Q2)

ACTIVITY\_MAIN.XML:

<?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:id="@+id/activity\_main"

android:orientation="vertical"

tools:context=".MainActivity">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Guess a number between 1 and 1000"

android:textAlignment="center"/>

<EditText

android:id="@+id/guessEditText"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Enter in a number"

android:inputType="number"/>

<Button

android:id="@+id/guessButton"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Guess"

android:gravity="center"

android:onClick="onGuessSubmit"

/>

</LinearLayout>

MAINACTIVITY:

package com.example.guessinggame;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import java.util.Random;

public class MainActivity extends AppCompatActivity {

private int answer;

public enum RandomNumberRange {

MAX(100), MIN(1);

private RandomNumberRange(int value)

{

this.value = value;

}

private int value;

public int getValue()

{

return this.value;

}

public void setValue(int value)

{

this.value = value;

}

}

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

answer = createRandomNumber();

}

private int createRandomNumber() {

Random randy = new Random();

return randy.nextInt(MainActivity.RandomNumberRange.MAX.getValue()) + MainActivity.RandomNumberRange.MIN.getValue();

}

private Boolean isGuessCorrect(int userGuess) {

return userGuess == answer;

}

private void validateAndCheckGuess(String userInput) {

try {

int guessInput = Integer.parseInt(userInput);

if (guessInput <= RandomNumberRange.MIN.getValue()) {

Toast.makeText(this, "Invalid input", Toast.LENGTH\_SHORT).show();

} else {

checkGuess(guessInput);

}

} catch (NumberFormatException e){

Toast.makeText(this, "Invalid input", Toast.LENGTH\_SHORT).show();

}

}

private void checkGuess(int guessInput) {

if (isGuessCorrect(guessInput)) {

Toast.makeText(this, "You guessed correctly!", Toast.LENGTH\_LONG).show();

clearUserInputEditText();

answer = createRandomNumber();

} else {

String message = "The correct answer is lower";

if (guessInput>answer) {

message = "The correct answer is higher";

}

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

}

}

private void clearUserInputEditText() {

EditText userEditText = findViewById(R.id.guessEditText);

userEditText.setText("");

}

public void onGuessSubmit(View view) {

EditText userEditText = findViewById(R.id.guessEditText);

String userInput = userEditText.getText().toString();

validateAndCheckGuess(userInput);

}

}

Output:

