

योग: कर्मसु कौशलम्



Parul[®]
University

NAAC
ACCREDITED UNIVERSITY



FACULTY OF ENGINEERING AND TECHNOLOGY

BACHELOR OF TECHNOLOGY

MOBILE APPLICATION DEVELOPMENT LABORATORY (303105379)

VI SEMESTER

Computer Science and Engineering Department

Laboratory Manual Session 2024-25

CERTIFICATE

This is to certify that

Mr. SHIVANSHU SHUKLA with enrolment no.

2203051050539 has successfully completed

his/her laboratory experiments in the **Mobile Application Development (303105379)** from the department of **COMPUTER SCIENCE & ENGINEERING** during the academic year 2024-2025.

योग: कर्मसु कौशलम्



Date of Submission:..... Staff In charge:.....

Head Of Department:.....

योग: कर्मसु कौशलम्



Parul[®]
University

NAAC
ACCREDITED UNIVERSITY

PREFACE

We are delighted to present the first edition of the *Mobile Application Development Practical Book* for BTech 6th semester students at Parul University.

The theory and laboratory courses in Mobile Application Development (MAD) at Parul University, Waghodia, Vadodara, are meticulously designed to bridge theoretical knowledge with practical application. In the theory classes, students gain foundational insights into the subject, while the laboratory sessions offer hands-on experience to implement the concepts learned. The primary goal of the MAD laboratory course is to enable students to *Learn Mobile Application Development through Experimentation*.

This *MAD Practical Book* provides a comprehensive guide to all experiments included in the laboratory syllabus. Each experiment is crafted to address key problems in different areas of MAD and introduces students to practical implementations and real-world applications. The book offers detailed explanations of the fundamental concepts, problem analysis techniques, and guidelines for interpreting and discussing results.

We extend our gratitude to the authors and publishers of the resources referenced during the creation of this practical book. We hope that this *MAD Practical Book* will be a valuable resource for students and fulfill its intended purpose of enhancing their learning experience.

TABLE OF CONTENT



Sr. No	Experiment Title	Page No From To	Date of Start	Date of Completion	Sign	Marks (out of 10)
1.	Compare various operating systems with Android OS.					
2.	Install and configure java development kit (JDK), android studio and android SDK					
3.	Configure android development tools (ADT) plug-in and create android virtual device					
4.	Develop a program to display Hello World on screen.					
5.	Develop a program to implement linear layout and absolute layout.					
6.	Develop a program to implement frame layout, table layout and relative layout.					
	Create an application that takes the name from a text box and shows hello message along					
7.	with the name entered in text box,					
	when the user clicks the OK button					
	Create a screen that has input boxes for User Name, Password,					
	Address, Gender (radio					
8.	buttons for male and female), Age (numeric), Date of Birth (Date Picket), State (Spinner)					
	and a Submit button. On clicking the submit button, print all the data below the Submit					
	Button (use any layout)					
9.	Design an android application to create page using Intent and one Button and pass the Values from one Activity to second Activity					
10	Design an android application Send SMS using Intent.					
11	Create an android application using Fragments.					
12.	Design an android application Using Radio button.					
13.	Design an android application for menu.					

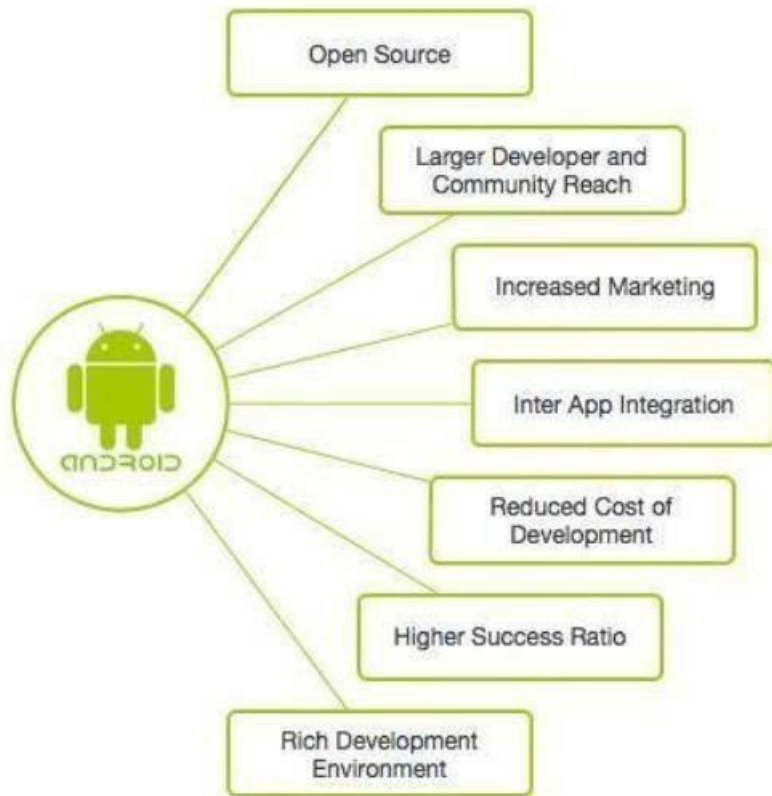
PRACTICAL NO: 1



AIM: Compare various operating systems with Android OS.

- 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.
- The goal of android project is to create a successful real-world product that improves the mobile 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:



•

- After learning what is android, let's see the features of android. The important features of android are given below:

1. It is open-source.
2. Anyone can customize the Android Platform.
3. There are a lot of mobile applications that can be chosen by the consumer.
4. It provides many interesting features like weather details, opening screen, live RSS (Really Simple Syndication) feeds etc.
5. It provides support for messaging services (SMS and MMS), web browser, storage (SQLite), connectivity (GSM, CDMA, Blue Tooth, Wi-Fi etc.), media, handset layout etc.
 - Categories of Android applications:
 - There are many android applications in the market. The top categories are:

i. Entertainment ii. Tools iii. Communication iv. Productivity

v. Personalization vi. Music and Audio vii. Social viii. Media and Video ix. Travel and Local etc.

- Windows Operation System:
 - Microsoft Windows is an operating system that was developed by Microsoft

Cooperation. It is one of the most well-known operating systems in the world. It employs a graphical user interface (GUI).

•

- It enables users to save data, watch videos, run applications, play games, and connect to the Internet.
- The first version of Microsoft Windows is version 1.0, which was released on November 10, 1983. Microsoft Windows is available in various flavors, including Windows XP, Vista, Windows 95, Windows 7, 8, 10, 11, and 12.
- The first business-oriented Windows operating system version was released in 1993, known as Windows NT 3.1.
- Windows launched the next versions, including Windows 3.5, 4/0, and Windows 2000. It developed several versions of Windows XP for home and the commercial environment when it was released in 2001. It was made with standard x86

hardware, including AMD and Intel processors. As a result, it could run on various hardware, including HP, Dell, Sony, and custom-built PCs.

- **Linux OS:**
 - Linux is a powerful and flexible family of operating systems that are free to use and share. It was created by a person named Linus Torvalds in 1991.
 - What's cool is that anyone can see how the system works because its source code is open for everyone to explore and modify. This openness encourages people from all over the world to work together and make Linux better and better. Since its beginning, Linux has grown into a stable and safe system used in many different things, like computers, smartphones, and big supercomputers.
 - It's known for being efficient, meaning it can do a lot of tasks quickly, and it's also cost-effective, which means it doesn't cost a lot to use. As technology keeps moving forward, Linux will keep evolving and staying important in the world of computers.



- **iOS Operating System:**
- iOS is an abbreviation for the iPhone operating system. iOS is the operating system that runs on many of Apple's mobile devices, including the iPhone and iPod Touch.
- It is the world's second most popular mobile OS, trailing only Android.
- It is the basis for three other Apple operating systems: iPadOS, tvOS, and watchOS. It is a part of proprietary software. Some are open source under the Apple Public Source License and other licenses.
- The iOS OS was first released in 2007 for the first-generation iPhone and has since been updated to support more Apple devices, including the iPod Touch and iPad. Apple's App Store had over 2.1 million iOS apps as of March 2018, with 1 million unique to iPads.
- Apple releases a new major version of iOS every year. On September 20, 2021, the current stable version, iOS 15, was released to the public.
- **Comparison between various Operating System:**

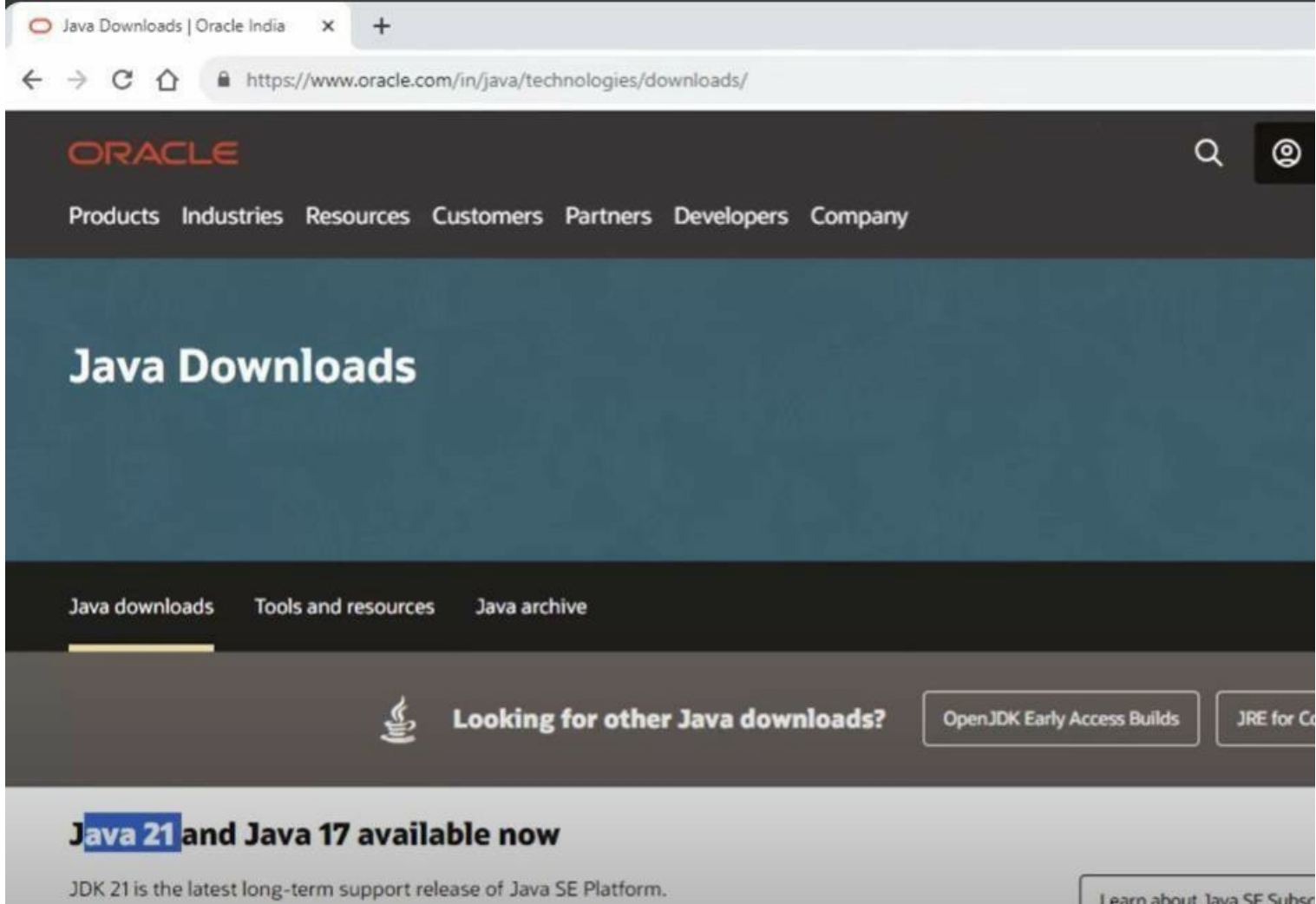
Aspect	Android OS	Windows OS	Linux OS	iOS
Development Company	Google	Microsoft	Various	Apple
Kernel	Linux	Windows NT	Linux Varied (Depends on Desktop Environment)	XNU (Hybrid)
User Interface	Material Design	Modern UI	Varied (Repositories, Software Center, etc.)	Flat Design
App Ecosystem	Google Store	Play	Microsoft Store	Apple Store
Programming Language	Java, Kotlin	C#, C++, .Net	Various (C, C++, Python, etc.)	Objective-C, Swift
Security	Varied (Depends on OEM and updates)	High (with regular updates)	Varied (Depends on Distribution)	High (with regular updates)
Licensing Model	Open Source (AOSP)	Proprietary (Commercial)	Open Source (Most distributions)	Closed Source (Proprietary)
Development Tools	Android Studio, SDK	.NET Framework	Various (GCC, Clang, etc.)	Xcode, Swift Playgrounds

PRACTICAL NO: 2



Aim: Install and configure java development kit (JDK), android studio and android SDK.

- - **JDK Installation:**



1. Download the JDK from the official site of Oracle “Java Downloads | Oracle India” .
2. Once downloaded open the setup and follow the steps.



1. Choose the destination folder where you want to install the jdk. Click on next this will start the installation process.



1. Once done installing it will show the following pop-up.



- - **Android Studio Installation:**
- 1. Open the site "[Download Android Studio & App Tools - Android Developers](#)" and click on the download now option to start the download for android studio.



ANDROID STUDIO

Download

Android Studio editor


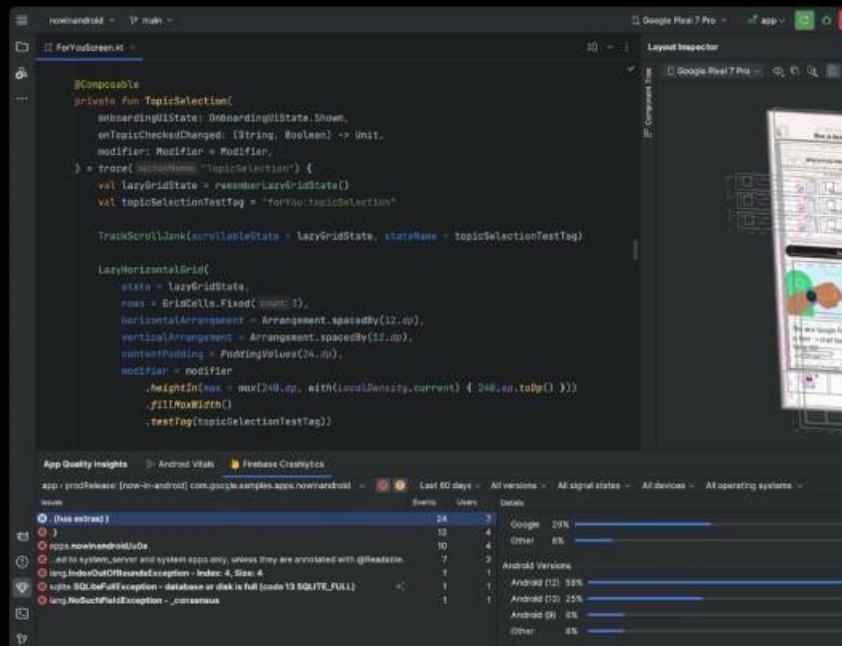

Android Gradle Plugin

SDK tools

Preview

Android Studio

Get the official Integrated Development Environment (IDE) for Android app development.

[Download Android Studio Hedgehog](#) [Read release notes](#) 

1. Once downloaded open the android studio setup and the following window will popup.

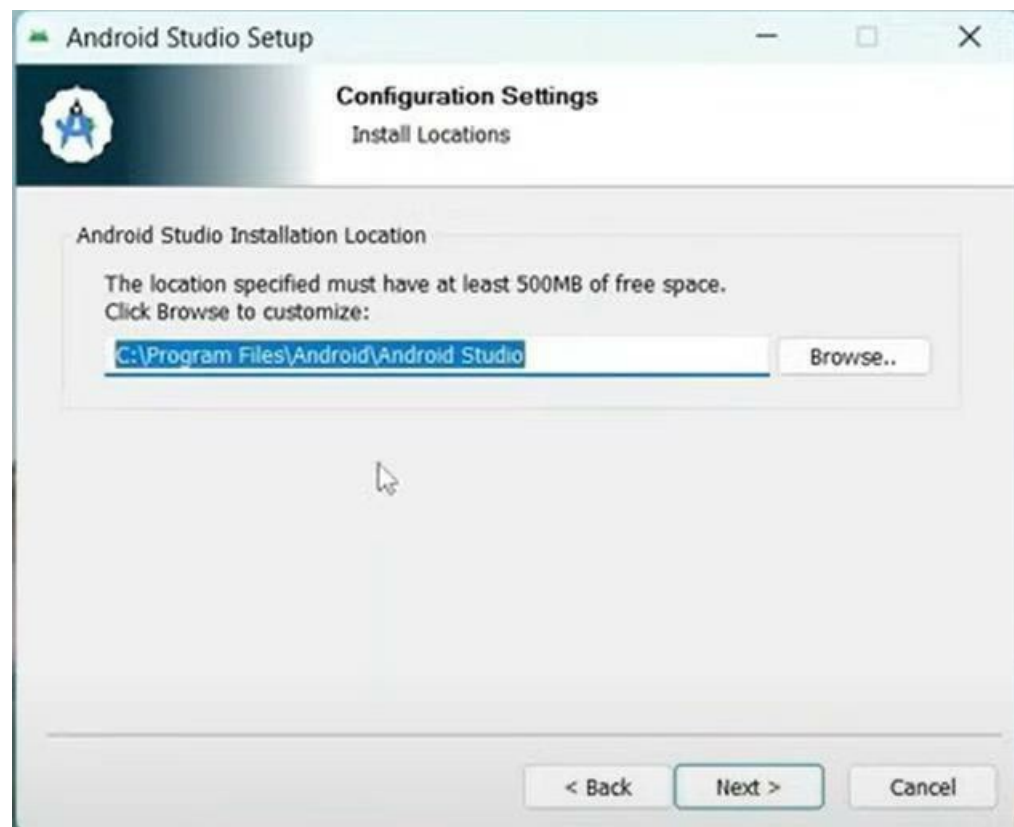




1. Choose the component you want to install in android studio and click on next



1. Select the installation location and click on next to start the installation.





1. Once done installation you are all-set to open android studio and start developing your android apps.



PRACTICAL 3



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

On starting android studio for first time the setup android development tools will open up.

- Follow the steps to configure the ADT and create a virtual device:

1. This is the first screen you will see when you open android studio for the first time.



Welcome

Android Studio

Welcome! This wizard will set up your development environment for Android Studio.
Additionally, the wizard will help port existing Android apps into Android Studio
or create a new Android application project.



Previous

Next

Cancel

1. Select the installation type for android studio.



Install Type

Choose the type of setup you want for Android Studio:

☒ Standard

Android Studio will be installed with the most common settings and options.
Recommended for most users.

☐ Custom

You can customize installation settings and components installed.

Previo...

Next

Cancel



University[®] A⁺⁺

1. Select the components you want to install.



SDK Components Setup

Check the components you want to update/install. Click Next to continue.

- ☒ Android SDK – (411 MB)
- ☐ Android SDK Platform
 - ☒ Android 14.0 ("UpsideDownCake"; API 34) – (159 MB)
- ☒ Performance (Intel® HAXM) – (500 KB)
- ☒ Performance (Android Emulator hypervisor driver) – (165 KB)
- ☒ Android Virtual Device – (1.44 GB)

The collection of Android platform APIs, tools and utilities that enables you to debug, profile, and compile your app. The setup wizard will update your current Android SDK installation (if necessary) or install a new version.

Android SDK Location:

C:\Users\gs202\AppData\Local\Android\Sdk

Total download size:

Available disk space: 72.7 GB

Previo...

Next

Cancel

1. Click on next, agree the licenses and click on next this will start the installation for the components.



License Agreement

Read and agree to the licenses for the components which will be installed

Licenses

android-sdk-license

↓ Android Emulator

↓ Android Emulator hypervisor

↓ Sources for Android 34

↓ Android SDK Platform 34

↓ Android SDK Build-Tools 34

↓ Android SDK Platform-Tools

android-sdk-arm-dbt-license

↓ Google APIs Intel x86_64

intel-android-extra-license

↓ Intel x86 Emulator Accelerator

trademarks, service marks, logos, domain names, or other distinctive brand features.

3.8 You agree that you will not remove, obscure, or alter any proprietary rights notices (including copyright and trademark notices) that may be affixed to or contained within the SDK.

4. Use of the SDK by You

4.1 Google agrees that it obtains no right, title or interest from you (or your licensors) in the License Agreement in or to any software applications that you develop using the SDK, including any intellectual property rights that subsist in those applications.

4.2 You agree to use the SDK and write applications only for purposes that are permitted by (a) the License Agreement and (b) any applicable law, regulation or generally accepted practices or guidelines in the relevant jurisdictions (including any laws regarding the export of data or software to and from the United States or other relevant countries).

4.3 You agree that if you use the SDK to develop applications for general public users, you will protect the privacy and legal rights of those users. If the users provide you with user names, passwords, or other login information or personal information, you must make the use of the SDK subject to a privacy policy.

☒ Decline

Previo...

Next

Cancel

com



University[®] A⁺⁺

1. Click on finish and you are ready.



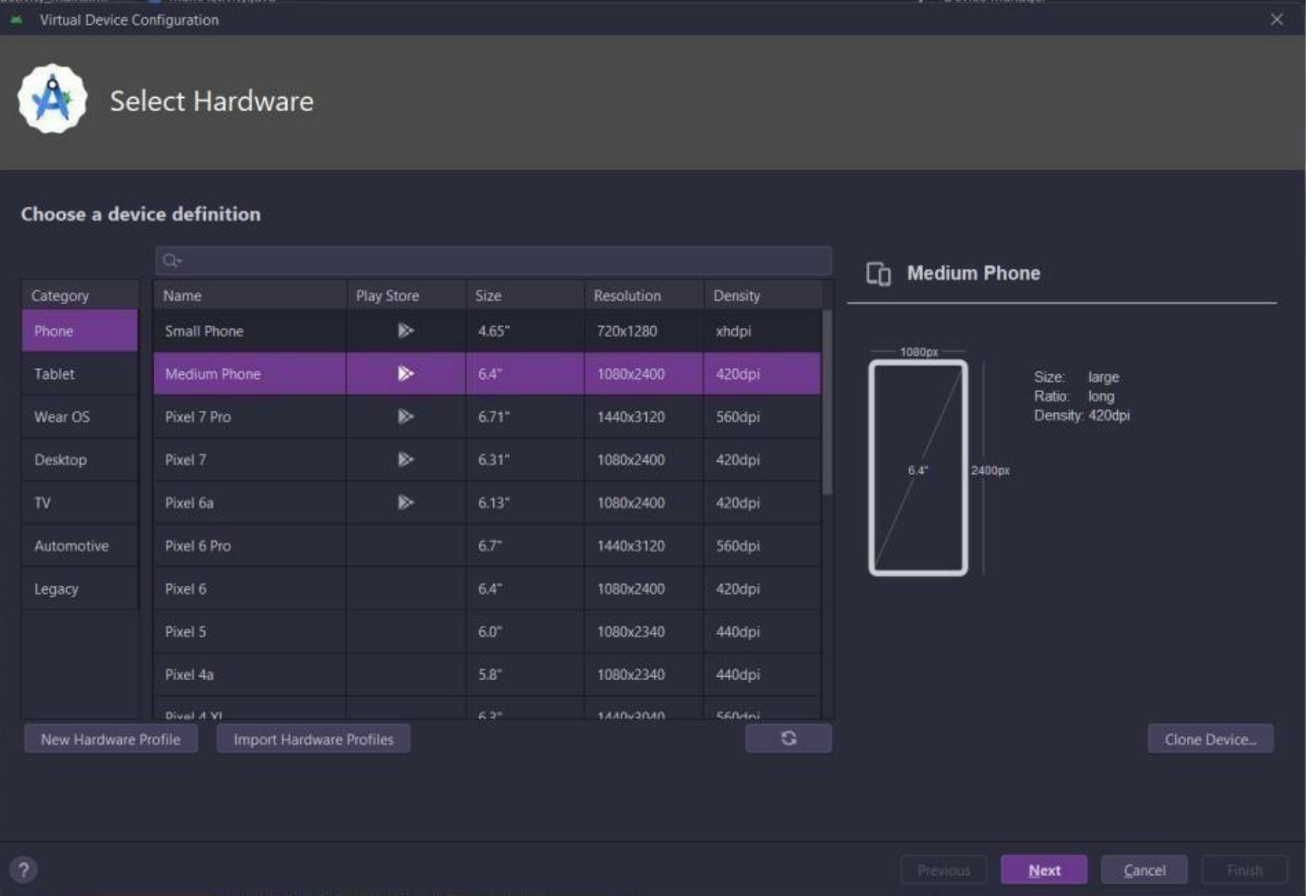
Downloading Components

```
Preparing "Install Android Emulator v.33.1.23".
Downloading https://dl.google.com/android/repository/emulator-windows_x64-11150993
.zip
"Install Android Emulator v.33.1.23" ready.
Installing Android Emulator in C:\Users\gs202\AppData\Local\Android\Sdk\emulator
"Install Android Emulator v.33.1.23" complete.
"Install Android Emulator v.33.1.23" finished.
Parsing C:\Users\gs202\AppData\Local\Android\Sdk\build-tools\34.0.0\package.xml
Parsing C:\Users\gs202\AppData\Local\Android\Sdk\emulator\package.xml
Parsing C:\Users\gs202\AppData\Local\Android\Sdk\platform-tools\package.xml
Parsing C:\Users\gs202\AppData\Local\Android\Sdk\platforms\android-34\package.xml
Parsing C:\Users\gs202\AppData\Local\Android\Sdk\sources\android-34\package.xml
Android SDK is up to date.
```

[Previo...](#)[Next](#)[Cancel](#)

- Steps to setup an AVD:

1. In Android Studio, launch the Android Virtual Device Manager by selecting Tools, then Android, and then AVD Manager.
2. In the Your Virtual Devices screen, click Create Virtual Device.
3. In the Select Hardware screen, select a phone device, such as Pixel, and then click Next.



1. In the System Image screen, click Download for one of the recommended system images. Agree to the terms to complete the download.





System Image

Select a system image

Recommended

x86 Images

Other Images

Release Name	API Level	ABI	Target
VanillaIceCream	VanillaIceCream	x86_64	Android API VanillaIceCream (Google Play)
UpsideDownCakePrivacySan...	UpsideDownCakePr	x86_64	Android API UpsideDownCakePrivacyS
TiramisuPrivacySandbox	TiramisuPrivacySan	x86_64	Android 14.0 (Google Play)
UpsideDownCake	34	x86_64	Android 14.0 (Google Play)
Tiramisu	33	x86_64	Android 13.0 (Google Play)
Sv2	32	x86_64	Android 12L (Google Play)
S	31	x86_64	Android 12.0 (Google Play)
R	30	x86	Android 11.0 (Google Play)
Q	29	x86	Android 10.0 (Google Play)
Pie	28	x86	Android 9.0 (Google Play)
Oreo	27	x86	Android 8.1 (Google Play)
Oreo	26	x86	Android 8.0 (Google Play)



Tiramisu

API Level

33

Type

Google Play

Android

13.0**Google Inc.**

System Image

x86_64

We recommend these Google Play images compatible with Google Play.

[Questions on API level?](#)[See the API level distribution chart](#)[Previous](#)[Next](#)

1. After the download completes, select the system image from the list and click Next.
2. On the next screen, leave all the configuration settings unchanged and click Finish.



Android Virtual Device (AVD)

Verify Configuration

AVD Name

Medium Phone API 33



Medium Phone

6.4 1080x2400 420dpi

Change...



Tiramisu

Android 13.0 x86_64

Change...

Startup orientation



Portrait



Landscape

Emulated
Performance

Graphics:

Automatic

Device Frame



Enable device frame

Show Advanced Settings

AVD Name

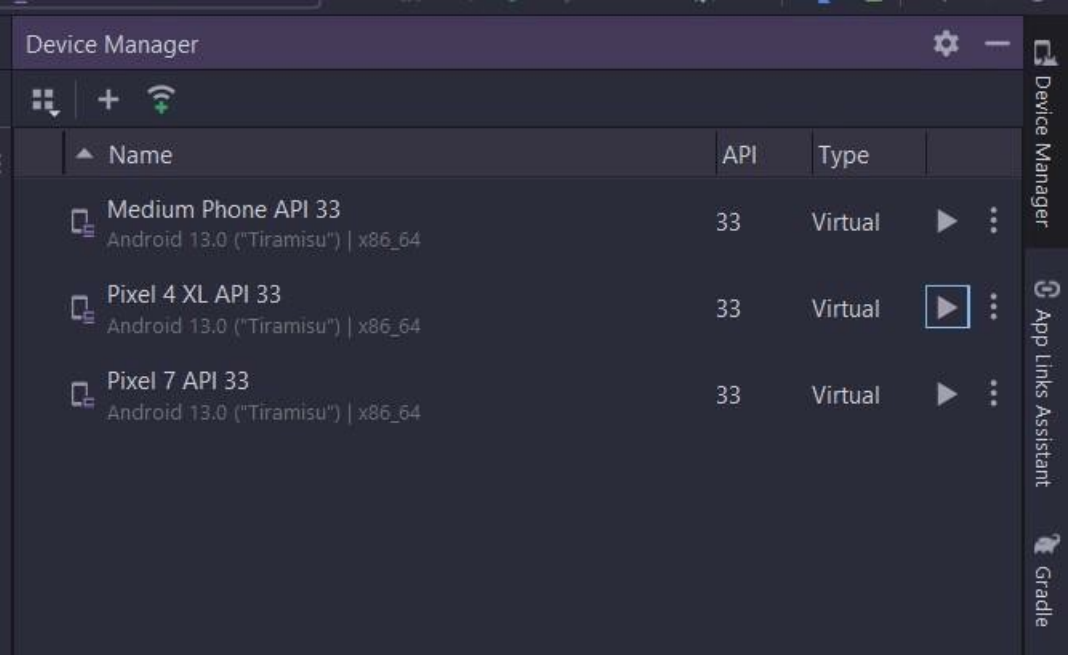
The name of this AVD.



1. In the Your Virtual Devices screen, select the device you just created and click Launch this AVD in the emulator.



University[®] A⁺⁺



PRACTICAL 4



Aim: Develop a program to display Hello World on screen.

Codes:

activity_main.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:layout_width="wrap_content" android:layout_height="wrap_content" android:text="Hello World" android:textSize="30dp"
    android:textStyle="bold" android:layout_marginTop="280dp" app:layout_constraintTop_toTopOf="parent" app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"/>

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

MainActivity.java code:

```
package com.example.pract_4_hello_world;

import androidx.appcompat.app.AppCompatActivity; import android.os.Bundle;

public class MainActivity extends AppCompatActivity {
```



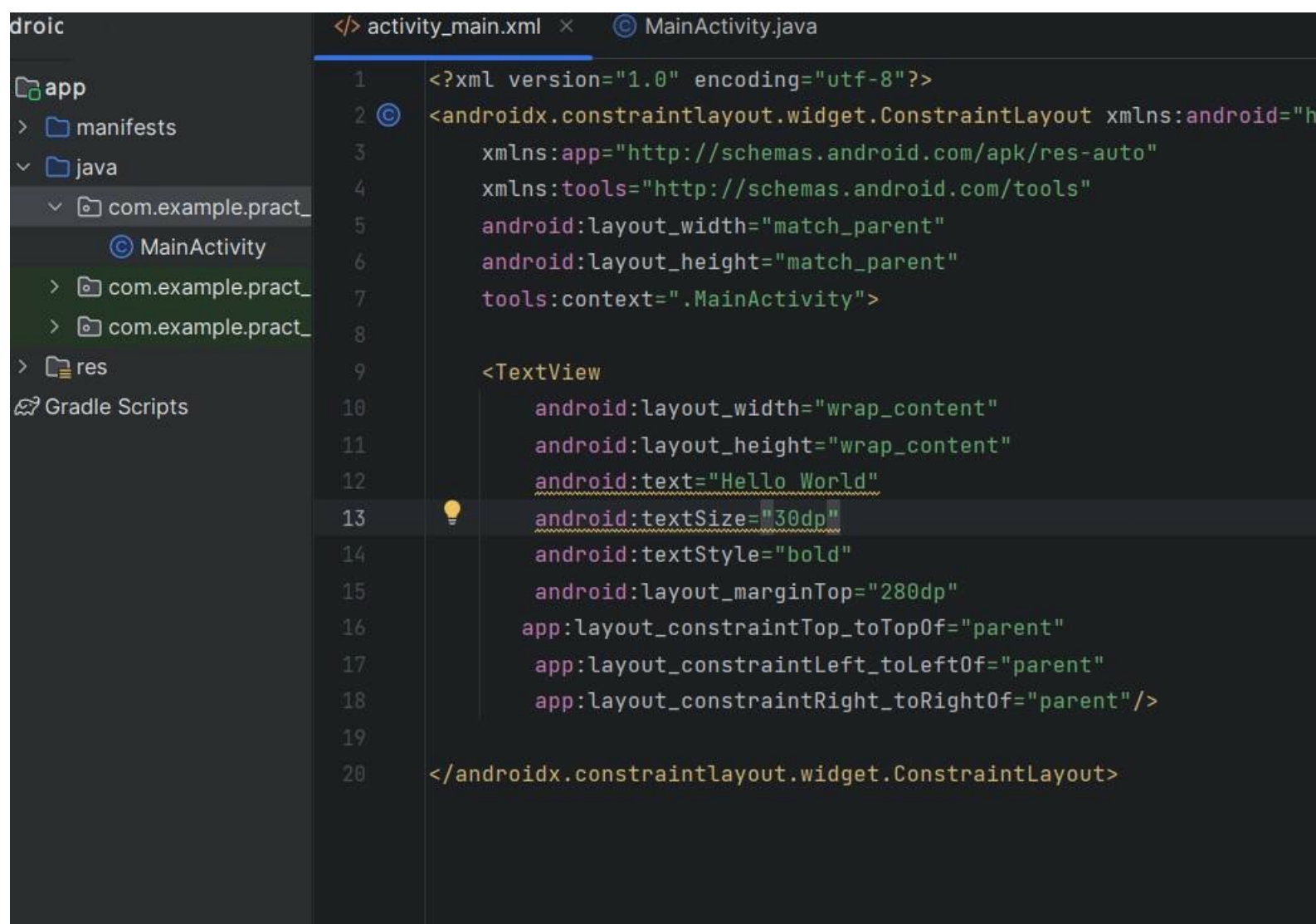
```
@Override

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

}

}
```


Output:



```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="h
    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:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World"
        android:textSize="30dp"
        android:textStyle="bold"
        android:layout_marginTop="280dp"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"/>

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

PRACTICAL 5



Aim: Develop a program to implement linear layout.

Code: activity_main.xml

```
<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"
android:gravity="center">
<TextView
android:id="@+id/textView"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
```

```

android:text="Linear Layout Example"

android:textSize="24sp"

android:layout_marginBottom="16dp"

android:textAlignment="center"

android:textStyle="bold" />

<Button

android:id="@+id/button1"

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:text="Button 1"

android:layout_marginBottom="8dp" />

<Button

android:id="@+id/button2"

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:text="Button 2" />

</LinearLayout>

```

MainActivity.java

```

package com.example.linearlayoutdemo;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

    }

}

```



PRACTICAL 6

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

Code: activity_frame_layout.xml

```

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

<FrameLayout

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

android:layout_width="match_parent"

android:layout_height="match_parent">

```

```

<ImageView
android:id="@+id/imageView"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:src="@drawable/example_image"
android:scaleType="centerCrop" />

<TextView
android:id="@+id/textOverlay"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_gravity="center"
android:text="Frame Layout Example"
android:textSize="24sp"
android:textColor="#FFFFFF"
android:background="#80000000"
android:padding="8dp" />

</FrameLayout>

```

Code: activity_table_layout.xml

```

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

<TableLayout
xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:stretchColumns="1">

<TableRow>

<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Name:"
android:padding="8dp" />

<EditText
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Enter your name" />

</TableRow>

<TableRow>

<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"

```

```

android:text="Age:"

android:padding="8dp" />

<EditText

android:layout_width="match_parent"

android:layout_height="wrap_content"

android:hint="Enter your age" />

</TableRow>

<TableRow>

<Button

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:text="Submit"

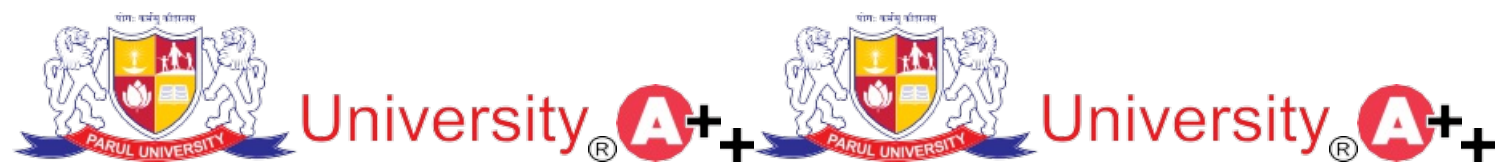
android:layout_span="2"

android:gravity="center" />

</TableRow>

</TableLayout>

```



Code: activity_relative_layout.xml

```

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

<RelativeLayout

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

android:layout_width="match_parent"

android:layout_height="match_parent"

android:padding="16dp">

<TextView

android:id="@+id/labelText"

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:text="Relative Layout Example"

android:textSize="24sp"

android:textStyle="bold"

android:layout_centerHorizontal="true" />

<Button

android:id="@+id/button1"

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:text="Button 1"

```

```

android:layout_below="@id/labelText"
android:layout_marginTop="16dp"
android:layout_alignParentStart="true" />
<Button
android:id="@+id/button2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Button 2"
android:layout_toEndOf="@id/button1"
android:layout_below="@id/labelText"
android:layout_marginStart="8dp"
android:layout_marginTop="16dp" />
</RelativeLayout

```

PRACTICAL 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

Codes:

activity_practical7.xml Code:

```

<RelativeLayout 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:padding="16dp">

<EditText android:id="@+id/editTextName" android:layout_width="match_parent" android:layout_height="wrap_content" android:hint="Enter your name"
android:inputType="text" android:layout_marginBottom="16dp"/>

<Button
android:id="@+id/buttonOK" android:layout_width="wrap_content" android:layout_height="wrap_content" android:layout_below="@id/editTextName"
android:layout_centerHorizontal="true" android:text="OK"/>

<TextView android:id="@+id/textViewMessage" android:layout_width="wrap_content" android:layout_height="wrap_content"
android:layout_below="@id/buttonOK" android:layout_centerHorizontal="true" android:layout_marginTop="16dp" android:text="" android:textSize="18sp"/>

</RelativeLayout>

```

Practical7Activity.java Code:

```

package com.priyamtripz.mad;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;

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

```



```
public class Practical7Activity extends AppCompatActivity { EditText editTextName;

Button buttonOK;

TextView textViewMessage; @Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity_practical7); editTextName =
findViewById(R.id.editTextName); buttonOK = findViewById(R.id.buttonOK); textViewMessage = findViewById(R.id.textViewMessage);

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

public void onClick(View v) {

String name = editTextName.getText().toString(); String message = "Hello, " + name + "!"; textViewMessage.setText(message);

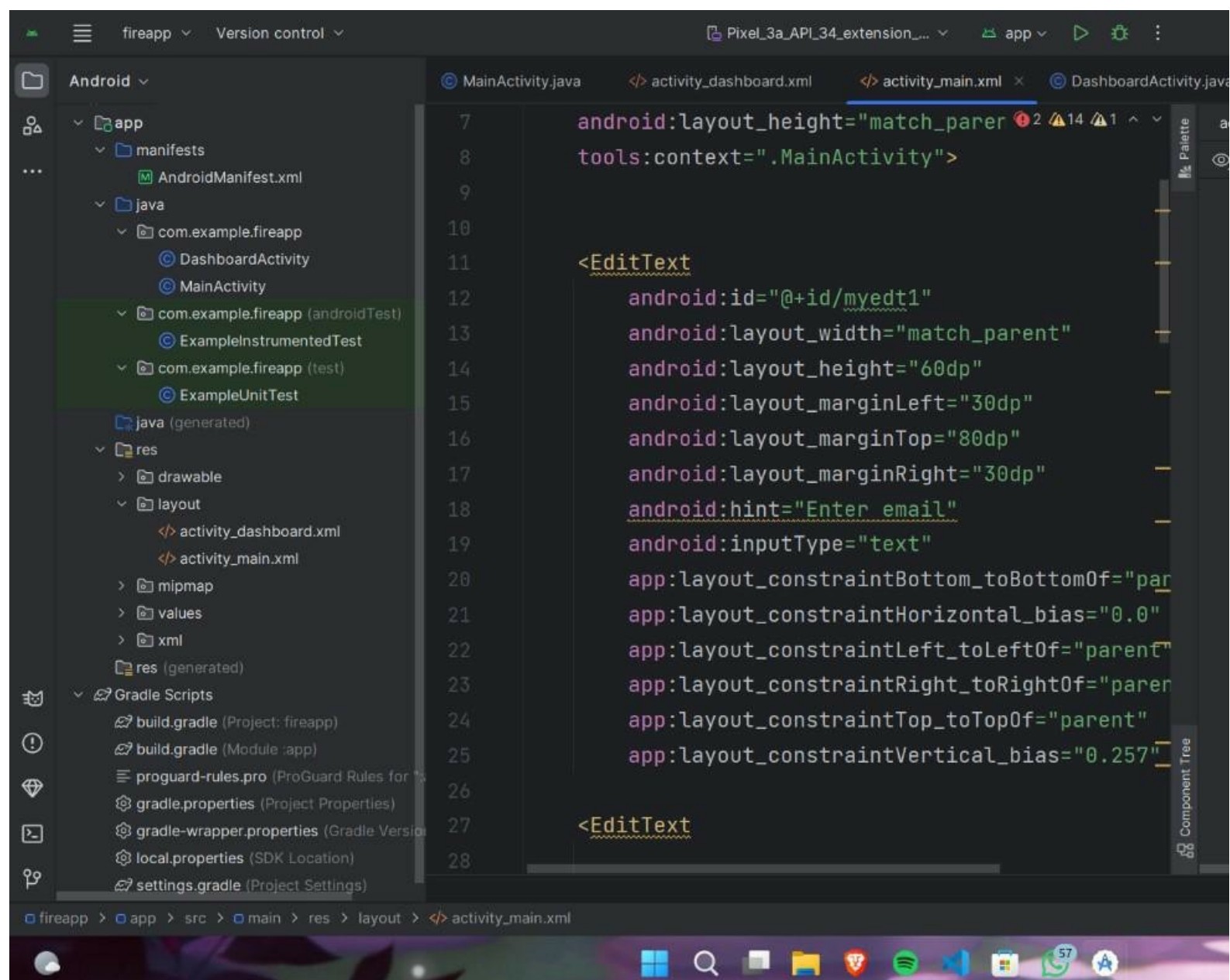
}

});

}

}
```

Output:



PRACTICAL 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)

Codes:

activity_practical8.xml code:

```
<ScrollView 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:padding="16dp">

<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="User Name"
android:inputType="text"/>

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

<EditText android:id="@+id/editTextAddress" android:layout_width="match_parent" android:layout_height="wrap_content" android:hint="Address"
android:inputType="text"/>

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

<RadioButton android:id="@+id/radioButtonMale"

android:layout_width="wrap_content" android:layout_height="wrap_content" android:text="Male"/>

<RadioButton android:id="@+id/radioButtonFemale" 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="Age"
android:inputType="number"/>

<DatePicker android:id="@+id/datePickerDOB" android:layout_width="match_parent" android:layout_height="wrap_content"/>

<Button

android:id="@+id/buttonSubmit" android:layout_width="wrap_content" android:layout_height="wrap_content" android:text="Submit"/>

<TextView android:id="@+id/textViewData" android:layout_width="match_parent" android:layout_height="wrap_content" android:layout_marginTop="16dp"
android:text="" android:textSize="18sp"/>

</LinearLayout>

</ScrollView>
```

Practical8Activity.java code:

```
package com.priyamtripz.mad;

import androidx.appcompat.app.AppCompatActivity; import android.os.Bundle;

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

import android.widget.DatePicker; import android.widget.EditText; import android.widget.RadioButton; import android.widget.RadioGroup; import android.widget.TextView;



```
public class Practical8Activity extends AppCompatActivity {

    EditText editTextUserName, editTextPassword, editTextAddress, editTextAge; RadioGroup radioGroupGender;

    RadioButton radioButtonMale, radioButtonFemale; DatePicker datePickerDOB;

    Button buttonSubmit; TextView textViewData; @Override

    protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity_practical8);

    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);
    radioButtonMale = findViewById(R.id.radioButtonMale); radioButtonFemale = findViewById(R.id.radioButtonFemale); datePickerDOB =
    findViewById(R.id.datePickerDOB); buttonSubmit = findViewById(R.id.buttonSubmit); textViewData = findViewById(R.id.textViewData);

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

    public void onClick(View v) {

    String userName = editTextUserName.getText().toString(); String password = editTextPassword.getText().toString(); String address =
    editTextAddress.getText().toString(); String age = editTextAge.getText().toString();

    String gender = radioButtonMale.isChecked() ? "Male" : "Female"; String dob = getDateFromDatePicker(datePickerDOB);

    String data = "User Name: " + userName + "\n" + "Password: " + password + "\n" +

    "Address: " + address + "\n" + "Gender: " + gender + "\n" + "Age: " + age + "\n" +

    "Date of Birth: " + dob;

    textViewData.setText(data);

    }

    });

    }

    private String getDateFromDatePicker(DatePicker datePicker) { int day = datePicker.getDayOfMonth();

    int month = datePicker.getMonth() + 1; int year = datePicker.getYear();

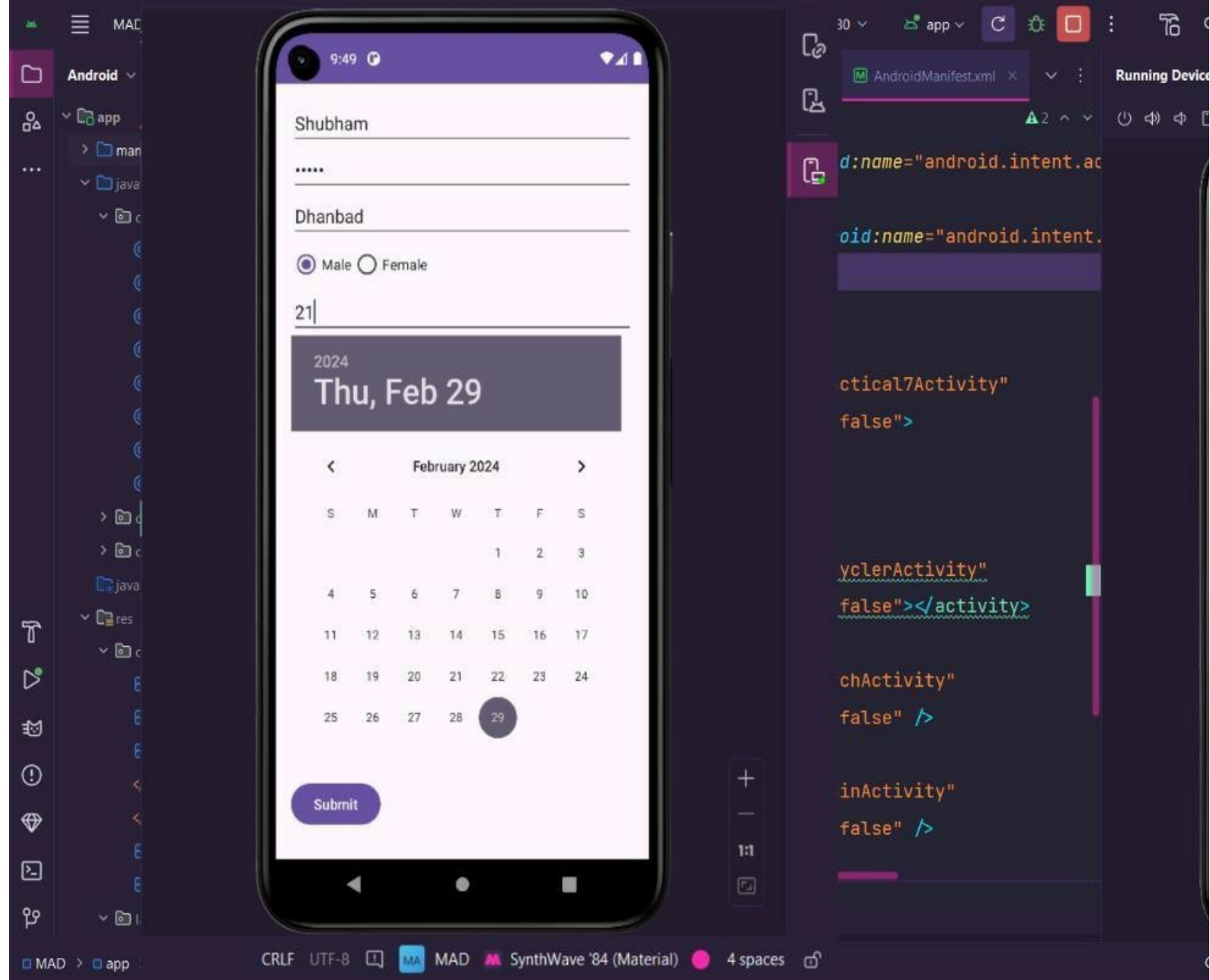
    return day + "/" + month + "/" + year;

    }

    }
```



Output:



PRACTICAL 9



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

Codes:

activity_main.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">

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

    android:layout_width="match_parent" android:layout_height="wrap_content" app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintLeft_toLeftOf="parent" app:layout_constraintTop_toTopOf="parent" android:hint="Enter Data" android:textSize="20sp"
    android:textColor="@color/black" android:layout_marginLeft="30dp" android:layout_marginRight="30dp" android:layout_marginTop="100dp"/>

    <Button
```

```

android:id="@+id/databtn" android:layout_width="wrap_content" android:layout_height="wrap_content" app:layout_constraintRight_toRightOf="parent"
app:layout_constraintLeft_toLeftOf="parent" app:layout_constraintBottom_toBottomOf="parent" android:text="SUBMIT"

android:textSize="25sp" android:layout_marginLeft="30dp" android:layout_marginRight="30dp" android:layout_marginBottom="200dp"/>

</androidx.constraintlayout.widget.ConstraintLayout>

```

MainActivity.java Code:

```

package com.priyamtripz.mad;

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;

public class MainActivity extends AppCompatActivity { EditText editText;

Button submitbtn; @Override

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

editText = findViewById(R.id.etuser); submitbtn = findViewById(R.id.databtn);

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

public void onClick(View view) {

String name = editText.getText().toString();

Intent intent = new Intent(MainActivity.this, FetchActivity.class); intent.putExtra("data", name );

startActivity(intent);

}

});

}

```

activity_fetch.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:orientation="vertical" android:layout_height="match_parent" android:gravity="center"

android:padding="16dp">

<TextView android:id="@+id/textViewDisplay" android:layout_width="wrap_content" android:layout_height="wrap_content" android:textSize="30sp"

android:textColor="@android:color/black"/>

</LinearLayout>

```



FetchActivity.java code:

```

package com.priyamtripz.mad;

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.TextView;

```



```
public class FetchActivity extends AppCompatActivity { TextView textViewDisplay;
```

@Override

```
protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity_fetch); textViewDisplay = findViewById(R.id.textViewDisplay); Intent intent = getIntent();
```

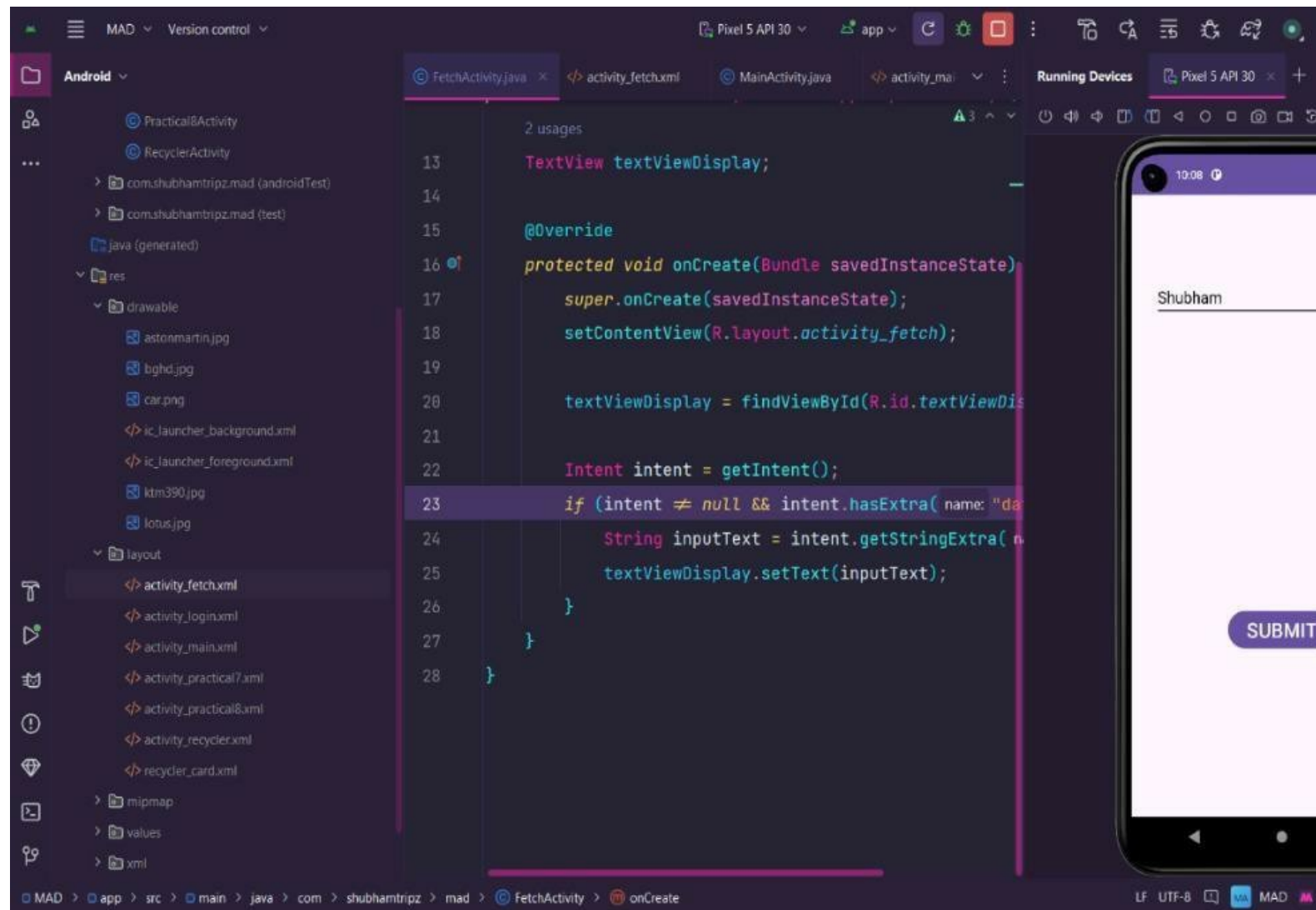
```
if (intent != null && intent.hasExtra("data")) { String inputText = intent.getStringExtra("data"); textViewDisplay.setText(inputText);
```

```
}
```

```
}
```

```
}
```

Output:



PRACTICAL 10

Aim : Design an android application Send SMS using Intent.

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">
```

```
<EditText
```

```

android:id="@+id/phoneNumber"

android:layout_width="match_parent"

android:layout_height="wrap_content"

android:hint="Enter phone number"

android:inputType="phone" />

<EditText

android:id="@+id/messageText"

android:layout_width="match_parent"

android:layout_height="wrap_content"

android:hint="Enter message"

android:inputType="textMultiLine"

android:layout_marginTop="8dp" />

<Button

android:id="@+id/sendButton"

android:layout_width="match_parent"

android:layout_height="wrap_content"

android:text="Send SMS"

android:layout_marginTop="16dp" />

</LinearLayout>

```

MainActivity.java:

```

package com.example.sendsms;

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 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);

        EditText phoneNumber = findViewById(R.id.phoneNumber);

        EditText messageText = findViewById(R.id.messageText);

        Button sendButton = findViewById(R.id.sendButton);

        sendButton.setOnClickListener(new View.OnClickListener() {

            @Override

```

```

public void onClick(View v) {
String phone = phoneNumber.getText().toString();
String message = messageText.getText().toString();

if (phone.isEmpty() || message.isEmpty()) {
Toast.makeText(MainActivity.this, "Please fill in both fields", Toast.LENGTH_SHORT).show();

} else {

Intent intent = new Intent(Intent.ACTION_VIEW);

intent.setData(Uri.parse("sms:" + phone));

intent.putExtra("sms_body", message);

try {

startActivity(intent);

} catch (Exception e) {

Toast.makeText(MainActivity.this, "SMS sending failed", Toast.LENGTH_SHORT).show();

}

}

});

}

}

```

PRACTICAL 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();

    }

}

```

PRACTICAL 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);

        RadioGroup radioGroup = findViewById(R.id.radioGroup);

        Button submitButton = findViewById(R.id.submitButton);

        TextView resultText = 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 {  
  
RadioButton selectedRadioButton = findViewById(selectedId);  
  
String selectedText = selectedRadioButton.getText().toString();  
  
resultText.setText("You selected: " + selectedText);  
  
}  
  
}  
  
});  
  
}  
  
}
```

PRACTICAL 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 TextView displayText;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        displayText = findViewById(R.id.displayText);

    }

    @Override

    public boolean onCreateOptionsMenu(Menu menu) {

        // Inflate the menu

        getMenuInflater().inflate(R.menu.main_menu, menu);

        return true;

    }

    @Override

```

```
public boolean onOptionsItemSelected(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);  
    }  
}  
}
```