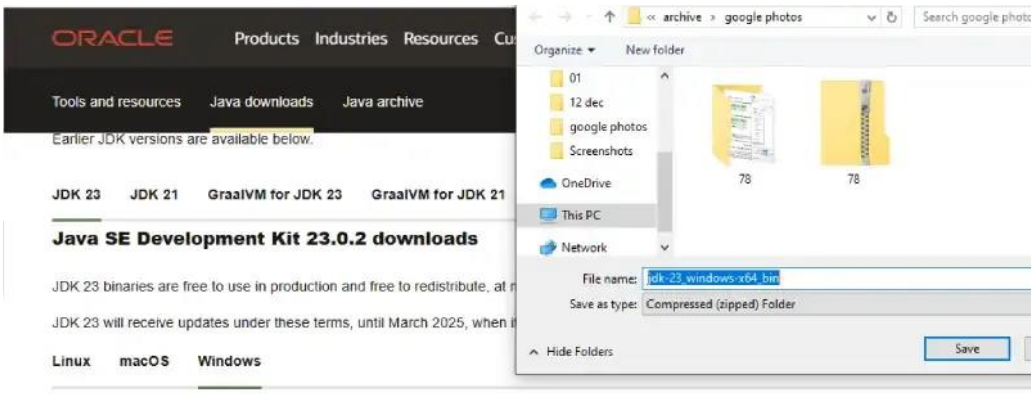
International use restrictions

Due to limited intellectual property protection and enforcement in certain countries, the source code may only be distributed to an authorized list of countries. You will not be able to access the source code if you are downloading from a country that is not on this list. We are continuously reviewing this list for addition of other countries.

© 2025 Oracle Corporation and/or its affiliates
Terms of Use · Privacy · Trademarks

Step 2: Select the Appropriate Version

As of 2025, the latest stable versions are **JDK 23** (SE) and **JDK 21** (LTS). Select the compatible version as per your operating system (Windows, Mac or Linux)



Earlier JDK versions are available below.

Java SE Development Kit 23.0.2 downloads

JDK 23 binaries are free to use in production and free to redistribute, at

JDK 23 will receive updates under these terms, until March 2025, when

File name: jdk-23_windows-x64_bin

Save as type: Compressed (zipped) Folder

Step 3: License & Agreement

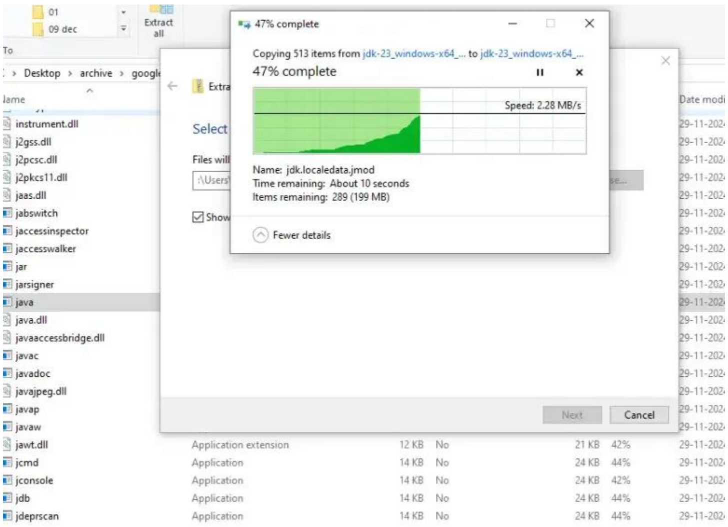
Go through all the License and Agreement before downloading (from Oracle website), it will not ask if you'll download it from OpenJDK website.

Install JDK on Windows

Follow the below steps to install JDK on Windows environment *i.e.* **Windows 7, Windows 8, Windows 8.1, Windows 10**, and **Windows 11**.

Step 1: Run the Java Development Kit (JDK) Installer

Locate the downloaded .exe file **(e.g. jdk-23-windows-x64_bin.exe)** and make the double click to begin the Installation process. Follow the installation wizard prompts to complete the installation process.



47% complete

Copying 513 items from jdk-23_windows-x64_... to jdk-23_windows-x64_...

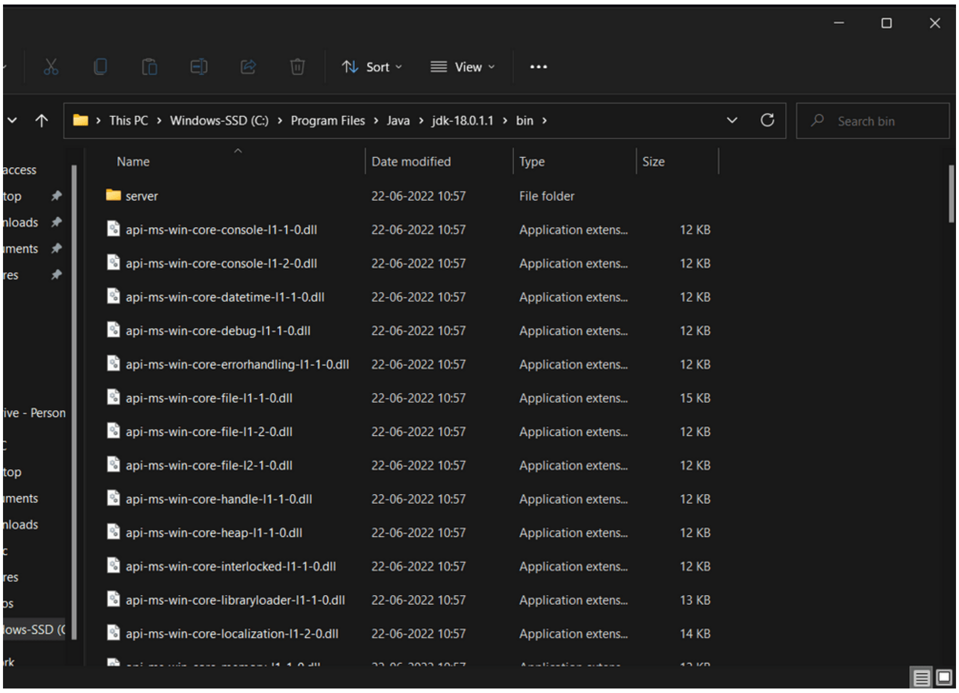
47% complete

Speed: 2.28 MB/s

Name: jdk.localedata.jmod
Time remaining: About 10 seconds
Items remaining: 289 (199 MB)

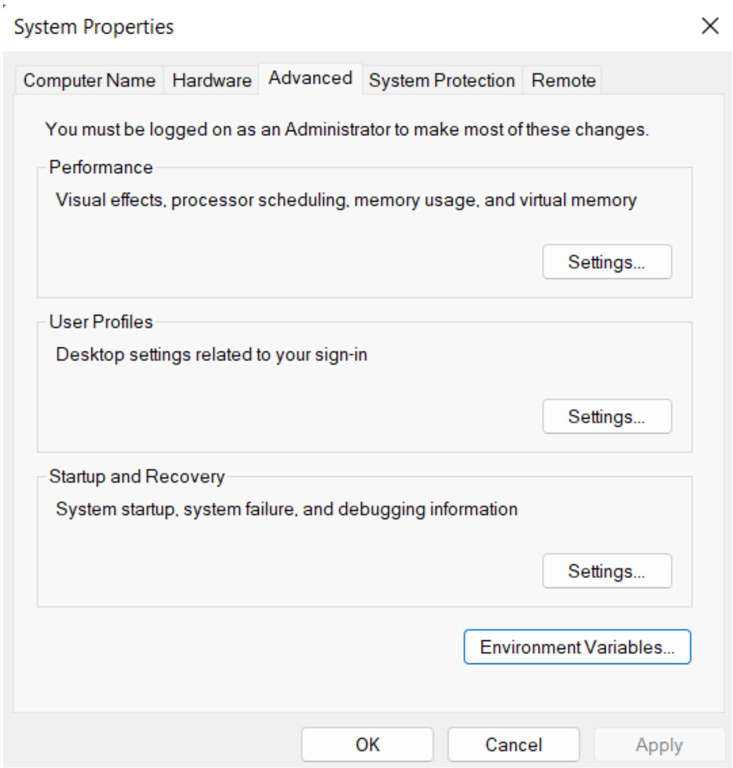
Step 2: Setup the Environment Variables

Once the installation gets completed, you need to configure environment variables to notify the system about the directory in which the **JDK files** are located.



Name	Date modified	Type	Size
server	22-06-2022 10:57	File folder	
api-ms-win-core-console-l1-1-0.dll	22-06-2022 10:57	Application extens...	12 KB
api-ms-win-core-console-l1-2-0.dll	22-06-2022 10:57	Application extens...	12 KB
api-ms-win-core-datetime-l1-1-0.dll	22-06-2022 10:57	Application extens...	12 KB
api-ms-win-core-debug-l1-1-0.dll	22-06-2022 10:57	Application extens...	12 KB
api-ms-win-core-errorhandling-l1-1-0.dll	22-06-2022 10:57	Application extens...	12 KB
api-ms-win-core-file-l1-1-0.dll	22-06-2022 10:57	Application extens...	15 KB
api-ms-win-core-file-l1-2-0.dll	22-06-2022 10:57	Application extens...	12 KB
api-ms-win-core-file-l2-1-0.dll	22-06-2022 10:57	Application extens...	12 KB
api-ms-win-core-handle-l1-1-0.dll	22-06-2022 10:57	Application extens...	12 KB
api-ms-win-core-heap-l1-1-0.dll	22-06-2022 10:57	Application extens...	12 KB
api-ms-win-core-interlocked-l1-1-0.dll	22-06-2022 10:57	Application extens...	12 KB
api-ms-win-core-libraryloader-l1-1-0.dll	22-06-2022 10:57	Application extens...	13 KB
api-ms-win-core-localization-l1-2-0.dll	22-06-2022 10:57	Application extens...	14 KB

Step 2.1: To set the Environment Variables, you need to search Environment Variables in the Task Bar and click on **"Edit the system environment variables"**.



System Properties

Computer Name | Hardware | Advanced | System Protection | Remote

You must be logged on as an Administrator to make most of these changes.

Performance

Visual effects, processor scheduling, memory usage, and virtual memory

Settings...

User Profiles

Desktop settings related to your sign-in

Settings...

Startup and Recovery

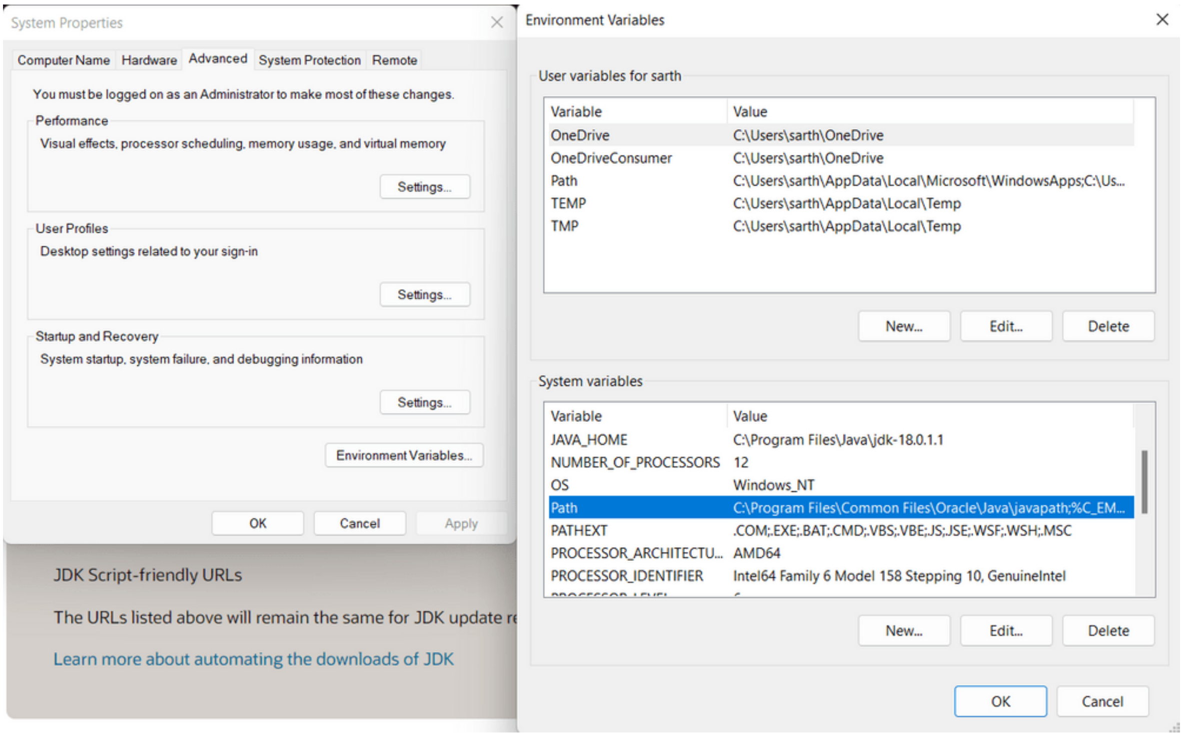
System startup, system failure, and debugging information

Settings...

Environment Variables...

OK | Cancel | Apply

Step 2.2: Under the **Advanced** section, Click on **"Environment Variables"**.



You must be logged on as an Administrator to make most of these changes.

Visual effects, processor scheduling, memory usage, and virtual memory

Desktop settings related to your sign-in

System startup, system failure, and debugging information

JDK Script-friendly URLs

The URLs listed above will remain the same for JDK update r

Learn more about automating the downloads of JDK

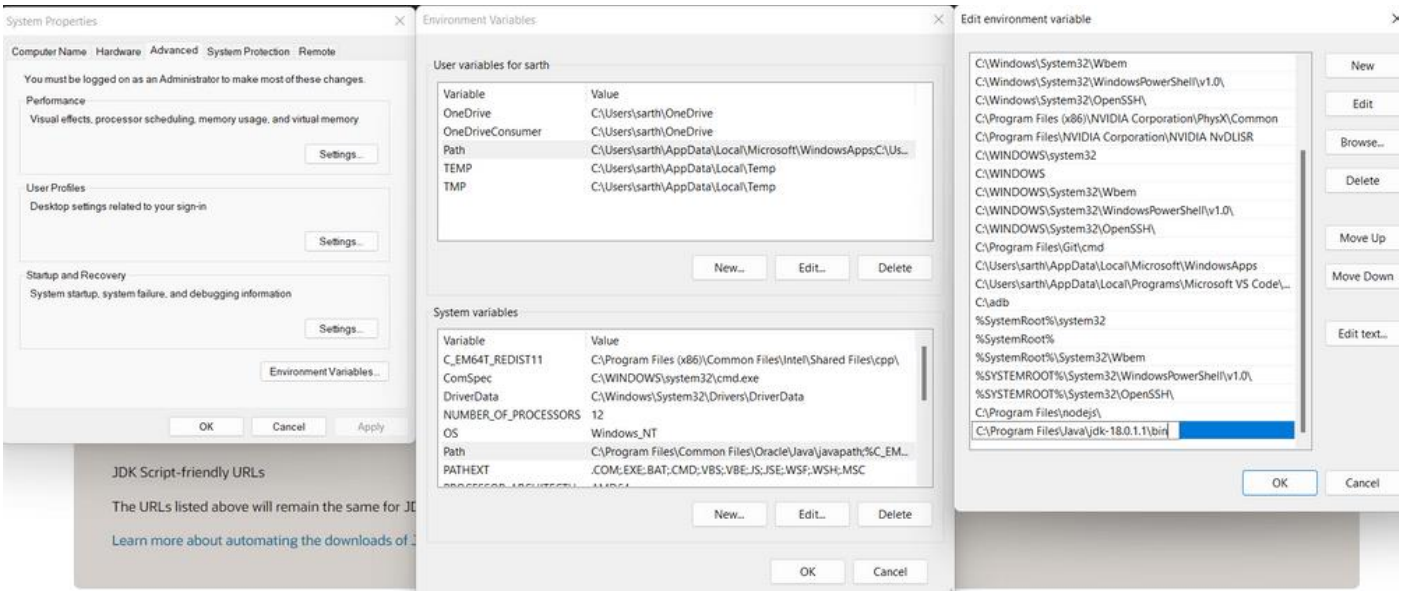
User variables for sarth

Variable	Value
OneDrive	C:\Users\sarth\OneDrive
OneDriveConsumer	C:\Users\sarth\OneDrive
Path	C:\Users\sarth\AppData\Local\Microsoft\WindowsApps;C:\Us...
TEMP	C:\Users\sarth\AppData\Local\Temp
TMP	C:\Users\sarth\AppData\Local\Temp

System variables

Variable	Value
JAVA_HOME	C:\Program Files\Java\jdk-18.0.1.1
NUMBER_OF_PROCESSORS	12
OS	Windows_NT
Path	C:\Program Files\Common Files\Oracle\Java\javapath;%C_EM...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTU...	AMD64
PROCESSOR_IDENTIFIER	Intel64 Family 6 Model 158 Stepping 10, GenuineIntel

Step 2.3: Under **System variables**, select the **"Path"** variable and click on **"Edit"**. Click on **"New"** then paste the Path Address i.e. **C:\Program Files\Java\jdk-**{YOUR_JDK_VERSION}**\bin**. Click on **"OK"**.



You must be logged on as an Administrator to make most of these changes.

Visual effects, processor scheduling, memory usage, and virtual memory

Desktop settings related to your sign-in

System startup, system failure, and debugging information

JDK Script-friendly URLs

The URLs listed above will remain the same for JI

Learn more about automating the downloads of J

User variables for sarth

Variable	Value
OneDrive	C:\Users\sarth\OneDrive
OneDriveConsumer	C:\Users\sarth\OneDrive
Path	C:\Users\sarth\AppData\Local\Microsoft\WindowsApps;C:\Us...
TEMP	C:\Users\sarth\AppData\Local\Temp
TMP	C:\Users\sarth\AppData\Local\Temp

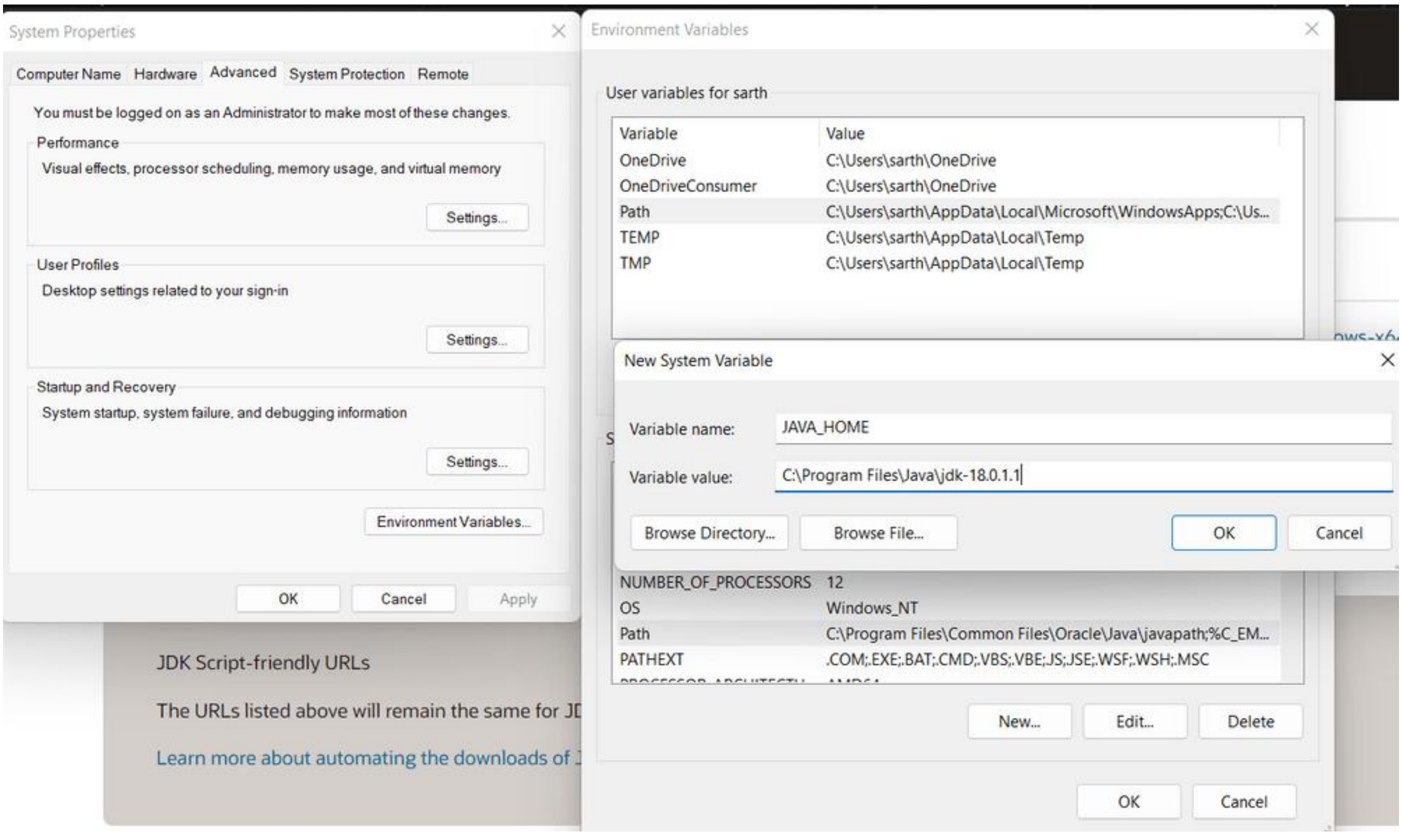
System variables

Variable	Value
C_EM64T_REDIST11	C:\Program Files (x86)\Common Files\Intel\Shared Files\cpp\
ComSpec	C:\WINDOWS\system32\cmd.exe
DriverData	C:\Windows\System32\Drivers\DriverData
NUMBER_OF_PROCESSORS	12
OS	Windows_NT
Path	C:\Program Files\Common Files\Oracle\Java\javapath;%C_EM...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC

```
C:\Windows\System32\Wbem
C:\Windows\System32\WindowsPowerShell\v1.0\
C:\Windows\System32\OpenSSH\
C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common
C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR
C:\WINDOWS\system32
C:\WINDOWS
C:\WINDOWS\System32\Wbem
C:\WINDOWS\System32\WindowsPowerShell\v1.0\
C:\WINDOWS\System32\OpenSSH\
C:\Program Files\Git\cmd
C:\Users\sarth\AppData\Local\Microsoft\WindowsApps
C:\Users\sarth\AppData\Local\Programs\Microsoft VS Code\...
C:\adb
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
%SYSTEMROOT%\System32\OpenSSH\
C:\Program Files\nodejs\
C:\Program Files\Java\jdk-18.0.1.1\bin
```

Step 2.4: Now, in the **Environment Variables** dialogue, under **System variables**, click on **"New"** and then under **Variable name: JAVA_HOME** and **Variable value:** paste address i.e.

C:\Program Files\Java\jdk-{YOUR_JDK_VERSION}. Click on **OK => OK => OK**.



You must be logged on as an Administrator to make most of these changes.

Performance
Visual effects, processor scheduling, memory usage, and virtual memory

User Profiles
Desktop settings related to your sign-in

Startup and Recovery
System startup, system failure, and debugging information

JDK Script-friendly URLs

The URLs listed above will remain the same for JD

Learn more about automating the downloads of J

User variables for sarth

Variable	Value
OneDrive	C:\Users\sarth\OneDrive
OneDriveConsumer	C:\Users\sarth\OneDrive
Path	C:\Users\sarth\AppData\Local\Microsoft\WindowsApps;C:\Us...
TEMP	C:\Users\sarth\AppData\Local\Temp
TMP	C:\Users\sarth\AppData\Local\Temp

Variable name: JAVA_HOME

Variable value: C:\Program Files\Java\jdk-18.0.1.1

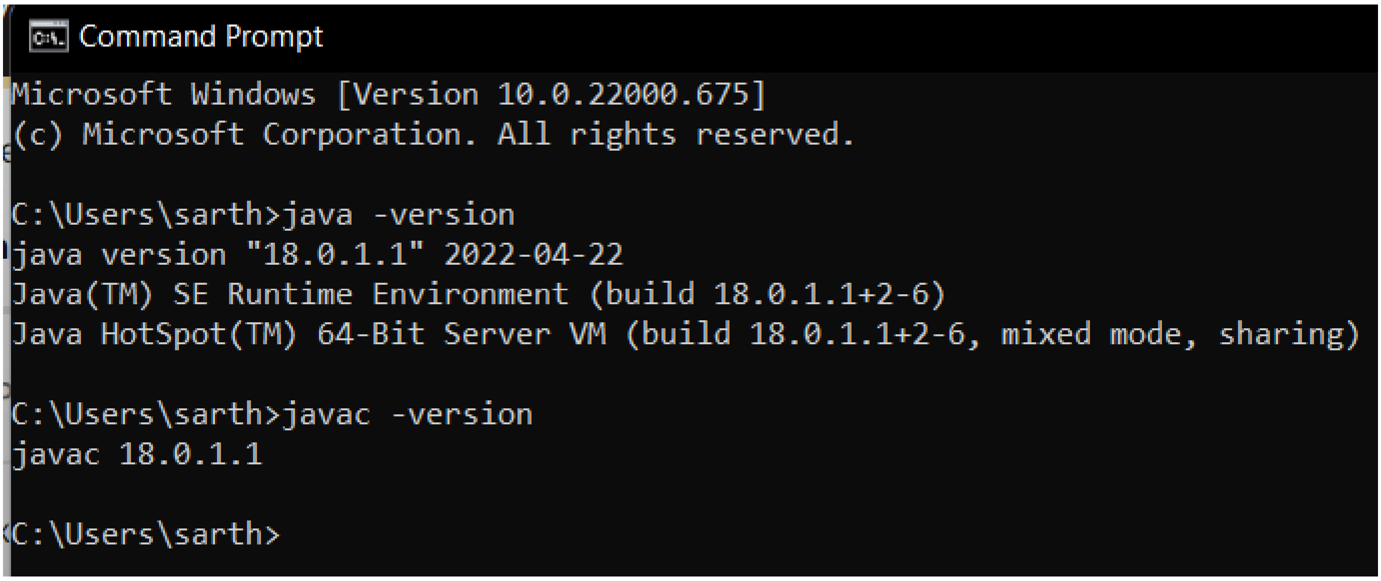
NUMBER_OF_PROCESSORS	12
OS	Windows_NT
Path	C:\Program Files\Common Files\Oracle\Java\javapath;%C_EM...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC

Step 3: Check the Java Version

Open **Command Prompt** and enter the following commands:

java -version

javac -version



```
Microsoft Windows [Version 10.0.22000.675]
(c) Microsoft Corporation. All rights reserved.

C:\Users\sarth>java -version
java version "18.0.1.1" 2022-04-22
Java(TM) SE Runtime Environment (build 18.0.1.1+2-6)
Java HotSpot(TM) 64-Bit Server VM (build 18.0.1.1+2-6, mixed mode, sharing)

C:\Users\sarth>javac -version
javac 18.0.1.1

C:\Users\sarth>
```

Download AndroidStudioAnd Android Sdk:

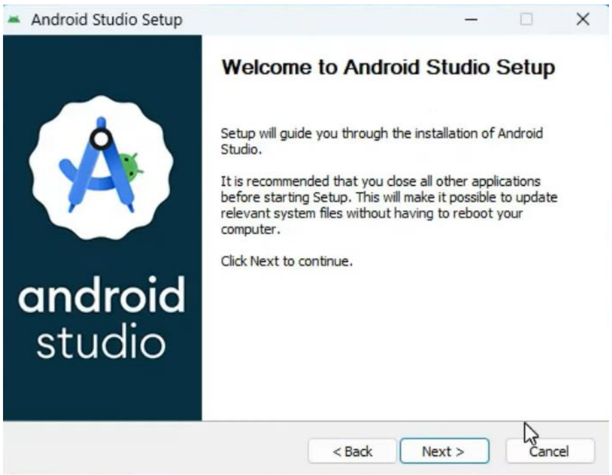
Installation on Windows Platform

Requirements:

- 3 GB RAM minimum, 8 GB RAM recommended
- 1280 x 800 minimum screen resolution
- Microsoft® Windows® 7/8/10 (32- or 64-bit)

Installation

1. Launch android-studio.exe to start the installation process. The installer responds by presenting the Android Studio Setup dialog box as shown. Click **Next** to go the **Android Virtual Device** (AVD) screen.



Android Studio Setup

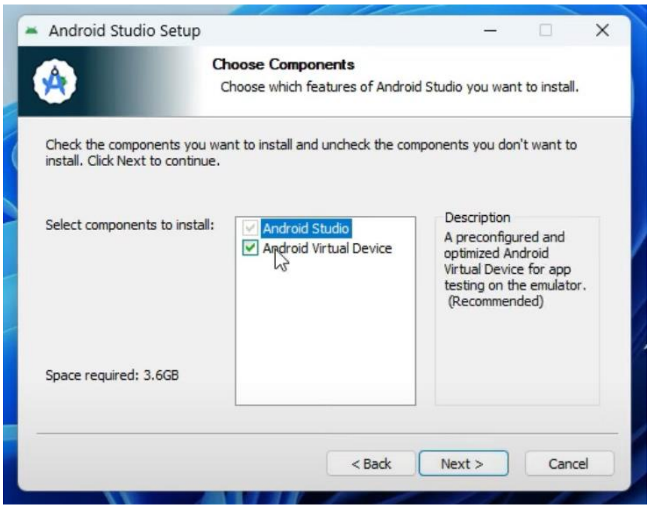
Welcome to Android Studio Setup

Setup will guide you through the installation of Android Studio.

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

Click Next to continue.

2. Choose the default settings and click **Next**.



Android Studio Setup

Choose Components

Choose which features of Android Studio you want to install.

Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.

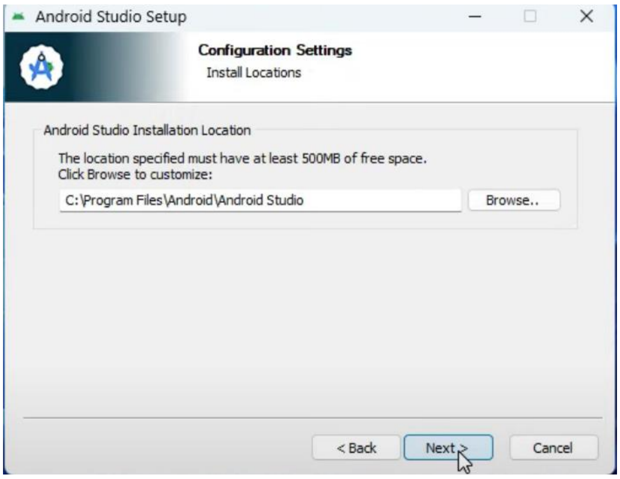
Select components to install:

☑ Android Studio
☑ Android Virtual Device

Description
A preconfigured and optimized Android Virtual Device for app testing on the emulator. (Recommended)

Space required: 3.6GB

3. After clicking Next,the **Configuration Settings** panel will show up, where the place to store android files have to be specified. After providing the location, click **Next**.



Android Studio Setup

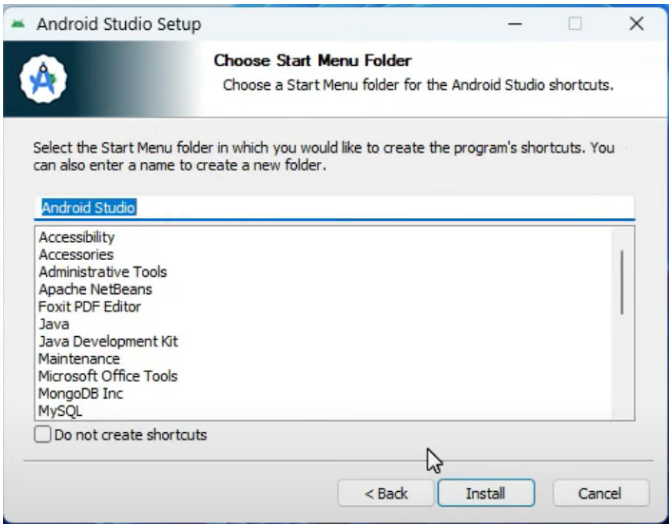
Configuration Settings
Install Locations

Android Studio Installation Location

The location specified must have at least 500MB of free space.
Click Browse to customize:

C:\Program Files\Android\Android Studio

4. On clicking Next, **Choose Start Menu Folder panel** shows up. Keep the default setting and click **Install**.



Android Studio Setup

Choose Start Menu Folder
Choose a Start Menu folder for the Android Studio shortcuts.

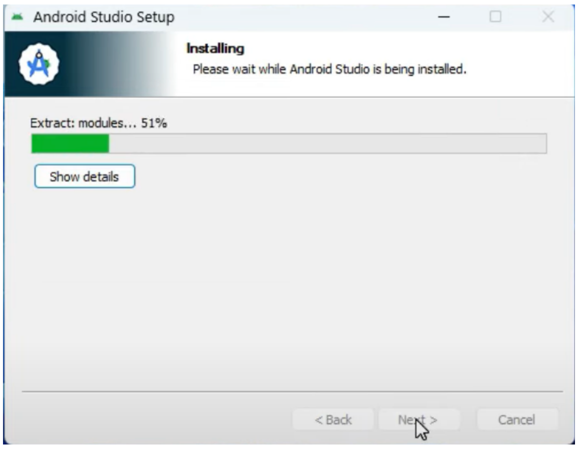
Select the Start Menu folder in which you would like to create the program's shortcuts. You can also enter a name to create a new folder.

Android Studio

Accessibility
Accessories
Administrative Tools
Apache NetBeans
Foxit PDF Editor
Java
Java Development Kit
Maintenance
Microsoft Office Tools
MongoDB Inc
MySQL

□ Do not create shortcuts

5. The following Installing panel appears:

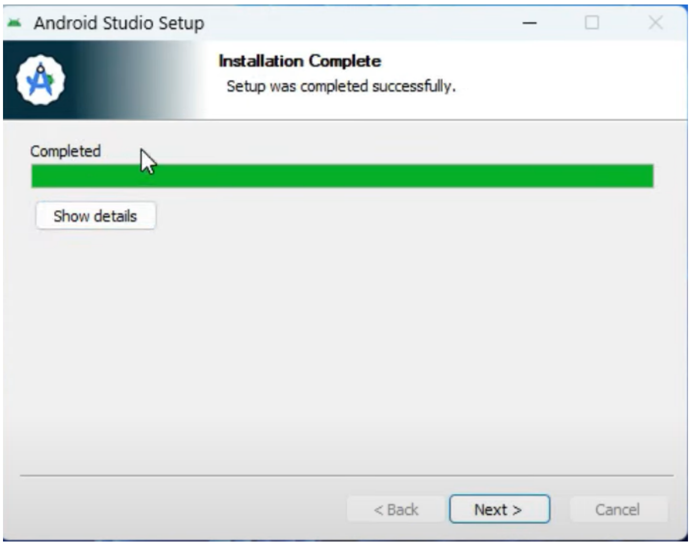


Android Studio Setup

Installing
Please wait while Android Studio is being installed.

Extract: modules... 51%

6. Clicking **Show details** shows the names of files being installed and other activities. When installation finishes, the Installation Complete panel appears.



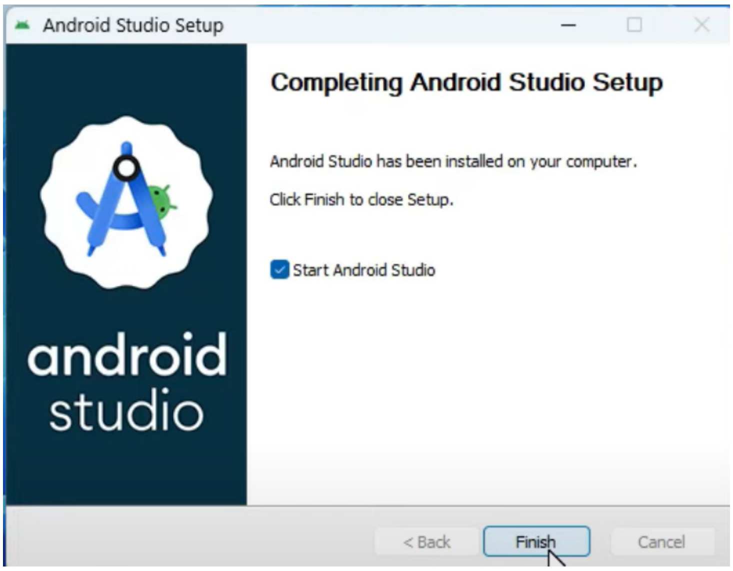
Android Studio Setup

Installation Complete

Setup was completed successfully.

Completed

7. To complete the installation, leave the **Start Android Studio** box checked and click **Finish**.



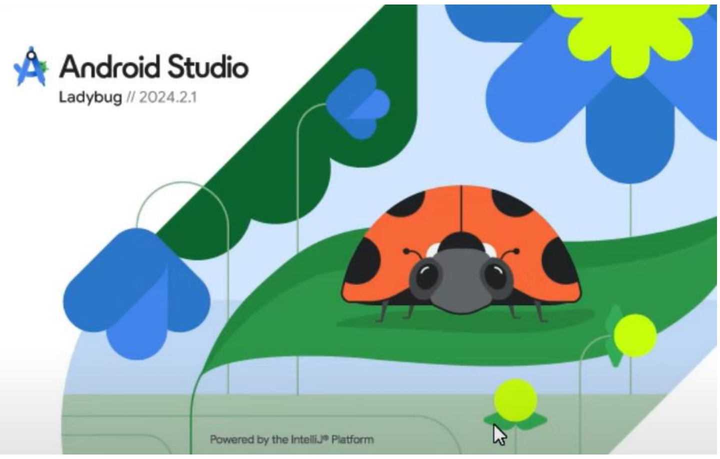
Android Studio Setup

Completing Android Studio Setup

Android Studio has been installed on your computer.

Click Finish to close Setup.

☑ Start Android Studio



Practical No:3

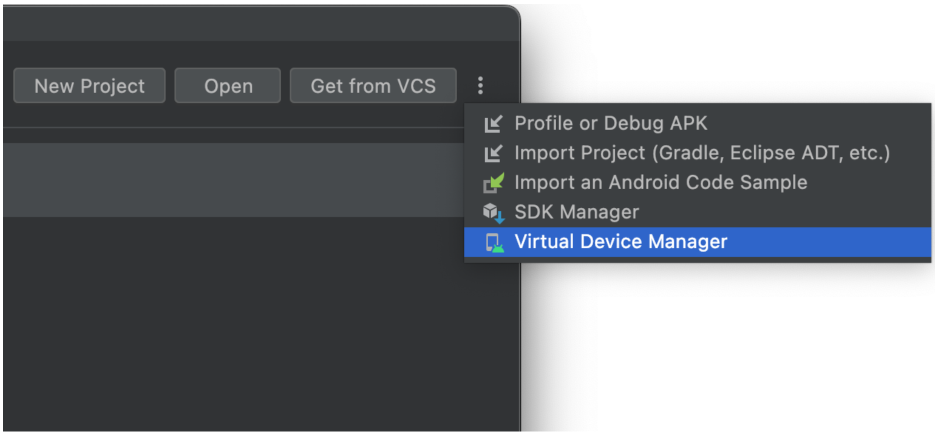
Aim :

Configure android development tools (ADT) plug-in and create android virtual device.

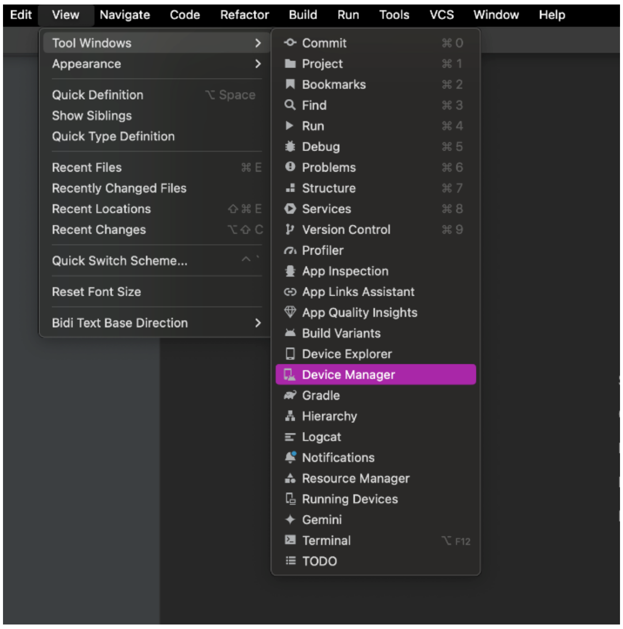
Create and manage virtual devices

An Android Virtual Device (AVD) is a configuration that defines the characteristics of an Android phone, tablet, Wear OS, Android TV, or Automotive OS device that you want to simulate in the Android Emulator. The Device Manager is a tool you can launch from Android Studio that helps you create and manage AVDs. To open the new **Device Manager**, do one of the following:

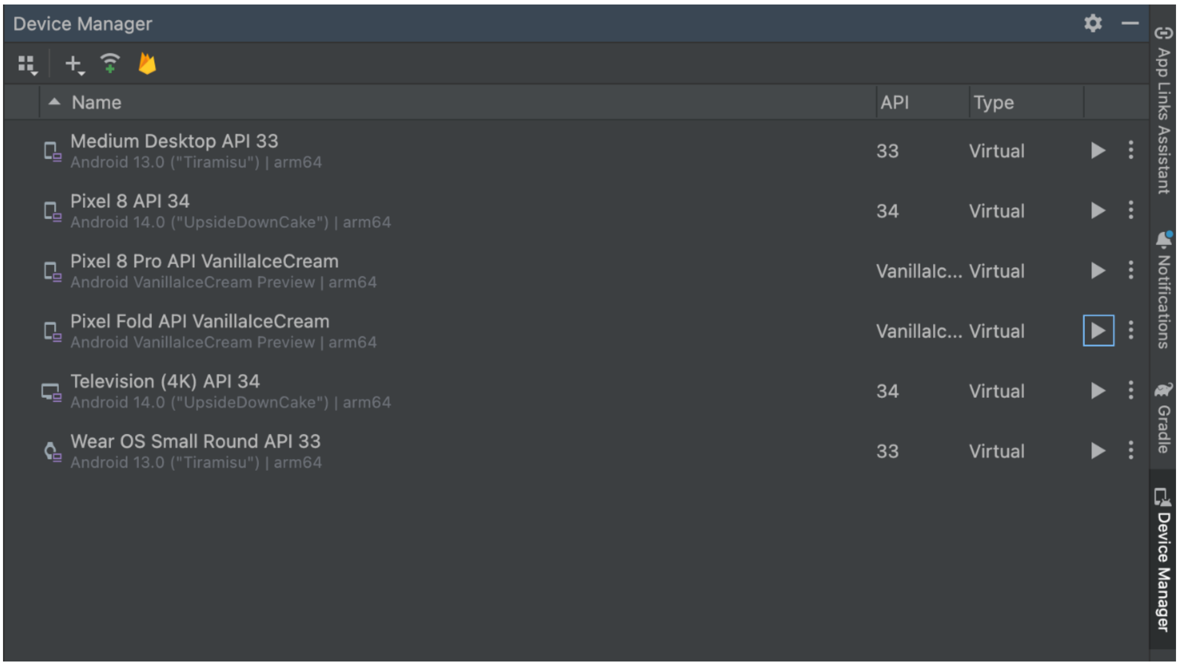
- From the Android Studio Welcome screen, select **More Actions** > **Virtual Device Manager**.



- After opening a project, select **View** > **Tool Windows** > **Device Manager** from the main menu bar, then click the **+**, and then click **Create Virtual Device**.



After creating your devices, you will be able to see a list of all the devices on the device manager panel.



Device Manager

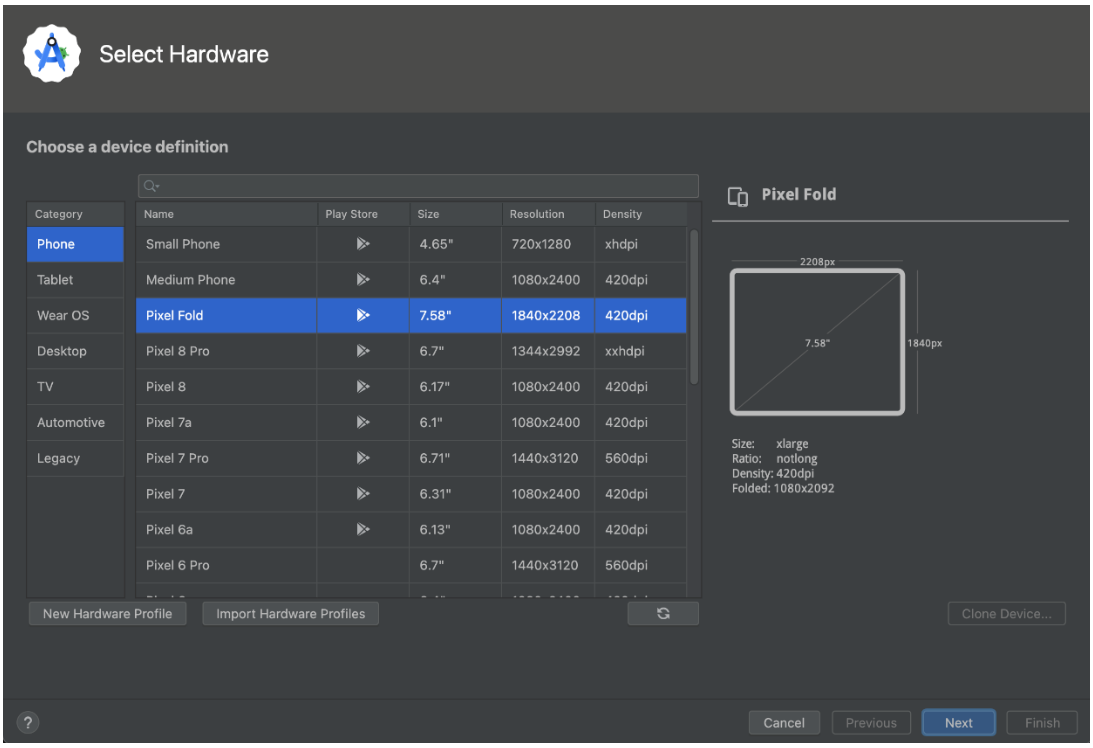
Name	API	Type
Medium Desktop API 33 Android 13.0 ("Tiramisu") | arm64	33	Virtual
Pixel 8 API 34 Android 14.0 ("UpsideDownCake") | arm64	34	Virtual
Pixel 8 Pro API VanillaIceCream Android VanillaIceCream Preview | arm64	Vanillalc...	Virtual
Pixel Fold API VanillaIceCream Android VanillaIceCream Preview | arm64	Vanillalc...	Virtual
Television (4K) API 34 Android 14.0 ("UpsideDownCake") | arm64	34	Virtual
Wear OS Small Round API 33 Android 13.0 ("Tiramisu") | arm64	33	Virtual

Create an AVD

To create a new AVD:

1. Open the Device Manager.
2. Click **Create Device**.

The **Select Hardware** window appears.



Select Hardware

Choose a device definition

Category
Phone
Tablet
Wear OS
Desktop
TV
Automotive
Legacy

Name	Play Store	Size	Resolution	Density
Small Phone	▶	4.65"	720x1280	xhdpi
Medium Phone	▶	6.4"	1080x2400	420dpi
Pixel Fold	▶	7.58"	1840x2208	420dpi
Pixel 8 Pro	▶	6.7"	1344x2992	xxhdpi
Pixel 8	▶	6.17"	1080x2400	420dpi
Pixel 7a	▶	6.1"	1080x2400	420dpi
Pixel 7 Pro	▶	6.71"	1440x3120	560dpi
Pixel 7	▶	6.31"	1080x2400	420dpi
Pixel 6a	▶	6.13"	1080x2400	420dpi
Pixel 6 Pro		6.7"	1440x3120	560dpi

Pixel Fold

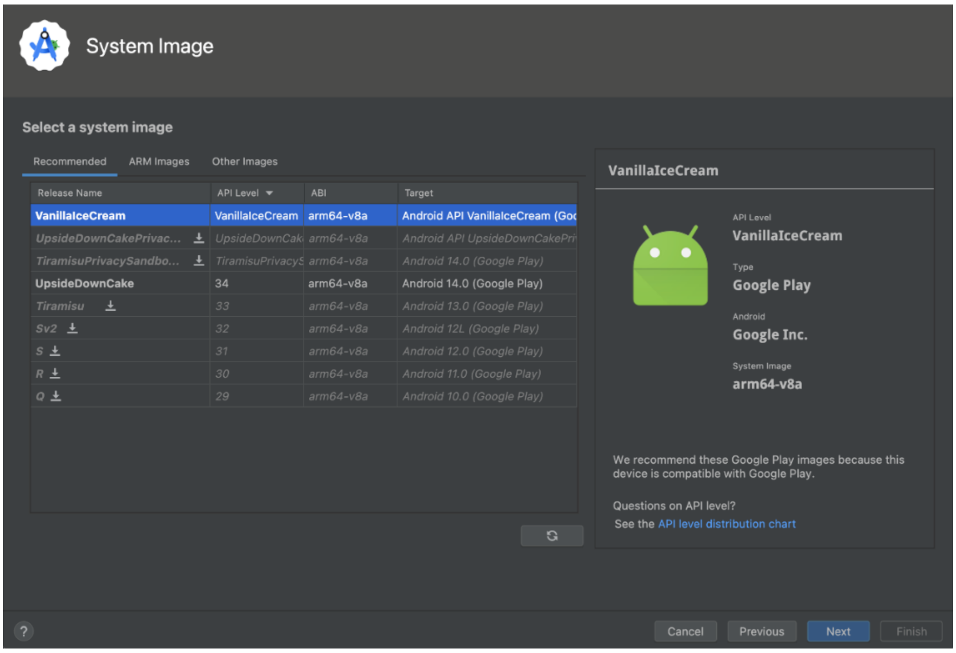
Size: xlarge
Ratio: notlong
Density: 420dpi
Folded: 1080x2092

Notice that only some hardware profiles include **Play Store**. These profiles are fully CTS compliant and might use system images that include the Play Store app.

3. Select a hardware profile, then click **Next**.

If you don't see the hardware profile you want, you can create or import a hardware profile, as described in other sections on this page.

The **System Image** window appears.



System Image

Select a system image

Recommended | ARM Images | Other Images

Release Name	API Level	ABI	Target
VanillaIceCream	VanillaIceCream	arm64-v8a	Android API VanillaIceCream (Goo
UpsideDownCakePrivac... ⤓	UpsideDownCak	arm64-v8a	Android API UpsideDownCakePri
TiramisuPrivacySandbo... ⤓	TiramisuPrivacyS	arm64-v8a	Android 14.0 (Google Play)
UpsideDownCake	34	arm64-v8a	Android 14.0 (Google Play)
Tiramisu ⤓	33	arm64-v8a	Android 13.0 (Google Play)
Sv2 ⤓	32	arm64-v8a	Android 12L (Google Play)
S ⤓	31	arm64-v8a	Android 12.0 (Google Play)
R ⤓	30	arm64-v8a	Android 11.0 (Google Play)
Q ⤓	29	arm64-v8a	Android 10.0 (Google Play)

VanillaIceCream

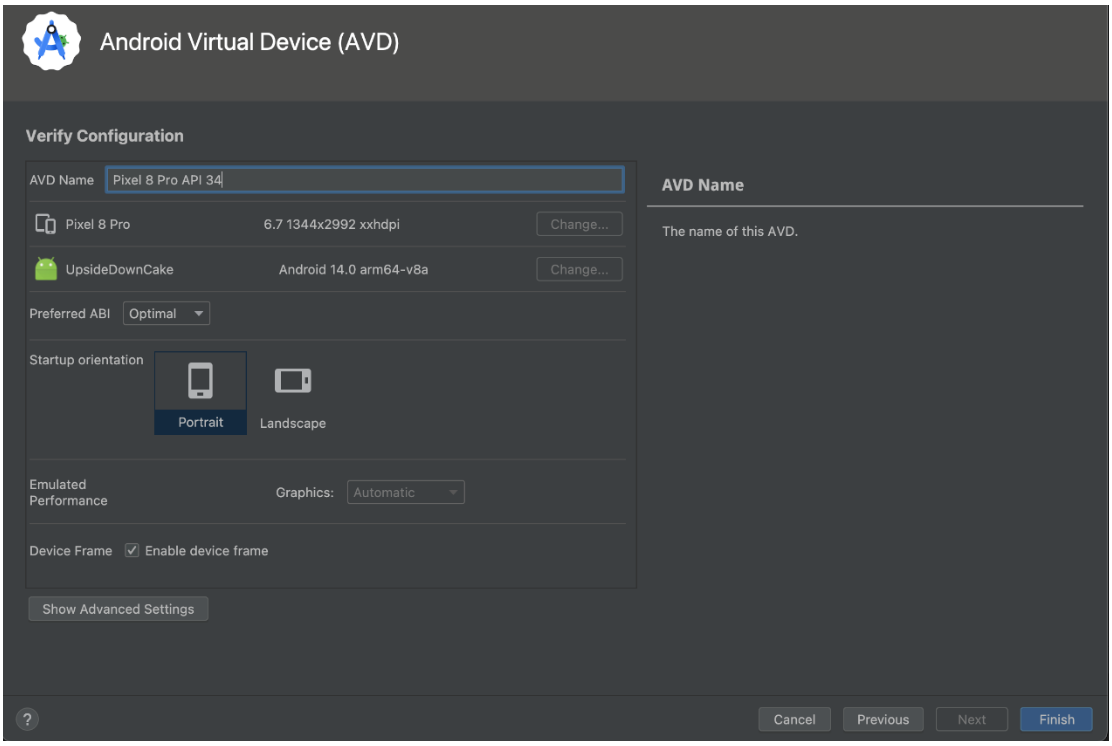
API Level: VanillaIceCream
Type: Google Play
Android: Google Inc.
System Image: arm64-v8a

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

Questions on API level?
See the API level distribution chart

4. Select the system image for a particular API level, and then click **Next**.

The **Verify Configuration** window appears.



Android Virtual Device (AVD)

Verify Configuration

AVD Name: Pixel 8 Pro API 34

Pixel 8 Pro — 6.7 1344x2992 xxhdpi — Change...

UpsideDownCake — Android 14.0 arm64-v8a — Change...

Preferred ABI: Optimal

Startup orientation: Portrait | Landscape

Emulated Performance — Graphics: Automatic

Device Frame: ☑ Enable device frame

Show Advanced Settings

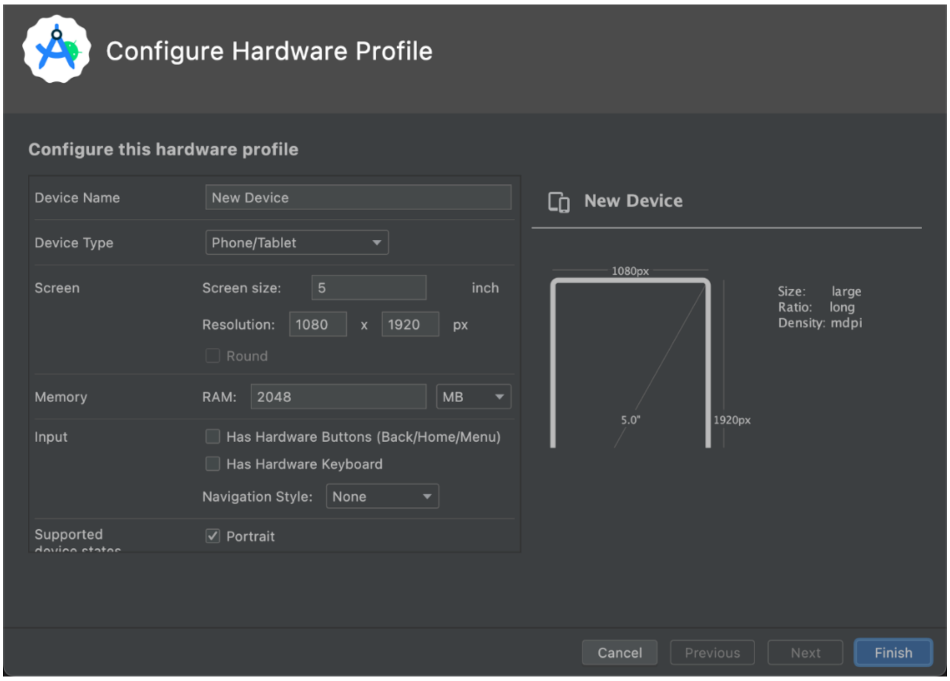
AVD Name

The name of this AVD.

5. Change the AVD properties as needed, and then click **Finish**.

To create a new hardware profile from the beginning:

1. In the Select Hardware window, click **New Hardware Profile**.
2. In the **Configure Hardware Profile** window, change the hardware profile properties as needed.



Configure Hardware Profile

Configure this hardware profile

Device Name	New Device				
Device Type	Phone/Tablet				
Screen	Screen size:	5			inch
	Resolution:	1080	x	1920	px
	□ Round				
Memory	RAM:	2048	MB		
Input	□ Has Hardware Buttons (Back/Home/Menu)				
	□ Has Hardware Keyboard				
	Navigation Style:	None			
Supported device states	☑ Portrait				

New Device

Size: large
Ratio: long
Density: mdpi

3. Click **Finish**.

Edit existing AVDs

You can perform the following operations on an AVD from the Device Manager's **Virtual** tab:

- To edit an AVD, click **Edit this AVD** ✎ and make your changes.
- To delete an AVD, click **Menu** ⋮ and select **Delete**.
- To show the associated AVD INI and IMG files on disk, click **Menu** ⋮ and select **Show on Disk**.
- To view AVD configuration details that you can include in bug reports to the Android Studio team, click **Menu** ⋮ and select **View Details**.

Run and stop an emulator and clear data

From the **Virtual** tab, you can perform the following operations on an emulator:

- To run an emulator that uses an AVD, click **Launch** ▶ .
- To stop a running emulator, click **Menu** ⋮ and select **Stop**.
- To clear the data for an emulator, click **Menu** ⋮ and select **Wipe Data**.

Practical No: 4

Aim :

Develop a program to display Hello World on screen.

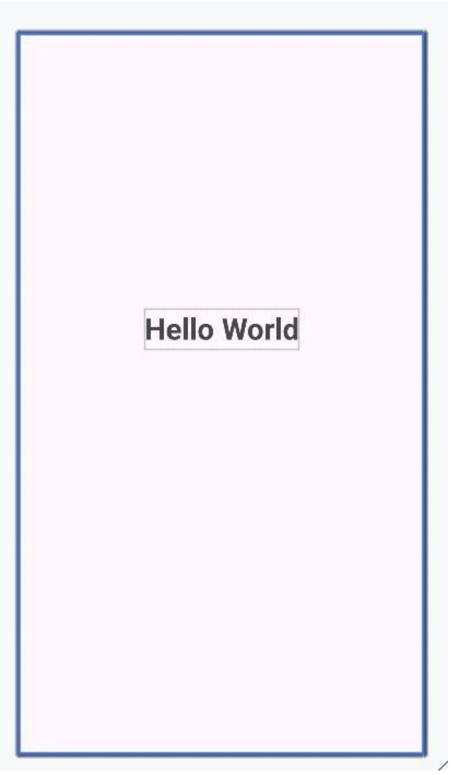
activity_main.xmlcode:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayoutxmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/main"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@drawable/myPhoto"
tools:context=".MainActivity">

<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Hello World!"
android:textColor="@color/white"
android:textSize="50dp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.319"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.768" />

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

MainActivity.java code:

```
package com.example.my_application;
import android.os.Bundle;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
EdgeToEdge.enable(this);
setContentView(R.layout.activity_main);

   }
}
```

Output:



Hello World

Practical No: 5

Aim :

Developaprogramtoimplementlinearlayout.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.appcompat.widget.LinearLayoutCompatxmlns: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/main"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:weightSum="8"
tools:context=".MainActivity">

<LinearLayout
android:layout_width="match_parent"
android:layout_height="0dp"
android:weightSum="8"
android:layout_weight="1">
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
```

```
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
</LinearLayout>

<LinearLayout
```

```
android:layout_width="match_parent"
android:layout_height="0dp"
android:weightSum="8"
android:layout_weight="1">
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
```

```
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
</LinearLayout>

<LinearLayout
android:layout_width="match_parent"
android:layout_height="0dp"
android:weightSum="8"
android:layout_weight="1">
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
```

```
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
</LinearLayout>

<LinearLayout
android:layout_width="match_parent"
android:layout_height="0dp"
android:weightSum="8"
android:layout_weight="1">
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
```

```
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
```

```
android:layout_weight="1"
android:background="@color/black"/>
</LinearLayout>

<LinearLayout
android:layout_width="match_parent"
android:layout_height="0dp"
android:weightSum="8"
android:layout_weight="1">
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
```

```
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
</LinearLayout>

<LinearLayout
android:layout_width="match_parent"
android:layout_height="0dp"
android:weightSum="8"
android:layout_weight="1">
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
```

```
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>

</LinearLayout>
<LinearLayout
android:layout_width="match_parent"
android:layout_height="0dp"
android:weightSum="8"
```

```
android:layout_weight="1">
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
```

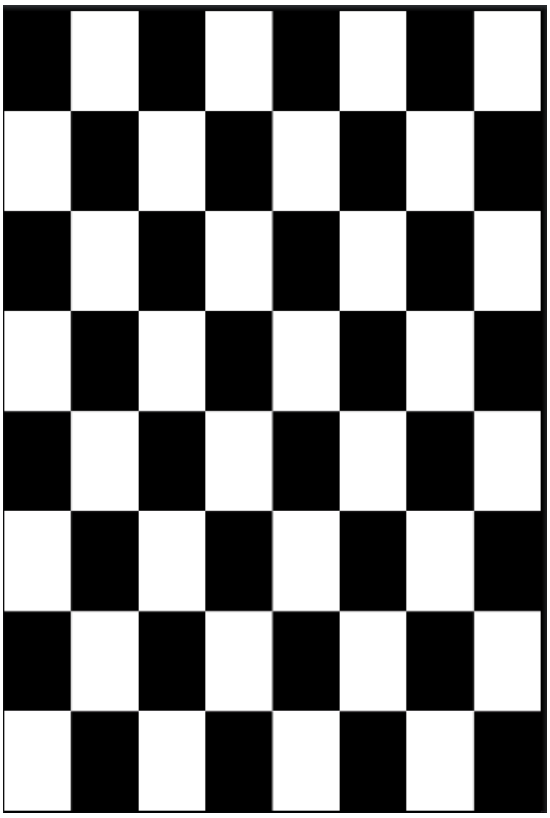
```
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>

</LinearLayout>
<LinearLayout
android:layout_width="match_parent"
android:layout_height="0dp"
android:weightSum="8"
android:layout_weight="1">
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
```

```
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/white"/>
<LinearLayout
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:background="@color/black"/>
</LinearLayout>
</androidx.appcompat.widget.LinearLayoutCompat>
```

MainActivity.java

```
package com.example.linearlayout;
import android.os.Bundle;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
EdgeToEdge.enable(this);
setContentView(R.layout.activity_main);
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
            return insets;
        });
    }
}
```

Output:



Practical No: 6

Aim :

Develop a program to implement frame layout, table layout and relative layout.

1. MainActivity.java

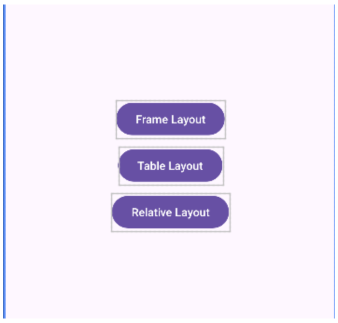
```
package com.example.layouts;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

   @Override

   protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_main);

   }

   public void openFrameLayout(View view) {

startActivity(new Intent(this, FrameLayoutActivity.class));

   }

   public void openTableLayout(View view) {

startActivity(new Intent(this, TableLayoutActivity.class));

   }

   public void openRelativeLayout(View view) {

startActivity(new Intent(this, RelativeLayoutActivity.class));

   }
}
```

2. activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:gravity="center"
android:padding="16dp">
<Button
android:text="Frame Layout"
android:onClick="openFrameLayout"
android:layout_width="wrap_content"
android:layout_height="wrap_content"/>
<Button
android:text="Table Layout"
android:onClick="openTableLayout"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="10dp"/>
<Button
android:text="Relative Layout"
android:onClick="openRelativeLayout"
android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"

android:layout_marginTop="10dp"/>

</LinearLayout>
```

Output:



FrameLayout Implementation

3. FrameLayoutActivity.java

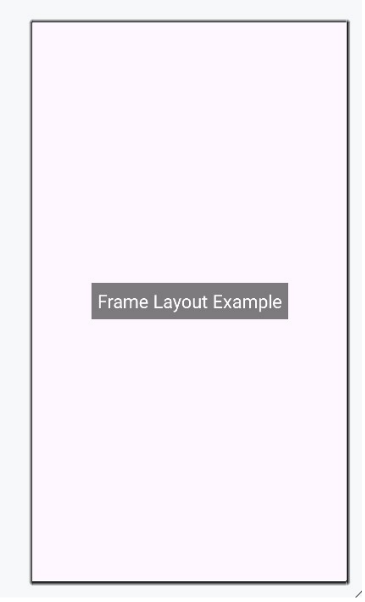
```
package com.example.layouts;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class FrameLayoutActivity extends AppCompatActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_frame_layout);

    }

}
```

4. activity_frame_layout.xml

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

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

android:layout_width="match_parent"
```

```
android:layout_height="match_parent">

<ImageView

android:layout_width="match_parent"

android:layout_height="match_parent"

android:src="@drawable/example_image"

android:scaleType="centerCrop"/>

<TextView

android:text="Frame Layout Example"

android:textSize="24sp"

android:textColor="#FFFFFF"

android:background="#80000000"

android:padding="8dp"

android:layout_gravity="center"/>

</FrameLayout>
```

Output:



TableLayout Implementation

5. TableLayoutActivity.java

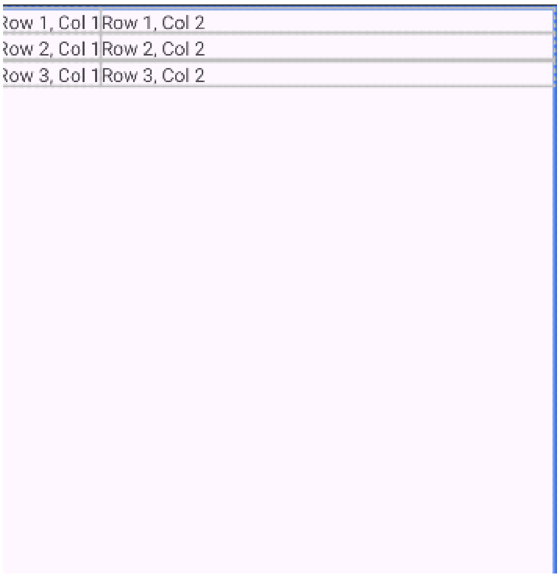
```
package com.example.layouts;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class TableLayoutActivity extends AppCompatActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_table_layout);

    }

}
```

6. activity_table_layout.xml

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

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

android:layout_width="match_parent"

android:layout_height="match_parent"

android:stretchColumns="1">

<TableRow>

<TextViewandroid:text="Row 1, Col 1" />

<TextViewandroid:text="Row 1, Col 2" />

</TableRow>

<TableRow>

<TextViewandroid:text="Row 2, Col 1" />

<TextViewandroid:text="Row 2, Col 2" />
```

```
</TableRow>

<TableRow>

<TextViewandroid:text="Row 3, Col 1" />

<TextViewandroid:text="Row 3, Col 2" />

</TableRow>

</TableLayout>
```

Output:



```
Row 1, Col 1Row 1, Col 2
Row 2, Col 1Row 2, Col 2
Row 3, Col 1Row 3, Col 2
```

RelativeLayout Implementation

7. RelativeLayoutActivity.java

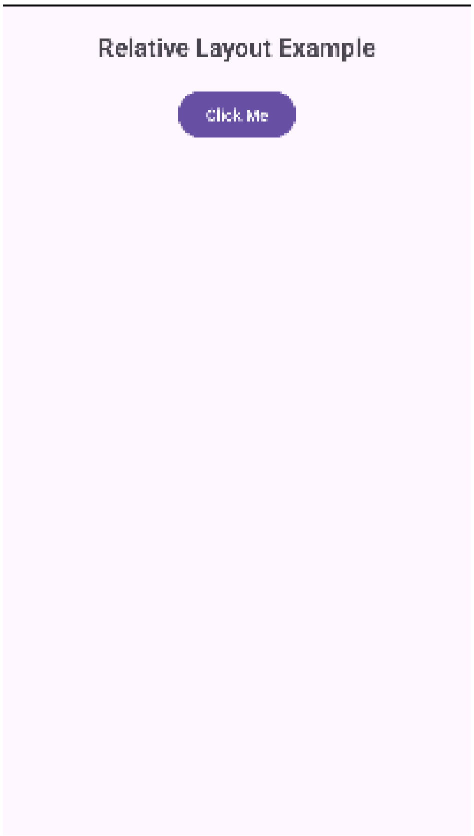
```
package com.example.layouts;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class RelativeLayoutActivity extends AppCompatActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);
setContentView(R.layout.activity_relative_layout);

    }

}
```

8. activity_relative_layout.xml

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

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

android:layout_width="match_parent"

android:layout_height="match_parent"

android:padding="20dp">

<TextView

android:id="@+id/textView1"

android:text="Relative Layout Example"

android:textSize="22sp"

android:textStyle="bold"

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:layout_centerHorizontal="true" />

<Button

android:id="@+id/button1"

android:text="Click Me"

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:layout_below="@id/textView1"

android:layout_marginTop="20dp"
```

```
android:layout_centerHorizontal="true"/>

</RelativeLayout>
```

Output:



AndroidManifest.xml:

```
<activity android:name=".FrameLayoutActivity"/>

<activity android:name=".TableLayoutActivity"/>

<activity android:name=".RelativeLayoutActivity"/>
```

Practical No: 7

Aim :

Create an application that takes the name from a text box and shows hello message along with the name entered in text box, when the user clicks the OK button

activity_main.xml

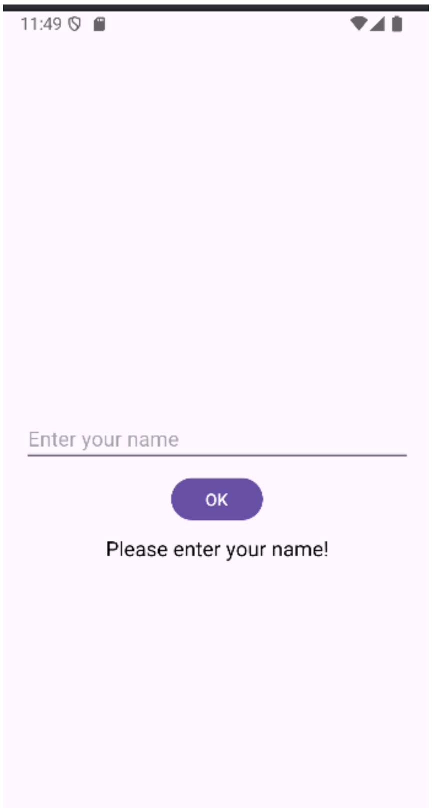
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:gravity="center"
android:padding="20dp">

<EditText
android:id="@+id/editTextName"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Enter your name"
android:textColorHint="@color/black"
android:inputType="textPersonName" />

<Button
android:id="@+id/buttonOK"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="10dp"
android:text="OK" />

<TextView
android:id="@+id/textViewMessage"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:textSize="18sp"
```

```
android:textColor="#000000"
android:layout_marginTop="10dp"/>
</LinearLayout>
```

MainActivity.java

```
package com.example.practical_7;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {

   @Override
   protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
EdgeToEdge.enable(this);
setContentView(R.layout.activity_main);
        // Get references to UI elements
EditTexteditTextName = findViewById(R.id.editTextName);
        Button buttonOK = findViewById(R.id.buttonOK);
TextViewtextViewMessage = findViewById(R.id.textViewMessage);

        // Set button click listener
buttonOK.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // Get user input
                String name = editTextName.getText().toString().trim();

                // Check if name is empty
                if (name.isEmpty()) {
textViewMessage.setText("Please enter your name!");
                } else {
textViewMessage.setText("Hello, " + name + "!");
                }
            }
        });
    }
}
```

Output:



Please enter your name!

Practical No: 8

Aim :

Create a screen that has input boxes for User Name, Password, Address, Gender (radio buttons for male and female), Age (numeric), Date of Birth (Date Picket) and a Submit button. On clicking the submit button, print all the data below the Submit Button (use any layout)

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollViewxmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:padding="20dp">

<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="vertical">
<EditText
android:id="@+id/editTextUserName"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Enter User Name" />

<EditText
android:id="@+id/editTextPassword"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Enter Password"
android:inputType="textPassword" />

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

```
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Enter Address"
android:inputType="textMultiLine"
android:minLines="3" />

<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Gender:" />
<RadioGroup
android:id="@+id/radioGroupGender"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:orientation="horizontal">
<RadioButton
android:id="@+id/radioMale"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Male" />
<RadioButton
android:id="@+id/radioFemale"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Female" />
</RadioGroup>

<EditText
android:id="@+id/editTextAge"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Enter Age"
android:inputType="number" />
```

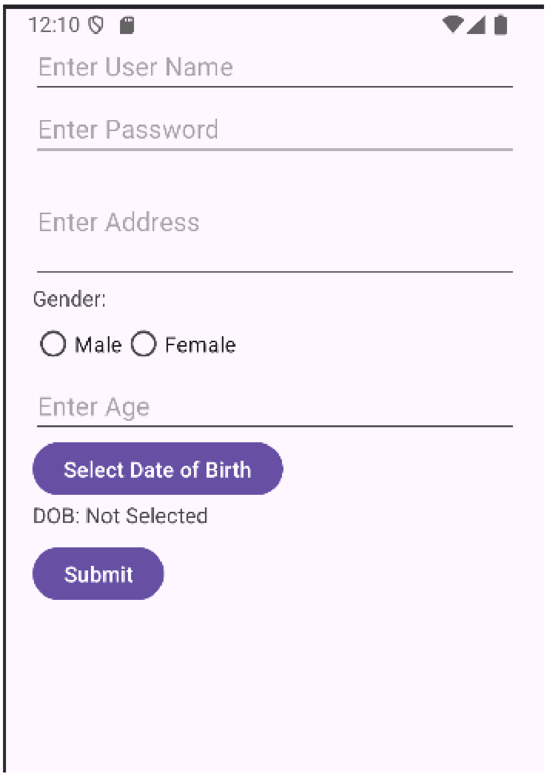
```
<Button
android:id="@+id/buttonDOB"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Select Date of Birth" />
<TextView
android:id="@+id/textViewDOB"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="DOB: Not Selected" />

<Button
android:id="@+id/buttonSubmit"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Submit"
android:layout_marginTop="10dp" />

<TextView
android:id="@+id/textViewOutput"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text=""
android:textSize="16sp"
android:textColor="#000000"
android:layout_marginTop="20dp" />
</LinearLayout>
</ScrollView>
```

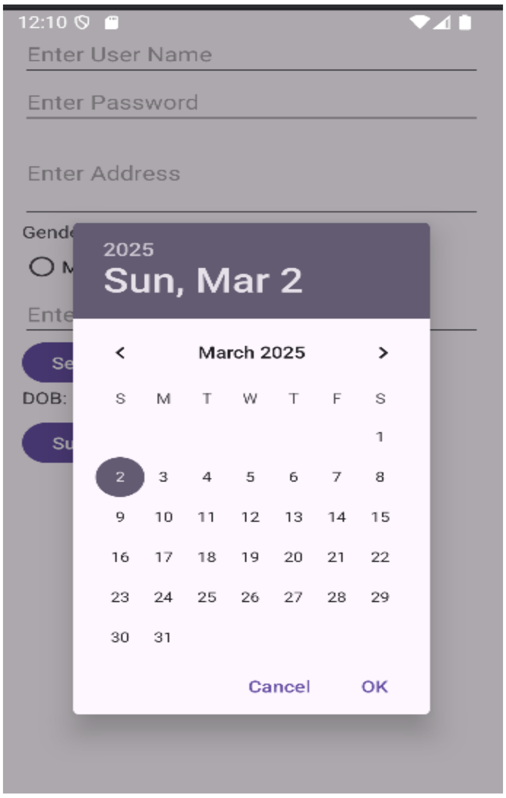
MainActivity.java

```
package com.example.practical_8;
import android.app.DatePickerDialog;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.RadioGroup;
import android.widget.TextView;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
import java.util.Calendar;
public class MainActivity extends AppCompatActivity {
    private EditTexteditTextUserName, editTextPassword, editTextAddress, editTextAge;
    private RadioGroupradioGroupGender;
    private TextViewtextViewDOB, textViewOutput;
    private Button buttonDOB, buttonSubmit;
    private String selectedDOB = "Not Selected";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
EdgeToEdge.enable(this);
setContentView(R.layout.activity_main);
editTextUserName = findViewById(R.id.editTextUserName);
editTextPassword = findViewById(R.id.editTextPassword);
editTextAddress = findViewById(R.id.editTextAddress);
editTextAge = findViewById(R.id.editTextAge);
radioGroupGender = findViewById(R.id.radioGroupGender);
textViewDOB = findViewById(R.id.textViewDOB);
textViewOutput = findViewById(R.id.textViewOutput);
buttonDOB = findViewById(R.id.buttonDOB);
buttonSubmit = findViewById(R.id.buttonSubmit);

        // Date Picker Dialog for DOB
```

```
buttonDOB.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Calendar calendar = Calendar.getInstance();
                int year = calendar.get(Calendar.YEAR);
                int month = calendar.get(Calendar.MONTH);
                int day = calendar.get(Calendar.DAY_OF_MONTH);
DatePickerDialogdatePickerDialog = new DatePickerDialog(MainActivity.this, new
DatePickerDialog.OnDateSetListener() {
                    @Override
                    public void onDateSet(DatePicker view, int selectedYear, int selectedMonth, int selectedDay) {
selectedDOB = selectedDay + "/" + (selectedMonth + 1) + "/" + selectedYear;
textViewDOB.setText("DOB: " + selectedDOB);
                    }
                }, year, month, day);
datePickerDialog.show();
            }
        });
        // Submit Button Click Event
buttonSubmit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // Get user input values
                String name = editTextUserName.getText().toString().trim();
                String password = editTextPassword.getText().toString().trim();
                String address = editTextAddress.getText().toString().trim();
                String age = editTextAge.getText().toString().trim();
                // Get selected gender
                int selectedId = radioGroupGender.getCheckedRadioButtonId();
                String gender = selectedId == R.id.radioMale? "Male" : "Female";
                // Validate fields
                if (name.isEmpty() || password.isEmpty() || address.isEmpty() || age.isEmpty()) {
textViewOutput.setText("Please fill all fields!");
                    return;
                }
                // Display the entered details
                String output = "User Name: " + name + "\n" +
                        "Password: " + password + "\n" +
                        "Address: " + address + "\n" +
                        "Gender: " + gender + "\n" +
                        "Age: " + age + "\n" +
                        "DOB: " + selectedDOB;
textViewOutput.setText(output);
            }
        });
      }
}
```

Output:





Practical No: 9

Aim :

Design an android application to create page using Intent and one Button and pass the Values from one Activity to second Activity

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:gravity="center"
android:padding="20dp">
<!-- Input Name -->
<EditText
android:id="@+id/editTextName"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Enter Your Name"
android:inputType="textPersonName"/>
<EditText
android:id="@+id/editTextAge"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Enter Your Age"
```

```
android:inputType="number"

android:layout_marginTop="10dp"/>

<Button

android:id="@+id/buttonSend"

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:text="Send Data"

android:layout_marginTop="20dp"/>

</LinearLayout>
```

MainActivity.java

```
package com.example.intentexample;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_main);

EditTexteditTextName = findViewById(R.id.editTextName);

EditTexteditTextAge = findViewById(R.id.editTextAge);
```

```
        Button buttonSend = findViewById(R.id.buttonSend);
buttonSend.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String name = editTextName.getText().toString().trim();
                String age = editTextAge.getText().toString().trim();
                Intent intent = new Intent(MainActivity.this, SecondActivity.class);
intent.putExtra("USER_NAME", name);
intent.putExtra("USER_AGE", age);
startActivity(intent);
            }
        });
    }
}
```

activity_second.xml

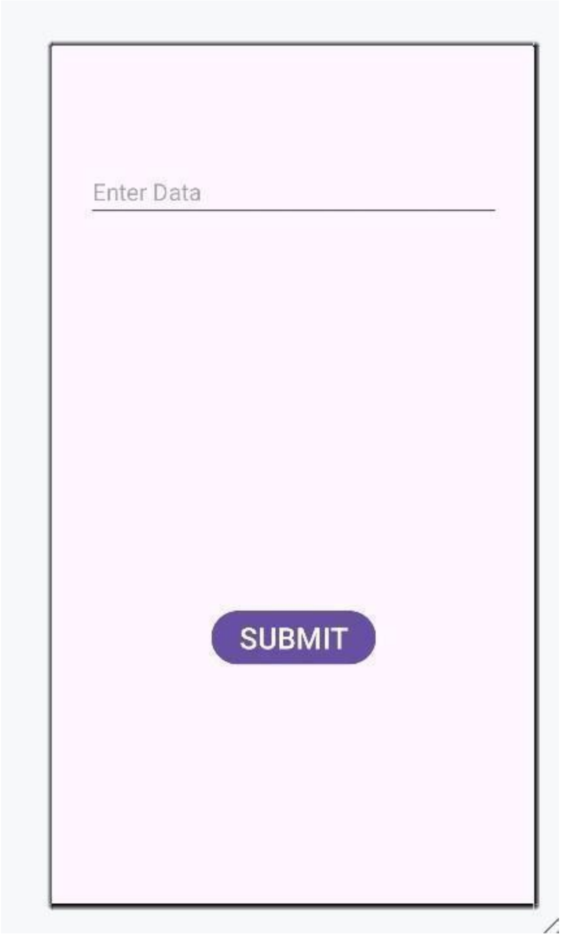
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:gravity="center"
android:padding="20dp">
<TextView
android:id="@+id/textViewResult"
```

```
android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:text="Received Data"

android:textSize="20sp"

android:textStyle="bold"/>

</LinearLayout>
```

SecondActivity.java

```
package com.example.intentexample;

import android.os.Bundle;

import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class SecondActivity extends AppCompatActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_second);

TextViewtextViewResult = findViewById(R.id.textViewResult);

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

        String age = getIntent().getStringExtra("USER_AGE");

        String message = "Name: " + name + "\nAge: " + age;

textViewResult.setText(message);

    }

}
```

Output:



Practical No: 10

Aim :

Design an android application Send SMS using Intent.

activity_main.xml (User Interface)

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:padding="20dp"
android:gravity="center">
<EditText
android:id="@+id/editTextPhone"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Enter Phone Number"
android:inputType="phone"/>
<EditText
android:id="@+id/editTextMessage"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Enter Message"
android:inputType="textMultiLine"
android:minLines="3"
android:layout_marginTop="10dp"/>
<Button
android:id="@+id/buttonSendSMS"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Send SMS"
android:layout_marginTop="20dp"/>
</LinearLayout>
```

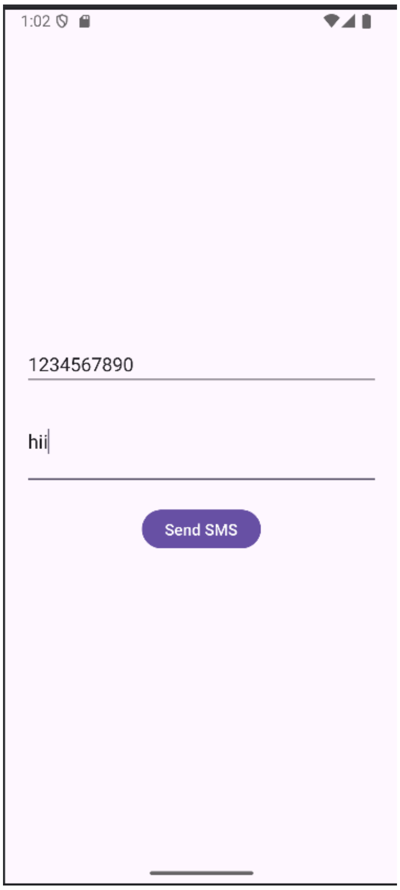
2MainActivity.java (SMS Sending Logic)

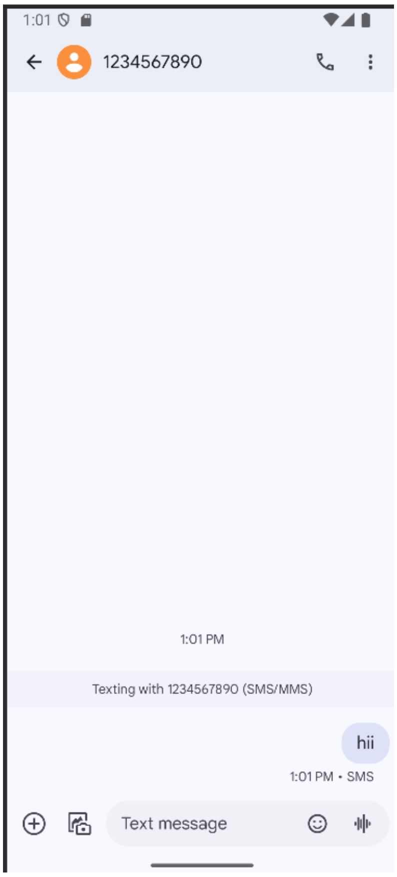
```
package com.example.smsintent;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
EditTexteditTextPhone = findViewById(R.id.editTextPhone);
EditTexteditTextMessage = findViewById(R.id.editTextMessage);
        Button buttonSendSMS = findViewById(R.id.buttonSendSMS);
buttonSendSMS.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String phoneNumber = editTextPhone.getText().toString().trim();
                String message = editTextMessage.getText().toString().trim();
                if (!phoneNumber.isEmpty() && !message.isEmpty()) {
                    // Open messaging app with pre-filled details
                    Intent intent = new Intent(Intent.ACTION_VIEW);
intent.setData(Uri.parse("sms:" + phoneNumber));
intent.putExtra("sms_body", message);
startActivity(intent);
                }
            }
        });
    }
}
```

3Register Activity in AndroidManifest.xml

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.example.smsintent">
<application
android:allowBackup="true"
android:theme="@style/Theme.SMSIntent"
android:label="SMS App">
<activity android:name=".MainActivity">
<intent-filter>
<action android:name="android.intent.action.MAIN" />
<category android:name="android.intent.category.LAUNCHER" />
</intent-filter>
</activity>
</application>
</manifest>
```

Output:





Practical No: 11

Aim :

Create an android application using Fragments.

activity-main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical">

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

<Button
        android:id="@+id/buttonFragment1"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Show Fragment 1"
android:layout_margin="8dp" />

<Button
        android:id="@+id/buttonFragment2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Show Fragment 2"
android:layout_margin="8dp" />
</LinearLayout>

<FrameLayout
```

```
android:id="@+id/fragmentContainer"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:layout_marginTop="16dp" />
</LinearLayout>
```

fragment-one.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:gravity="center"
android:padding="16dp"
android:background="#FFDDC1">
<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="This is Fragment 1"
android:textSize="24sp"
android:textStyle="bold" />
</LinearLayout>
```

fragment-two.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:gravity="center"
android:padding="16dp"
android:background="#C1E1FF">
```

```
<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="This is Fragment 2"
android:textSize="24sp"
android:textStyle="bold" />
</LinearLayout>
```

FragmentOne.java

```
package com.example.fragmentdemo;
import android.os.Bundle;
import androidx.fragment.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
public class FragmentOne extends Fragment {
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.fragment_one, container, false);
    }
}
```

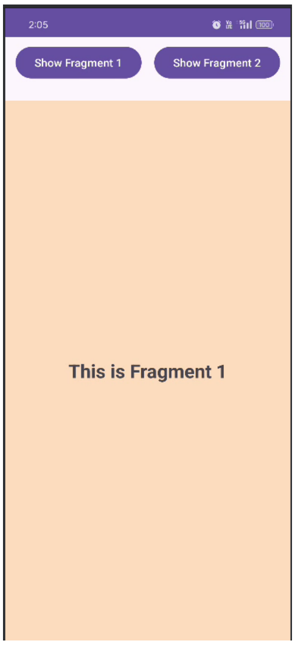
FragmentTwo.java

```
package com.example.fragmentdemo;
import android.os.Bundle;
import androidx.fragment.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
public class FragmentTwo extends Fragment {
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.fragment_two, container, false);
    }
}
```

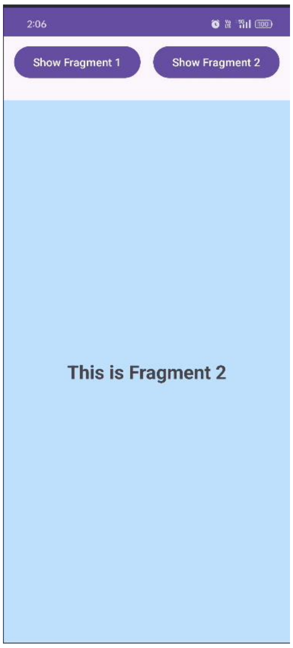
MainActivity.java

```
package com.example.fragmentdemo;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentTransaction;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
        Button buttonFragment1 = findViewById(R.id.buttonFragment1);
        Button buttonFragment2 = findViewById(R.id.buttonFragment2);
        buttonFragment1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
loadFragment(new FragmentOne());
            }
        });
        buttonFragment2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
loadFragment(new FragmentTwo());
            }
        });
        // Load the first fragment by default
loadFragment(new FragmentOne());
    }
    private void loadFragment(Fragment fragment) {
FragmentTransaction transaction = getSupportFragmentManager().beginTransaction();
transaction.replace(R.id.fragmentContainer, fragment);
transaction.commit();
    }
}
```

Output:



This is Fragment 1



This is Fragment 2

Practical No: 12

Aim :

Design an android application Using Radio button.

activity-main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:padding="16dp">

<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Select Your Favorite Programming Language:"
android:textSize="18sp"
android:layout_marginBottom="16dp" />

<RadioGroup
android:id="@+id/radioGroup"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="vertical">

<RadioButton
android:id="@+id/radioJava"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Java" />
```

```
<RadioButton
android:id="@+id/radioPython"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Python" />

<RadioButton
android:id="@+id/radioCpp"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="C++" />

<RadioButton
android:id="@+id/radioKotlin"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Kotlin" />
</RadioGroup>

<Button
android:id="@+id/submitButton"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Submit"
android:layout_marginTop="16dp" />

<TextView
android:id="@+id/resultText"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text=""
android:textSize="16sp"
android:textStyle="bold"
android:layout_marginTop="16dp"
```

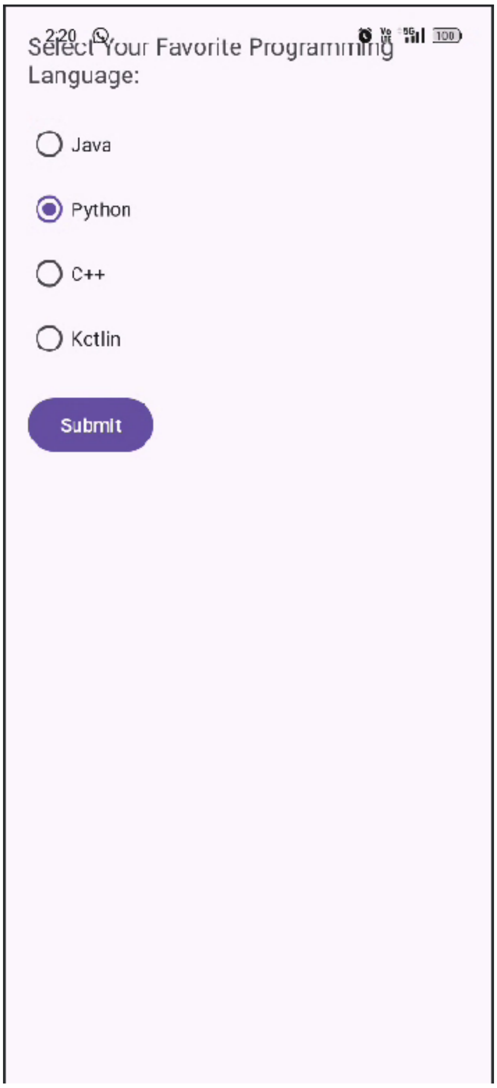
```
android:gravity="center" />

</LinearLayout>
```

MainActivity.java

```
package com.example.radiobuttondemo;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
RadioGroupradioGroup = findViewById(R.id.radioGroup);
        Button submitButton = findViewById(R.id.submitButton);
TextViewresultText = findViewById(R.id.resultText);
submitButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                int selectedId = radioGroup.getCheckedRadioButtonId();
                if (selectedId == -1) {
                    // No radio button selected
Toast.makeText(MainActivity.this, "Please select an option",Toast.LENGTH_SHORT).show();
                } else {
RadioButtonselectedRadioButton = findViewById(selectedId);
```

```
                    String selectedText = selectedRadioButton.getText().toString();
resultText.setText("You selected: " + selectedText);
                }
            }
        });
    }
}
```

Output:



2:20

Select Your Favorite Programming Language:

○ Java

● Python

○ C++

○ Kctlin

Practical No: 13

Aim :

Design an android application for menu.

activity-main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:gravity="center"
android:padding="16dp">

<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Welcome to Menu Demo"
android:textSize="20sp"
android:textStyle="bold"
android:layout_marginBottom="16dp" />

<TextView
android:id="@+id/displayText"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Select an option from the menu"
android:textSize="16sp"
android:layout_marginTop="16dp" />

</LinearLayout>
```

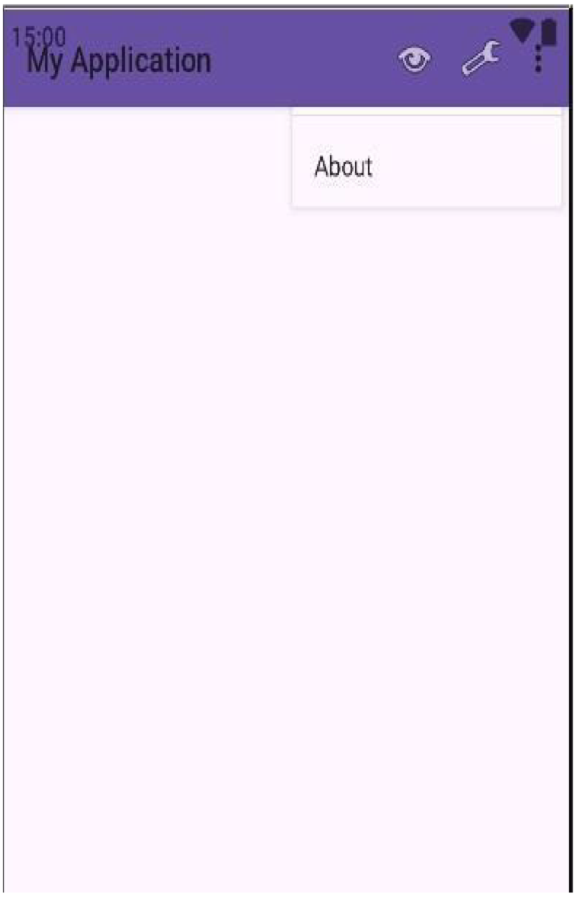
res/menu – main-menu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
<item
android:id="@+id/action_home"
android:title="Home"
android:icon="@android:drawable/ic_menu_view"
android:showAsAction="ifRoom" />

<item
android:id="@+id/action_settings"
android:title="Settings"
android:icon="@android:drawable/ic_menu_manage"
android:showAsAction="ifRoom" />

<item
android:id="@+id/action_about"
android:title="About"
android:icon="@android:drawable/ic_menu_info_details"
android:showAsAction="never" />

</menu>
```

MainActivity.java:

```
package com.example.menudemo;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {
private TextViewdisplayText;
@Override
protected void onCreate(Bundle savedInstanceState)
{ super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
displayText = findViewById(R.id.displayText);
}
@Override
public booleanonCreateOptionsMenu(Menu menu) {
// Inflate the menu
getMenuInflater().inflate(R.menu.main_menu, menu);
return true;
}
@Override
public booleanonOptionsItemSelected(MenuItem item)
{ switch (item.getItemId()) {
case R.id.action_home:
displayText.setText("Home selected");
Toast.makeText(this, "Home clicked", Toast.LENGTH_SHORT).show();
return true;
case R.id.action_settings:
displayText.setText("Settings selected");
Toast.makeText(this, "Settings clicked", Toast.LENGTH_SHORT).show();
return true;
case R.id.action_about:
displayText.setText("About selected");
Toast.makeText(this, "About clicked", Toast.LENGTH_SHORT).show();
return true;
default:
return super.onOptionsItemSelected(item);
}
}
}
```

Output :


15:00
My Application
About