Q1. Construct an Android application to accept two numbers in two EditText, with four buttons as ADD, SUB, DIV and MULT and display Result using Toast Control.

<?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”

Android:padding=”16dp”

Tools:context=”.MainActivity”>

<EditText

Android:id=”@+id/num1EditText”

Android:layout\_width=”match\_parent”

Android:layout\_height=”wrap\_content”

Android:hint=”Enter Number 1”

Android:inputType=”numberDecimal” />

<EditText

Android:id=”@+id/num2EditText”

Android:layout\_width=”match\_parent”

Android:layout\_height=”wrap\_content”

Android:layout\_below=”@id/num1EditText”

Android:layout\_marginTop=”16dp”

Android:hint=”Enter Number 2”

Android:inputType=”numberDecimal” />

<Button

Android:id=”@+id/addButton”

Android:layout\_width=”wrap\_content”

Android:layout\_height=”wrap\_content”

Android:layout\_below=”@id/num2EditText”

Android:layout\_marginTop=”16dp”

Android:text=”ADD” />

<Button

Android:id=”@+id/subButton”

Android:layout\_width=”wrap\_content”

Android:layout\_height=”wrap\_content”

Android:layout\_below=”@id/addButton”

Android:layout\_marginTop=”16dp”

Android:text=”SUB” />

<Button

Android:id=”@+id/multButton”

Android:layout\_width=”wrap\_content”

Android:layout\_height=”wrap\_content”

Android:layout\_below=”@id/subButton”

Android:layout\_marginTop=”16dp”

Android:text=”MULT” />

<Button

Android:id=”@+id/divButton”

Android:layout\_width=”wrap\_content”

Android:layout\_height=”wrap\_content”

Android:layout\_below=”@id/multButton”

Android:layout\_marginTop=”16dp”

Android:text=”DIV” />

</RelativeLayout>

**Main.java**

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 num1EditText, num2EditText;

Private Button addButton, subButton, multButton, divButton;

@Override

Protected void onCreate(Bundle savedInstanceState) {

Super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

num1EditText = findViewById(R.id.num1EditText);

num2EditText = findViewById(R.id.num2EditText);

addButton = findViewById(R.id.addButton);

subButton = findViewById(R.id.subButton);

multButton = findViewById(R.id.multButton);

divButton = findViewById(R.id.divButton);

addButton.setOnClickListener(new View.OnClickListener() {

@Override

Public void onClick(View v) {

performOperation(“+”);

}

});

subButton.setOnClickListener(new View.OnClickListener() {

@Override

Public void onClick(View v) {

performOperation(“-“);

}

});

multButton.setOnClickListener(new View.OnClickListener() {

@Override

Public void onClick(View v) {

performOperation(“\*”);

}

});

divButton.setOnClickListener(new View.OnClickListener() {

@Override

Public void onClick(View v) {

performOperation(“/”);

}

});

}

Private void performOperation(String operation) {

String num1Str = num1EditText.getText().toString();

String num2Str = num2EditText.getText().toString();

If (num1Str.isEmpty() || num2Str.isEmpty()) {

showToast(“Please enter both numbers”);

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) {

Result = num1 / num2;

} else {

showToast(“Cannot divide by zero”);

return;

}

Break;

}

showToast(“Result: “ + result);

}

Private void showToast(String message) {

Toast.makeText(this, message, Toast.LENGTH\_SHORT).show();

}

}

Q2. Construct a bank app to display different menu like withdraw, deposit etc.

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

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

Xmlns:tools=<http://schemas.android.com/tools>

Android:layout\_width=”match\_parent”

Android:layout\_height=”match\_parent”

Android:orientation=”vertical”

Android:padding=”16dp”

Tools:context=”.MainActivity”>

<TextView

Android:id=”@+id/accountNumberTextView”

Android:layout\_width=”wrap\_content”

Android:layout\_height=”wrap\_content”

Android:text=”Account Number: “

Android:textSize=”18sp”

Android:textStyle=”bold” />

<TextView

Android:id=”@+id/accountTypeTextView”

Android:layout\_width=”wrap\_content”

Android:layout\_height=”wrap\_content”

Android:text=”Account Type: “

Android:textSize=”18sp”

Android:textStyle=”bold” />

<TextView

Android:id=”@+id/balanceTextView”

Android:layout\_width=”wrap\_content”

Android:layout\_height=”wrap\_content”

Android:text=”Balance: “

Android:textSize=”18sp”

Android:textStyle=”bold” />

<Button

Android:id=”@+id/withdrawButton”

Android:layout\_width=”match\_parent”

Android:layout\_height=”wrap\_content”

Android:text=”Withdraw” />

<Button

Android:id=”@+id/depositButton”

Android:layout\_width=”match\_parent”

Android:layout\_height=”wrap\_content”

Android:text=”Deposit” />

</LinearLayout>

**Main.java**

Import android.os.Bundle;

Import android.view.View;

Import android.widget.Button;

Import android.widget.TextView;

Import androidx.appcompat.app.AppCompatActivity;

Public class MainActivity extends AppCompatActivity {

Private TextView accountNumberTextView, accountTypeTextView, balanceTextView;

Private Button withdrawButton, depositButton;

Private int accountNumber = 123456789;

Private String accountType = “Savings”;

Private double balance = 1000.0;

@Override

Protected void onCreate(Bundle savedInstanceState) {

Super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

accountNumberTextView = findViewById(R.id.accountNumberTextView);

accountTypeTextView = findViewById(R.id.accountTypeTextView);

balanceTextView = findViewById(R.id.balanceTextView);

withdrawButton = findViewById(R.id.withdrawButton);

depositButton = findViewById(R.id.depositButton);

accountNumberTextView.setText(“Account Number: “ + accountNumber);

accountTypeTextView.setText(“Account Type: “ + accountType);

balanceTextView.setText(“Balance: $” + balance);

withdrawButton.setOnClickListener(new View.OnClickListener() {

@Override

Public void onClick(View v) {

// Implement withdraw logic here

// For simplicity, let’s assume a fixed withdrawal amount

Double withdrawalAmount = 100.0;

If (balance >= withdrawalAmount) {

Balance -= withdrawalAmount;

updateBalance();

}

}

});

depositButton.setOnClickListener(new View.OnClickListener() {

@Override

Public void onClick(View v) {

// Implement deposit logic here

// For simplicity, let’s assume a fixed deposit amount

Double depositAmount = 200.0;

Balance += depositAmount;

updateBalance();

}

});

}

Private void updateBalance() {

balanceTextView.setText(“Balance: $” + balance);

}

}