20CYS404
Android Application Development

END SEM LAB EXAM

Name: Fiyan Mehfil Ayoob

Roll No: CH.EN.U4CYS21018

Aim:

To develop an Android application to play TicTacToe and print result after 5 games

Code:

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"
  android:background="#E0F7FA"
  tools:context=".MainActivity">

  <ImageView
    android:id="@+id/logoImageView"
    android:layout_width="70dp"
    android:layout_height="70dp"

    android:src="@drawable/amrita"
    android:layout_marginTop="20dp"/>

  <TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Fiyan Mehfil Ayoob - 21018"
    android:textSize="24sp"
    tools:layout_editor_absoluteX="56dp"
    tools:layout_editor_absoluteY="16dp"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    android:textColor="#004D40"
    android:layout_marginTop="20dp"
    android:layout_marginLeft="70dp"/>

  <TextView
    android:id="@+id/statusText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="100dp"
    android:text="Player Java's Turn"
    android:textColor="#004D40"
    android:textSize="24sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<Button
    android:id="@+id/btn_java"
    android:layout_width="100dp"
    android:layout_height="100dp"
    android:text="Java"
    android:textSize="19sp"
    android:textColor="#000"
    android:background="#FFFFFF"
    app:layout_constraintTop_toBottomOf="@+id/statusText"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toStartOf="@+id/btn_kotlin"
    android:layout_marginTop="20dp" />

<Button
    android:id="@+id/btn_kotlin"
    android:layout_width="100dp"
    android:layout_height="100dp"
    android:text="Kotlin"
    android:textSize="19sp"
    android:textColor="#000"
    android:background="#FFFFFF"
    app:layout_constraintTop_toBottomOf="@+id/statusText"
    app:layout_constraintStart_toEndOf="@+id/btn_java"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="20dp" />

<!-- GridLayout for Tic-Tac-Toe Buttons -->
<GridLayout
    android:id="@+id/gridLayout"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:rowCount="3"
    android:columnCount="3"
    app:layout_constraintTop_toBottomOf="@+id/btn_java"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:textColor="#004D40"
    android:layout_marginTop="32dp">

    <Button
        android:id="@+id/button00"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textColor="#000000"
        android:textSize="30sp" />

    <Button
        android:id="@+id/button01"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textColor="#000000"
        android:textSize="30sp" />

    <Button
        android:id="@+id/button02"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textColor="#000000"
        android:textSize="30sp" />

    <Button
        android:id="@+id/button10"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textColor="#000000"
        android:textSize="30sp" />

    <Button
        android:id="@+id/button11"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textColor="#000000"
        android:textSize="30sp" />

    <Button
        android:id="@+id/button12"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textColor="#000000"
        android:textSize="30sp" />

    <Button
        android:id="@+id/button20"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textColor="#000000"
        android:textSize="30sp" />

    <Button
        android:id="@+id/button21"
        android:layout_width="100dp"
```

```
            android:layout_height="100dp"
            android:textColor="#000000"
            android:textSize="30sp" />

        <Button
            android:id="@+id/button22"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:textColor="#000000"
            android:textSize="30sp" />
    </GridLayout>

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

MainActivity.java

```
package com.example.tictactoe;

import android.content.Intent;
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 boolean playerX = true; // Player X starts
    private int turnCount = 0;
    private int[][] board = new int[3][3]; // 0 for empty, 1 for X, 2 for O
    private int xWins = 0, oWins = 0, draws = 0;
    private int gamesPlayed = 0;

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

        final TextView statusText = findViewById(R.id.statusText);
        statusText.setText("Player Java's Turn");

        final Button[][] buttons = new Button[3][3];
        buttons[0][0] = findViewById(R.id.button00);
        buttons[0][1] = findViewById(R.id.button01);
        buttons[0][2] = findViewById(R.id.button02);
        buttons[1][0] = findViewById(R.id.button10);
```

```
buttons[1][1] = findViewById(R.id.button11);
buttons[1][2] = findViewById(R.id.button12);
buttons[2][0] = findViewById(R.id.button20);
buttons[2][1] = findViewById(R.id.button21);
buttons[2][2] = findViewById(R.id.button22);

// Click listener for the grid buttons
View.OnClickListener listener = new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Button button = (Button) v;
        String tag = button.getTag().toString();
        int row = Character.getNumericValue(tag.charAt(0));
        int col = Character.getNumericValue(tag.charAt(1));

        if (board[row][col] == 0) {
            board[row][col] = playerX ? 1 : 2;
            button.setText(playerX ? "X" : "O");
            button.setEnabled(false);
            turnCount++;

            if (checkWinner()) {
                if (playerX) {
                    xWins++;
                    statusText.setText("Player Java Wins!");
                } else {
                    oWins++;
                    statusText.setText("Player Kotlin Wins!");
                }
                resetBoard(buttons);
            } else if (turnCount == 9) {
                draws++;
                statusText.setText("It's a Draw!");
                resetBoard(buttons);
            } else {
                playerX = !playerX;
                statusText.setText(playerX ? "Player Java's Turn" : "Player Kotlin's Turn");
            }

            // If 5 games are played, move to ResultActivity
            if (gamesPlayed == 5) {
                Intent intent = new Intent(MainActivity.this, ResultsActivity.class);
                intent.putExtra("xWins", xWins);
                intent.putExtra("oWins", oWins);
                intent.putExtra("draws", draws);
                startActivity(intent);
            }
        }
```

```
            }
        };

        // Attach listeners to each button
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                buttons[i][j].setOnClickListener(listener);
                buttons[i][j].setTag("" + i + j);
            }
        }
    }

    private boolean checkWinner() {
        // Check rows, columns, and diagonals
        for (int i = 0; i < 3; i++) {
            if (board[i][0] == board[i][1] && board[i][1] == board[i][2] && board[i][0] != 0) {
                return true;
            }
            if (board[0][i] == board[1][i] && board[1][i] == board[2][i] && board[0][i] != 0) {
                return true;
            }
        }
        if (board[0][0] == board[1][1] && board[1][1] == board[2][2] && board[0][0] != 0) {
            return true;
        }
        if (board[0][2] == board[1][1] && board[1][1] == board[2][0] && board[0][2] != 0) {
            return true;
        }
        return false;
    }

    private void resetBoard(Button[][] buttons) {
        board = new int[3][3];
        turnCount = 0;
        playerX = true;
        gamesPlayed++;

        // Reset all buttons
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                buttons[i][j].setEnabled(true);
                buttons[i][j].setText("");
            }
        }
    }
}
```

Activity_results.xml

```
package com.example.tictactoe;

import android.content.Intent;
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 boolean playerX = true; // Player X starts
    private int turnCount = 0;
    private int[][] board = new int[3][3]; // 0 for empty, 1 for X, 2 for O
    private int xWins = 0, oWins = 0, draws = 0;
    private int gamesPlayed = 0;

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

        final TextView statusText = findViewById(R.id.statusText);
        statusText.setText("Player Java's Turn");

        final Button[][] buttons = new Button[3][3];
        buttons[0][0] = findViewById(R.id.button00);
        buttons[0][1] = findViewById(R.id.button01);
        buttons[0][2] = findViewById(R.id.button02);
        buttons[1][0] = findViewById(R.id.button10);
        buttons[1][1] = findViewById(R.id.button11);
        buttons[1][2] = findViewById(R.id.button12);
        buttons[2][0] = findViewById(R.id.button20);
        buttons[2][1] = findViewById(R.id.button21);
        buttons[2][2] = findViewById(R.id.button22);

        // Click listener for the grid buttons
        View.OnClickListener listener = new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Button button = (Button) v;
                String tag = button.getTag().toString();
                int row = Character.getNumericValue(tag.charAt(0));
                int col = Character.getNumericValue(tag.charAt(1));
```

```
      if (board[row][col] == 0) {
        board[row][col] = playerX ? 1 : 2;
        button.setText(playerX ? "X" : "O");
        button.setEnabled(false);
        turnCount++;

        if (checkWinner()) {
          if (playerX) {
            xWins++;
            statusText.setText("Player Java Wins!");
          } else {
            oWins++;
            statusText.setText("Player Kotlin Wins!");
          }
          resetBoard(buttons);
        } else if (turnCount == 9) {
          draws++;
          statusText.setText("It's a Draw!");
          resetBoard(buttons);
        } else {
          playerX = !playerX;
          statusText.setText(playerX ? "Player Java's Turn" : "Player Kotlin's Turn");
        }

        // If 5 games are played, move to ResultActivity
        if (gamesPlayed == 5) {
          Intent intent = new Intent(MainActivity.this, ResultsActivity.class);
          intent.putExtra("xWins", xWins);
          intent.putExtra("oWins", oWins);
          intent.putExtra("draws", draws);
          startActivity(intent);
        }
      }
    }
  };

  // Attach listeners to each button
  for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
      buttons[i][j].setOnClickListener(listener);
      buttons[i][j].setTag("" + i + j);
    }
  }
}

private boolean checkWinner() {
  // Check rows, columns, and diagonals
  for (int i = 0; i < 3; i++) {
```

```
            if (board[i][0] == board[i][1] && board[i][1] == board[i][2] && board[i][0] != 0) {
                return true;
            }
            if (board[0][i] == board[1][i] && board[1][i] == board[2][i] && board[0][i] != 0) {
                return true;
            }
        }
        if (board[0][0] == board[1][1] && board[1][1] == board[2][2] && board[0][0] != 0) {
            return true;
        }
        if (board[0][2] == board[1][1] && board[1][1] == board[2][0] && board[0][2] != 0) {
            return true;
        }
        return false;
    }

    private void resetBoard(Button[][] buttons) {
        board = new int[3][3];
        turnCount = 0;
        playerX = true;
        gamesPlayed++;

        // Reset all buttons
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                buttons[i][j].setEnabled(true);
                buttons[i][j].setText("");
            }
        }
    }
}
```

ResultsActivity.java

```
package com.example.tictactoe;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class ResultsActivity extends AppCompatActivity {

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

        TextView resultText = findViewById(R.id.resultText);
        Button resetButton = findViewById(R.id.btn_reset_results);

        // Retrieve Intent data
        int xWins = getIntent().getIntExtra("xWins", -1);
        int oWins = getIntent().getIntExtra("oWins", -1);
        int draws = getIntent().getIntExtra("draws", -1);

        // Check if data was received properly
        if (xWins == -1 || oWins == -1 || draws == -1) {
            // If there's a problem with the data, display an error
            resultText.setText("Error: Could not retrieve game results.");
        } else {
            // Display the results properly
            String results = "Results after 5 games:\n" +
                    "Player Java Wins: " + xWins + "\n" +
                    "Player Kotlin Wins: " + oWins + "\n" +
                    "Draws: " + draws;
            resultText.setText(results);
        }

        // Reset button functionality
        resetButton.setOnClickListener(v -> {
            // Optionally, start a new game or navigate back to the main activity
            Intent intent = new Intent(ResultsActivity.this, MainActivity.class); // Navigate back to
MainActivity
            startActivity(intent);
            finish(); // Optional: to close the ResultsActivity and clear it from the back stack
        });
    }
}
```

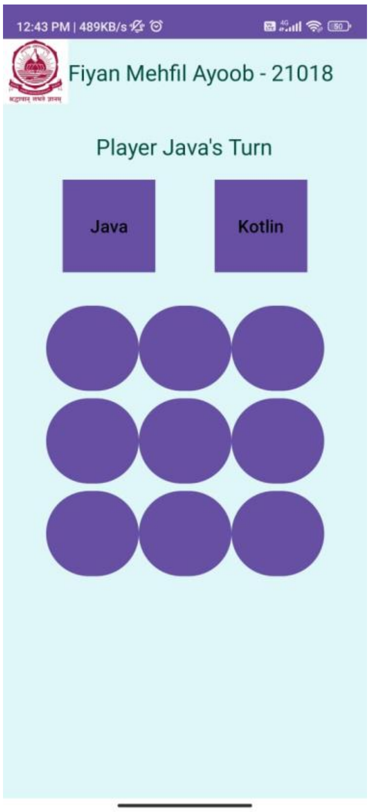
AndroidManifest.xml

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

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.TicTacToe"
        tools:targetApi="31">

        <!-- MainActivity requires android:exported="true" because it has an intent filter -->
        <activity
            android:name=".MainActivity"
            android:exported="true">  <!-- Explicitly set android:exported to true -->
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <!-- You can add other activities like ResultActivity without intent filters -->
        <activity
            android:name=".ResultsActivity"
            android:exported="false" /> <!-- Set exported to false, since it doesn't need to be launched
externally -->

    </application>

</manifest>
```

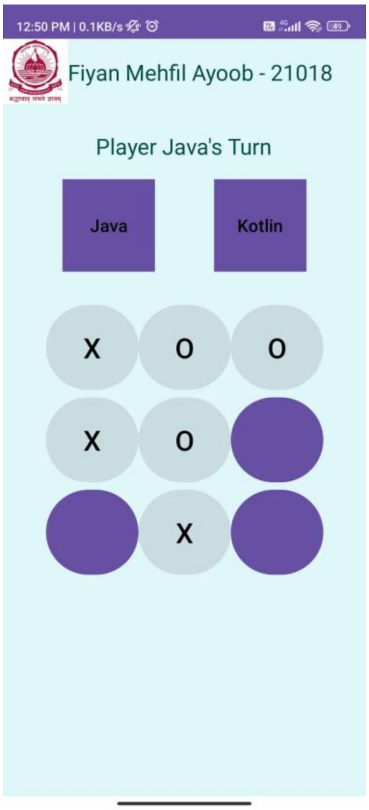
Style.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <!-- Base application theme. -->
    <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">#008080</item>
        <item name="android:statusBarColor">#008080</item>
        <item name="colorPrimaryDark">#008080</item>
        <item name="colorAccent">#008080</item>
    </style>

    <style name="ButtonStyle">
        <item name="android:layout_width">100dp</item>
        <item name="android:layout_height">100dp</item>
        <item name="android:textSize">30sp</item>
        <item name="android:background">?attr/selectableItemBackground</item>
    </style>

    <style name="TextStyle">
        <item name="android:textSize">24sp</item>
    <item name="android:layout_marginTop">16dp</item>
    <item name="android:textColor">#000000</item>
  </style>
</resources>
```

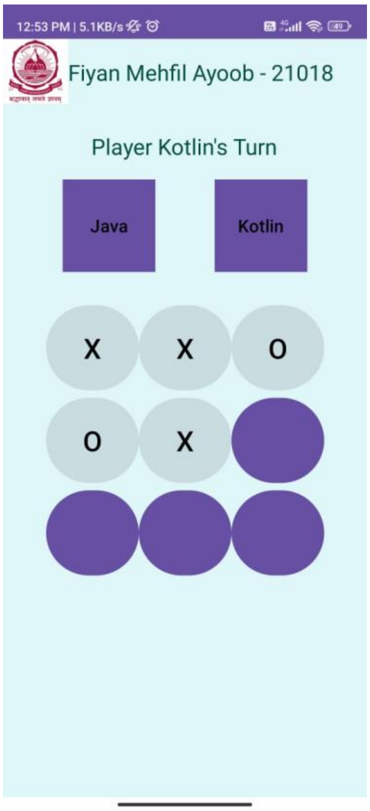
Output


Fiyan Mehfil Ayoob - 21018

Player Java's Turn


Fiyan Mehfil Ayoob - 21018

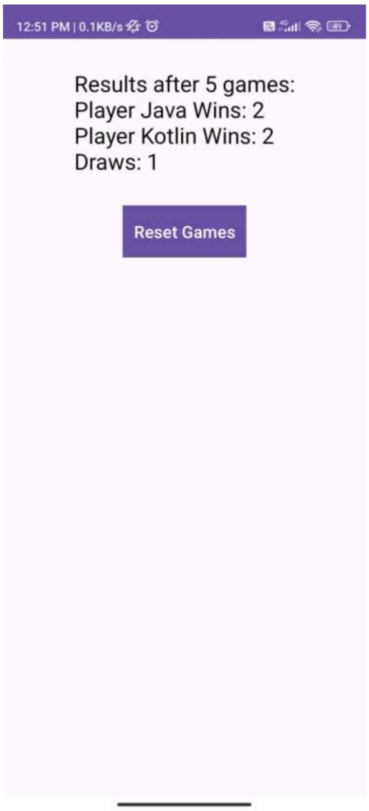
Player Java's Turn



Fiyan Mehfil Ayoob - 21018

Player Kotlin's Turn

X X O
O X



Results after 5 games:
Player Java Wins: 2
Player Kotlin Wins: 2
Draws: 1