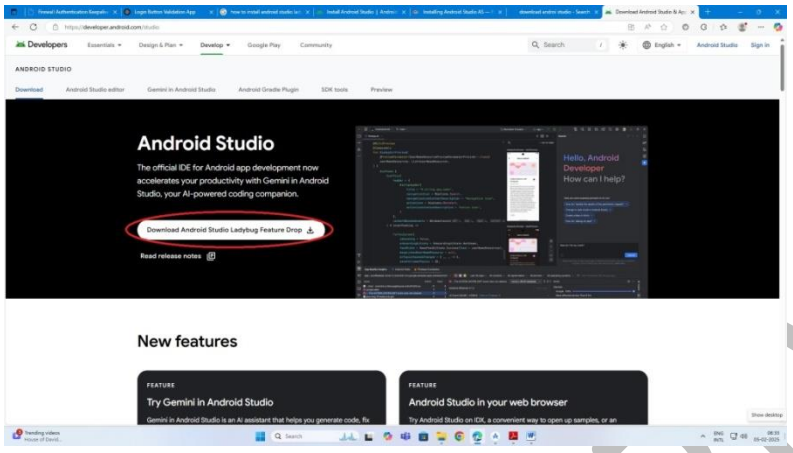
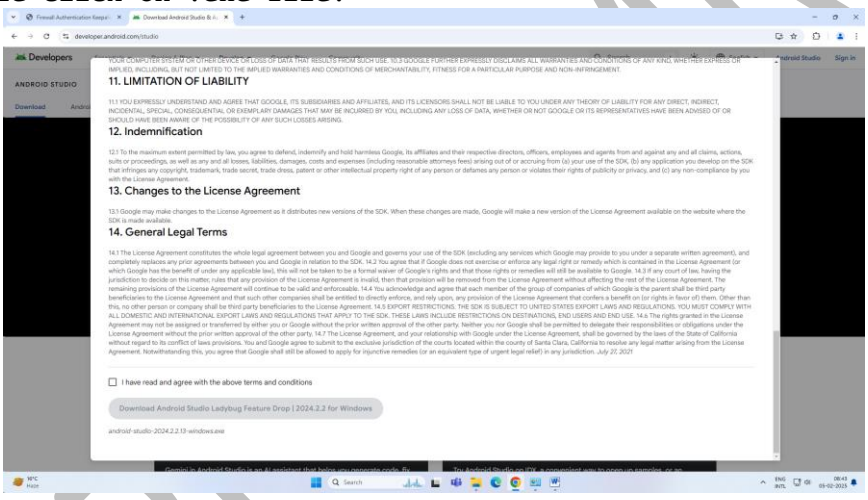
Practical 1: Installation and setup of java development kit (JDK), setup android SDK.

Step 1: visit the https://developer.android.com/studio to download android studio.

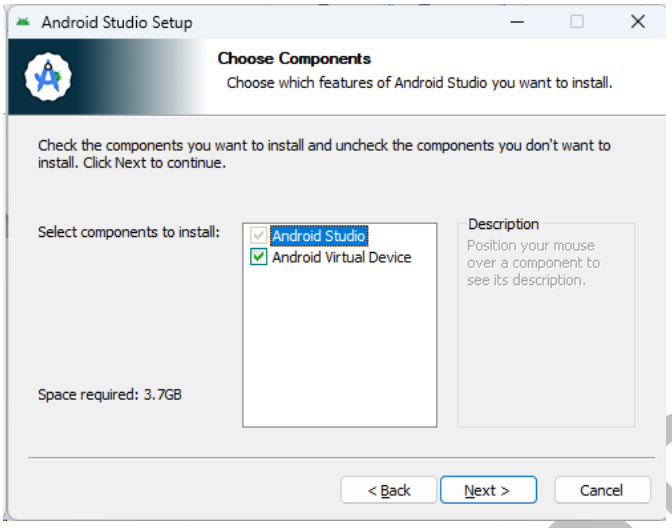


Step 2: Double click on .exe file.

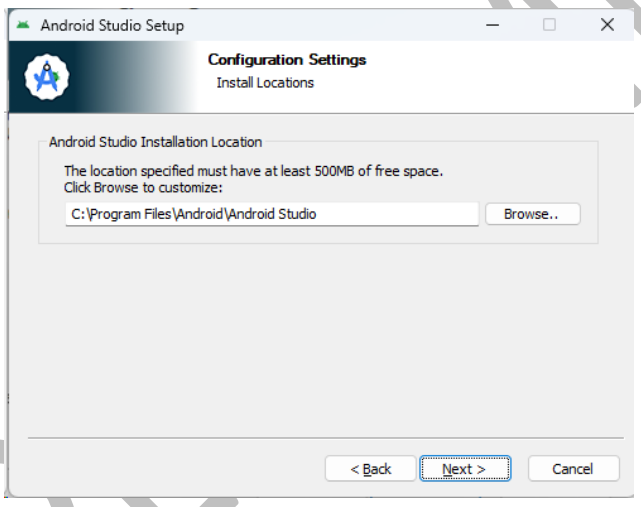


Step 3: Tick the checkbox shown in above window. And then click on download button.

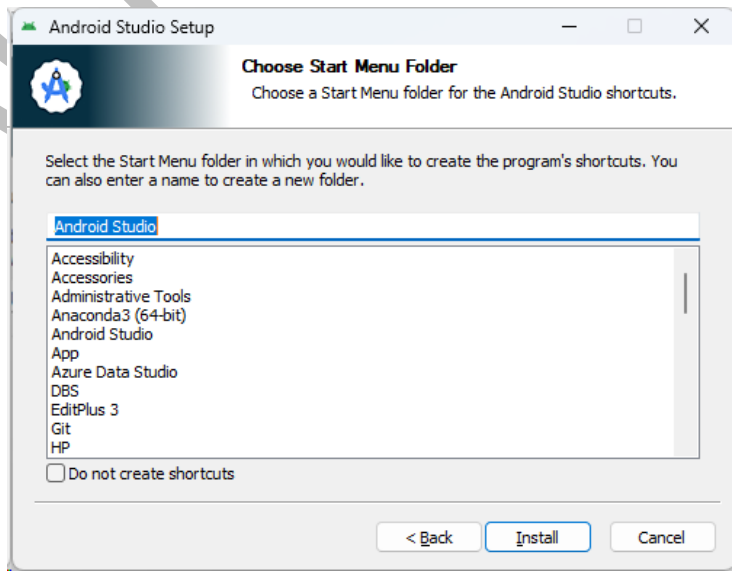
Step 4: Click on Next Button



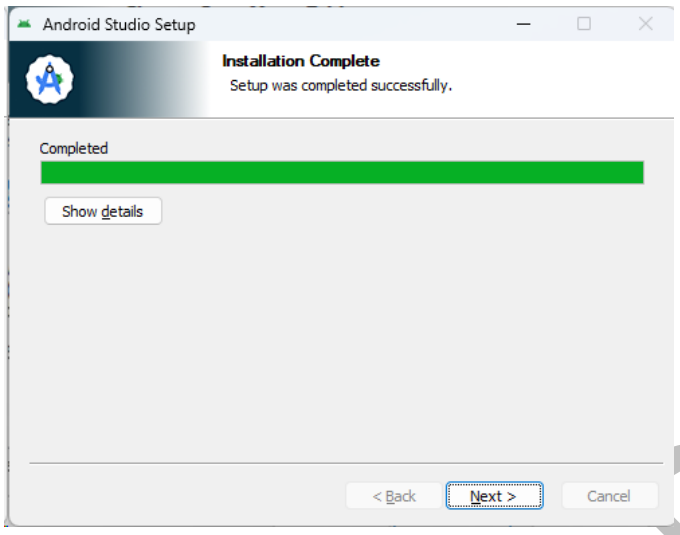
Step 5: Click on Next Button



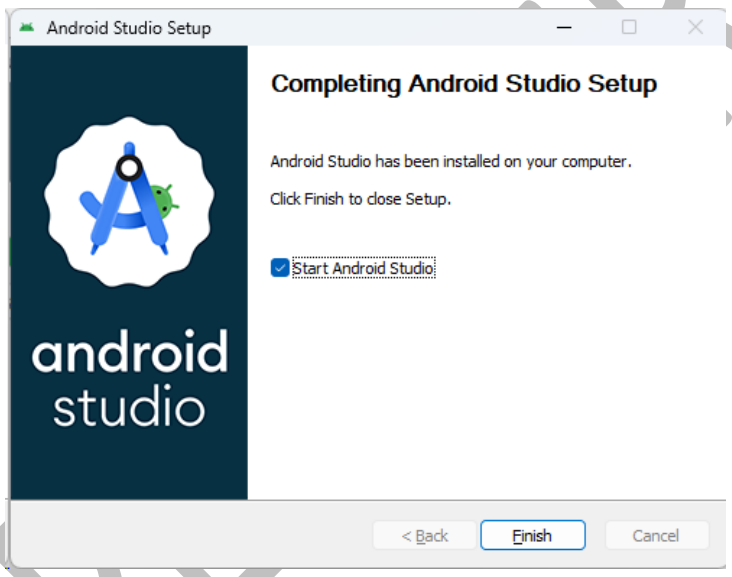
Step 6: Click on Next Button



Step 7: Click on Install Button

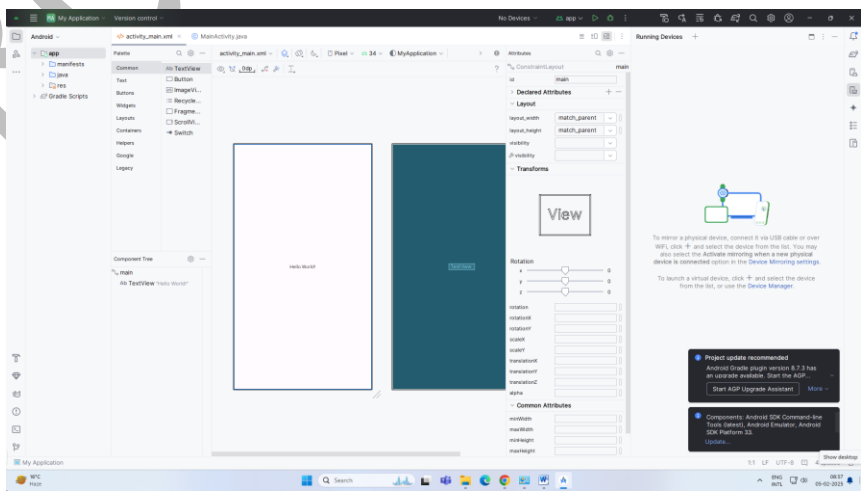


Step 8: Click on Install Button

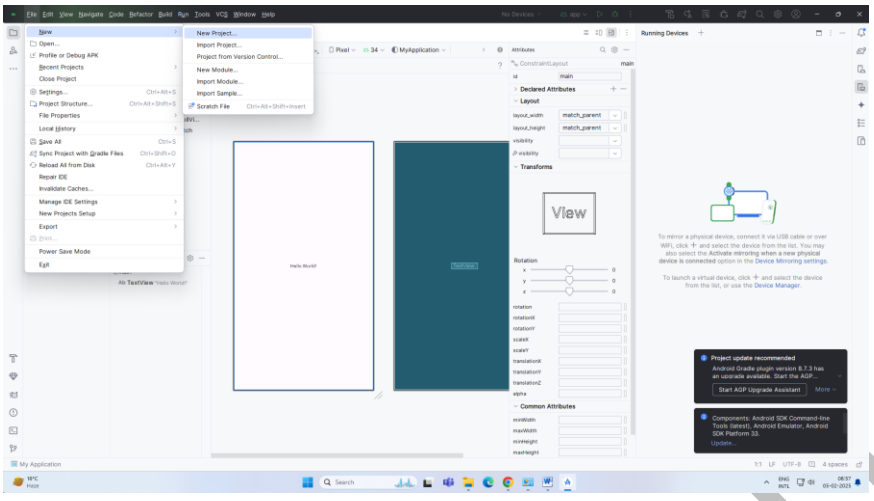


Step 9: Click on Finish Button

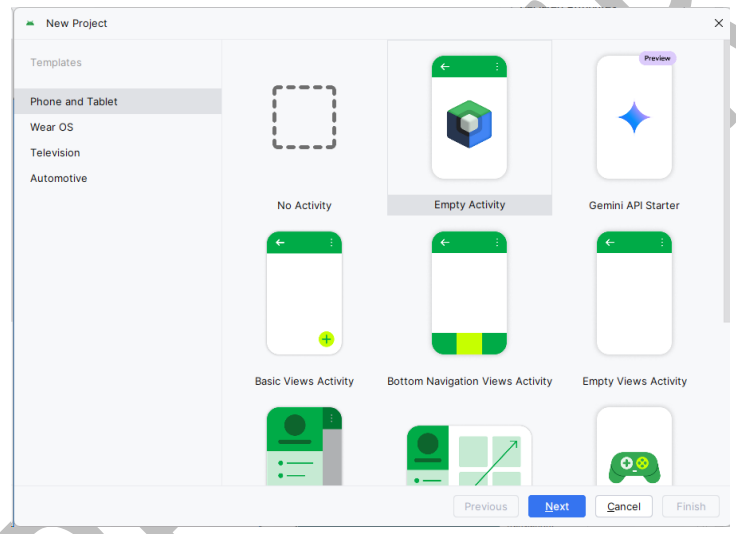
Your Android Studio will be launched.



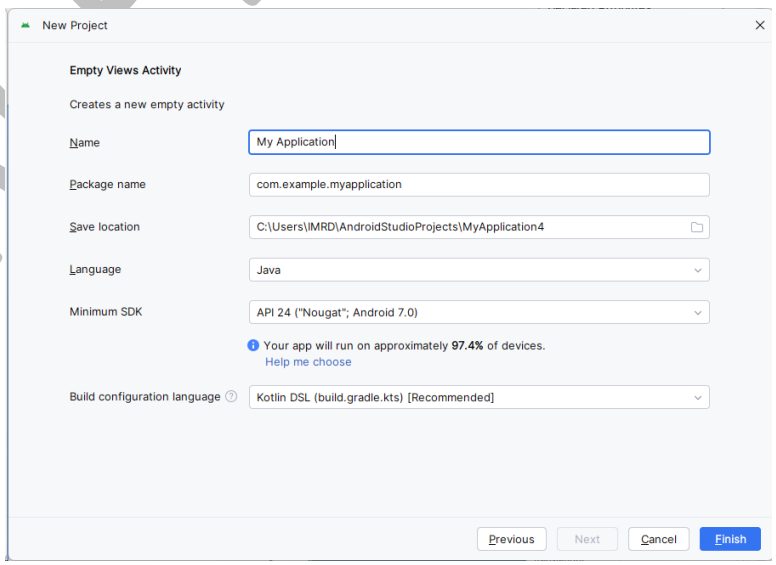
Click on Main Menu (Four Lines left corner)



Click on New-> new Project



Select Empty Views Activity and click on Next Button



Enter the application name and make sure language is selected as Java.

Click on Finish Button- Your project will be launched.

Practical 2: Create "Hello World" application. That will display "Hello World" in the middle of the screen using TextView Widget in the red color.

activity main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    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:textColor="#FF0000"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

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

MainActivity.java

```
package com.example.hellored;
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;
        });
    }
}
```

Practical 3 Create Registration Page to demonstrate basic widgets:(Sign up form)

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="32dp">

    <!-- Username Field -->
    <EditText
        android:id="@+id/etName"
        android:layout_width="match_parent"
        android:layout_height="56dp"
        android:hint="Username"
        android:inputType="text"
        android:padding="16dp"/> <!-- Added padding -->

    <!-- Email Field -->
    <EditText
        android:id="@+id/etEmail"
        android:layout_width="match_parent"
        android:layout_height="56dp"
        android:hint="Email"
        android:inputType="textEmailAddress"
        android:padding="16dp"/> <!-- Added padding -->

    <!-- Password Field -->
    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="56dp"
        android:hint="Password"
        android:inputType="textPassword"
        android:padding="16dp"/> <!-- Added padding -->

    <!-- Confirm Password Field -->
    <EditText
        android:id="@+id/etConfirmPassword"
        android:layout_width="match_parent"
        android:layout_height="56dp"
        android:hint="Confirm Password"
        android:inputType="textPassword"
        android:padding="16dp"/> <!-- Added padding -->

    <!-- Sign Up Button -->
    <Button
        android:id="@+id/btnSignUp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Sign Up" />

</LinearLayout>
```

MainActivity.java

```
package com.example.myapplication;
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 {

    private EditText etName, etEmail, etPassword, etConfirmPassword;
    private Button btnSignUp;

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

        // Initialize views
        etName = findViewById(R.id.etName);
        etEmail = findViewById(R.id.etEmail);
        etPassword = findViewById(R.id.etPassword);
        etConfirmPassword = findViewById(R.id.etConfirmPassword);
        btnSignUp=findViewById(R.id.btnSignUp);


        // Set up the signup button click listener
        btnSignUp.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // Capture user input
                String name = etName.getText().toString().trim();
                String email = etEmail.getText().toString().trim();
                String password = etPassword.getText().toString().trim();
                String confirmPassword =
etConfirmPassword.getText().toString().trim();

              // Simple validation for empty fields
     if (name.isEmpty() || email.isEmpty() || password.isEmpty() ||
confirmPassword.isEmpty()) {
                    Toast.makeText(MainActivity.this, "Please fill in all
fields", Toast.LENGTH_SHORT).show();
                } else if (!password.equals(confirmPassword)) {
                    // Check if password and confirm password match
                    Toast.makeText(MainActivity.this, "Passwords do not
match", Toast.LENGTH_SHORT).show();
                } else {
                    // Simulate a signup action (you can replace this with
actual logic)
                    Toast.makeText(MainActivity.this, "Sign up successful",
Toast.LENGTH_SHORT).show();

                    // Clear the fields after successful signup
                    etName.setText("");
                    etEmail.setText("");
                    etPassword.setText("");
                    etConfirmPassword.setText("");
                }
            }
        });
    }
}
```

Practical 4: Create sample application with login module.(Check username and password) On successful login, Change TextView "Login Successful". And on failing login, alert user using Toast "Login fail".

activity main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    tools:context=".MainActivity">

    <!-- Username EditText -->
    <EditText
        android:id="@+id/etUserName"
        android:layout_width="match_parent"
        android:layout_height="56dp"
        android:hint="Enter Username"
        android:layout_marginTop="50dp"
        android:padding="10dp" />

    <!-- Password EditText -->
    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="56dp"
        android:hint="Enter Password"
        android:layout_below="@id/etUserName"
        android:layout_marginTop="20dp"
        android:padding="10dp"
        android:inputType="textPassword" />

    <!-- Login Button -->
    <Button
        android:id="@+id/btnLogin"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Login"
        android:layout_below="@id/etPassword"
        android:layout_marginTop="30dp"
        android:layout_centerHorizontal="true" />

    <!-- TextView for displaying login status -->
    <TextView
        android:id="@+id/tvStatus"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Login Status"
        android:textSize="20sp"
        android:layout_below="@id/btnLogin"
        android:layout_marginTop="20dp"
        android:layout_centerHorizontal="true"/>

</RelativeLayout>
```

MainActivity.java

```
package com.example.logintest;
import android.annotation.SuppressLint;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    private EditText etUserName, etPassword;
    private Button btnLogin;
    private TextView tvStatus;
    // Sample hardcoded credentials
    private static final String USERNAME = "admin";
    private static final String PASSWORD = "password123";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        // Initialize views
        etUserName = findViewById(R.id.etUserName);
        etPassword = findViewById(R.id.etPassword);
        btnLogin = findViewById(R.id.btnLogin);
        tvStatus = findViewById(R.id.tvStatus);
        // Set click listener for the login button
        btnLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // Get the username and password from EditText fields
                String username = etUserName.getText().toString();
                String password = etPassword.getText().toString();

                // Check if the credentials are correct
                if (username.equals(USERNAME) && password.equals(PASSWORD)) {
                    // If successful, change TextView and display success
message
                    tvStatus.setText("Login Successful");
                } else {
                    // If login fails, show Toast message
                    Toast.makeText(MainActivity.this, "Login Fail",
Toast.LENGTH_SHORT).show();
                    tvStatus.setText("Login Status");
                }
            }
        });
    }
}
```

Practical 5 : Create an application for demonstration of Scroll view in android.

Vertical

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">

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:padding="16dp">

          <TextView
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="This is the first text view."
            android:textSize="18sp"
            android:layout_marginBottom="20dp"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="This is the second text view."
            android:textSize="18sp"
            android:layout_marginBottom="20dp"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="This is the third text view."
            android:textSize="18sp"
            android:layout_marginBottom="20dp"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="This is the fourth text view."
            android:textSize="18sp"
            android:layout_marginBottom="20dp"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="This is the fifth text view."
            android:textSize="18sp"
            android:layout_marginBottom="20dp"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="This is the sixth text view."
            android:textSize="18sp"
            android:layout_marginBottom="20dp"/>
```

```
        <TextView
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="This is the seventh text view."
            android:textSize="18sp"
            android:layout_marginBottom="20dp"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="This is the eighth text view."
            android:textSize="18sp"
            android:layout_marginBottom="20dp"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="This is the ninth text view."
            android:textSize="18sp"
            android:layout_marginBottom="20dp"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="This is the tenth text view."
            android:textSize="18sp"
            android:layout_marginBottom="20dp"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="This is the Elevanth text view."
            android:textSize="18sp"
            android:layout_marginBottom="20dp"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="This is the Twelve text view."
            android:textSize="18sp"
            android:layout_marginBottom="20dp"/>

        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Click Me" />

        </LinearLayout>
    </ScrollView>

</LinearLayout>
```

Horizontal Scroll View

```
<?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">

    <HorizontalScrollView
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:padding="16dp">

            <Button
                android:layout_width="200dp"
                android:layout_height="wrap_content"
                android:text="Button 1" />

            <Button
                android:layout_width="200dp"
                android:layout_height="wrap_content"
                android:text="Button 2" />

            <Button
                android:layout_width="200dp"
                android:layout_height="wrap_content"
                android:text="Button 3" />

            <Button
                android:layout_width="200dp"
                android:layout_height="wrap_content"
                android:text="Button 4" />

        </LinearLayout>
    </HorizontalScrollView>

</LinearLayout>
```

Practical 6: Create login application where you will have to validate username and passwords till the username and password is not validated, login button should remain disabled.

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="20dp">

    <EditText
        android:id="@+id/etUsername"
        android:layout_width="match_parent"
        android:layout_height="56dp"
        android:hint="Username" />

    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="56dp"
        android:hint="Password"
        android:inputType="textPassword"
        android:layout_marginTop="10dp" />

    <Button
        android:id="@+id/btnLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Login"
        android:enabled="false"
        android:layout_marginTop="20dp" />
</LinearLayout>
```

MainActivity.Java

```
package com.example.validatelogin;

import android.os.Bundle;
import android.text.Editable;
import android.text.TextWatcher;
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 {

    EditText etUsername, etPassword;
    Button btnLogin;

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

        etUsername = findViewById(R.id.etUsername);
        etPassword = findViewById(R.id.etPassword);
        btnLogin = findViewById(R.id.btnLogin);


        // Enable button only if both fields are not empty
        TextWatcher textWatcher = new TextWatcher() {
            @Override
            public void onTextChanged(CharSequence s, int start, int before,
int count) {
                                // Enable login button only if both fields are
filled
                String usernameInput = etUsername.getText().toString().trim();
                String passwordInput = etPassword.getText().toString().trim();
                btnLogin.setEnabled(!usernameInput.isEmpty() &&
!passwordInput.isEmpty());
            }
            @Override public void beforeTextChanged(CharSequence s, int start,
int count, int after) {}
            @Override public void afterTextChanged(Editable s) {}
        };

        etUsername.addTextChangedListener(textWatcher);
        etPassword.addTextChangedListener(textWatcher);

        btnLogin.setOnClickListener(v -> {
            if (etUsername.getText().toString().equals("admin") &&
etPassword.getText().toString().equals("1234")) {
                Toast.makeText(MainActivity.this, "Login Successful",
Toast.LENGTH_SHORT).show();
            } else {
                Toast.makeText(MainActivity.this, "Invalid Credentials",
Toast.LENGTH_SHORT).show();
                btnLogin.setEnabled(false);
            }
        });
    }
}
```

Practical 7: Create an application for calculator

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="20dp">
    <EditText
        android:id="@+id/etNumber1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter First Number"
        android:inputType="numberDecimal" />

    <EditText
        android:id="@+id/etNumber2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Second Number"
        android:inputType="numberDecimal"
        android:layout_marginTop="10dp" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:gravity="center"
        android:layout_marginTop="20dp">
        <Button
            android:id="@+id/btnAdd"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="+" />
        <Button
            android:id="@+id/btnSubtract"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="-"
            android:layout_marginLeft="10dp" />
        <Button
            android:id="@+id/btnMultiply"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="×"
            android:layout_marginLeft="10dp" />
        <Button
            android:id="@+id/btnDivide"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="÷"
            android:layout_marginLeft="10dp" />
    </LinearLayout>
    <TextView
        android:id="@+id/tvResult"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Result: "
        android:textSize="18sp"
        android:gravity="center"
        android:layout_marginTop="20dp" />

</LinearLayout>
```

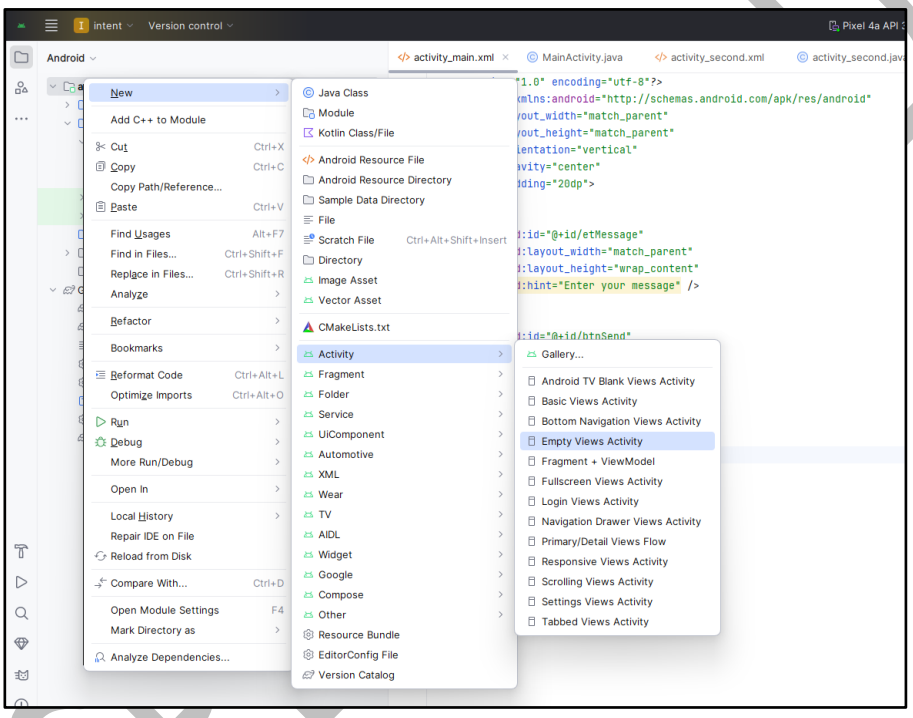
MainActivity.Java

```
package com.example.simplecalc;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    EditText etNumber1, etNumber2;
    Button btnAdd, btnSubtract, btnMultiply, btnDivide;
    TextView tvResult;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        etNumber1 = findViewById(R.id.etNumber1);
        etNumber2 = findViewById(R.id.etNumber2);
        btnAdd = findViewById(R.id.btnAdd);
        btnSubtract = findViewById(R.id.btnSubtract);
        btnMultiply = findViewById(R.id.btnMultiply);
        btnDivide = findViewById(R.id.btnDivide);
        tvResult = findViewById(R.id.tvResult);

        btnAdd.setOnClickListener(v -> calculate('+'));
        btnSubtract.setOnClickListener(v -> calculate('-'));
        btnMultiply.setOnClickListener(v -> calculate('*'));
        btnDivide.setOnClickListener(v -> calculate('/'));
    }
    private void calculate(char operation) {
        String num1Str = etNumber1.getText().toString();
        String num2Str = etNumber2.getText().toString();

        if (num1Str.isEmpty() || num2Str.isEmpty()) {
            Toast.makeText(this, "Please enter both numbers",
Toast.LENGTH_SHORT).show();
            return;
        }

        double num1 = Double.parseDouble(num1Str);
        double num2 = Double.parseDouble(num2Str);
        double result = 0;

        switch (operation) {
            case '+': result = num1 + num2; break;
            case '-': result = num1 - num2; break;
            case '*': result = num1 * num2; break;
            case '/':
                if (num2 == 0) {
                    Toast.makeText(this, "Cannot divide by zero",
Toast.LENGTH_SHORT).show();
                    return;
                }
                result = num1 / num2;
                break;
        }

        tvResult.setText("Result: " + result);
    }
}
```

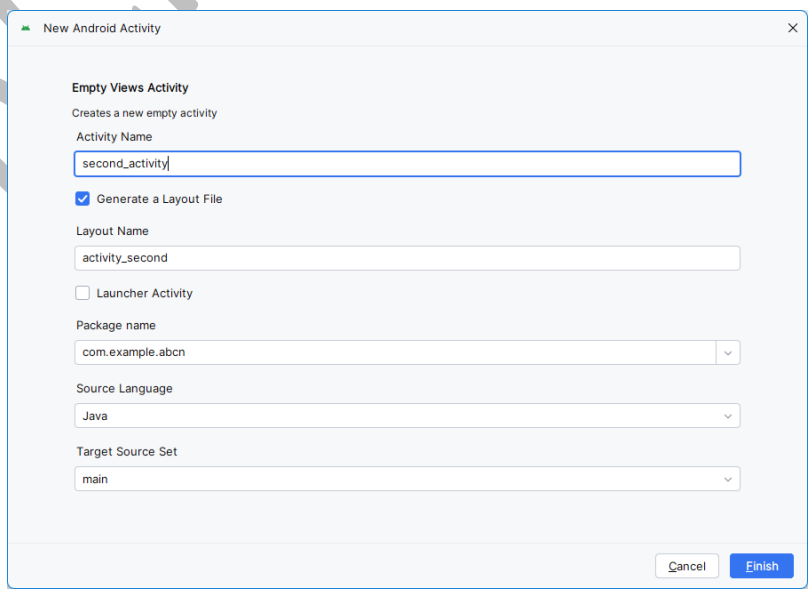
Practical 8: Demonstrate use of intent in android

Steps to Create This Example:

1. Create a new Android Studio project with an Empty Activity.
2. One activity "**MainActivity.java**" is already added in the project by default.
3. You need to add another second activity in application.
    - Right click on **app** folder
    - Click on new-> Activity -> Empty Views Activity



4. Give the name **second_activity**



Code:

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="20dp">

    <EditText
        android:id="@+id/etMessage"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter your message" />

    <Button
        android:id="@+id/btnSend"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Send to Next Activity"
        android:layout_marginTop="10dp" />
</LinearLayout>
```

MainActivity.Java

```
package com.example.intent;

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 {

    EditText etMessage;
    Button btnSend;

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

        etMessage = findViewById(R.id.etMessage);
        btnSend = findViewById(R.id.btnSend);

        btnSend.setOnClickListener(v -> {
            String message = etMessage.getText().toString();

            // Creating an Intent to go to SecondActivity
            Intent intent = new Intent(MainActivity.this,
activity_second.class);

            // Passing data to the next activity
            intent.putExtra("message_key", message);

            // Starting the SecondActivity
            startActivity(intent);
        });
    }
}
```

activity_second.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="20dp">

    <TextView
        android:id="@+id/tvReceivedMessage"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Message will appear here"
        android:textSize="18sp"
        android:gravity="center" />

</LinearLayout>
```

activity_second.java:

```
package com.example.intent;

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

public class activity_second extends AppCompatActivity {

    TextView tvReceivedMessage;

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

        tvReceivedMessage = findViewById(R.id.tvReceivedMessage);

        // Getting the Intent that started this activity
        String message = getIntent().getStringExtra("message_key");

        // Displaying the received message
        tvReceivedMessage.setText("Received Message: " + message);
    }
}
```

Practical 09: Create application to demonstrate menu option.

Step 1: Create empty views application

Step 2: Create the Menu XML File

📌 **Go to:** res/menu/
📌 **IF the** menu **folder doesn't exist, create it.**
📌 **Create a new file:** menu_main.xml

Code of menu_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:app="http://schemas.android.com/apk/res-auto">

    <item
        android:id="@+id/action_settings"
        android:title="Settings"
        android:icon="@android:drawable/ic_menu_preferences"
        app:showAsAction="never"/>

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

    <item
        android:id="@+id/action_exit"
        android:title="Exit"
        android:icon="@android:drawable/ic_menu_close_clear_cancel"
        app:showAsAction="never"/>
</menu>
```

Step 3:

Code of MainActivity.java

```
package com.example.menuoption;

import android.os.Bundle;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

    // Inflate the menu
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        MenuInflater inflater = getMenuInflater();
        inflater.inflate(R.menu.menu_main, menu);
        return true;
```

```
    }

    // Handle menu item clicks
    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        int id = item.getItemId();

        if (id == R.id.action_settings) {
            Toast.makeText(this, "Settings Selected",
Toast.LENGTH_SHORT).show();
            return true;
        } else if (id == R.id.action_about) {
            Toast.makeText(this, "About Selected", Toast.LENGTH_SHORT).show();
            return true;
        } else if (id == R.id.action_exit) {
            finish(); // Close the app
            return true;
        }

        return super.onOptionsItemSelected(item);
    }
}
```

Step 4:

Code of 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:gravity="center"
    android:orientation="vertical">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Menu Demo App"
        android:textSize="20sp"
        android:textStyle="bold"/>

</LinearLayout>
```

Step 4: Update **themes.xml**

📌 Go to: **res/values/themes.xml**

🔹 Modify the parent theme to enable the action bar:

```
<style name="Theme.MyApp"
parent="Theme.MaterialComponents.Light.DarkActionBar">
```

Step 5: Go to: AndroidManifest.xml

correct code in AndroidManifest.xml file.

```
<application
    android:theme="@style/Theme.MyApp">
```

Step 6: Run the app

Practical 10: Create application to demonstrate progress bar.

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="20dp">

    <!-- Circular Progress Bar -->
    <ProgressBar
        android:id="@+id/progress_circular"
        style="?android:attr/progressBarStyleLarge"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:visibility="gone"/>

    <!-- Horizontal Progress Bar -->
    <ProgressBar
        android:id="@+id/progress_horizontal"
        style="?android:attr/progressBarStyleHorizontal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:progress="0"
        android:max="100"
        android:visibility="gone"
        android:layout_marginTop="20dp"/>

    <!-- Start Progress Button -->
    <Button
        android:id="@+id/btn_start"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Start Progress"
        android:layout_marginTop="20dp"/>
</LinearLayout>
```

MainActivity.java

```
package com.example.progressapp;

import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.Button;
import android.widget.ProgressBar;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private ProgressBar progressCircular, progressHorizontal;
    private Button btnStart;
    private int progressStatus = 0;
    private Handler handler = new Handler();

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

```
        // Initialize UI elements
        progressCircular = findViewById(R.id.progress_circular);
        progressHorizontal = findViewById(R.id.progress_horizontal);
        btnStart = findViewById(R.id.btn_start);

        btnStart.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // Show progress bars
                progressCircular.setVisibility(View.VISIBLE);
                progressHorizontal.setVisibility(View.VISIBLE);
                progressStatus = 0;

                // Simulate progress update
                new Thread(new Runnable() {
                    @Override
                    public void run() {
                        while (progressStatus < 100) {
                            progressStatus += 10;
                            handler.post(new Runnable() {
                                @Override
                                public void run() {

progressHorizontal.setProgress(progressStatus);
                                }
                            });
                            try {
                                Thread.sleep(500); // Simulate work being done
                            } catch (InterruptedException e) {
                                e.printStackTrace();
                            }
                        }
                        // Hide circular progress bar when done
                        handler.post(new Runnable() {
                            @Override
                            public void run() {
                                progressCircular.setVisibility(View.GONE);
                            }
                        });
                    }
                }).start();
            }
        });
    }
}
```