

## Lab 2: Android Programming Report

學號：109511286

姓名：蔡佩蓉

### 1. 請說明下列程式作用

```
<intent-filter>
    <action android:name="android.intent.action.MAIN" />
    <category android:name="android.intent.category.LAUNCHER"/>
</intent-filter>
```

這段程式碼是指定 Android 應用程式的入口點（entry point），它是在 AndroidManifest.xml 檔案中的 <activity> 標籤中使用的。這段程式碼位於 <activity> 標籤內的 <intent-filter> 標籤中，其作用是告訴 Android 系統這個特定的活動（activity）是應用程式的主要活動（main activity），也就是應用程式啟動時首先運行的活動。

<intent-filter> 指定活動、服務或廣播接收器可回應的 intent 類型。Intent filter 會宣告其父項元件的功能，包括活動或服務可執行的作業，以及接收器可處理的廣播類型。

<action android:name="android.intent.action.MAIN" />：新增動作至 intent filter。<intent-filter> 元素必須包含一或多個 <action> 元素。如果 intent filter 中沒有任何 <action> 元素，該 filter 就不會接受任何 Intent 物件。

部分標準動作在 Intent 類別中定義為 ACTION\_string 常數。如要將其中一個動作指派給這個屬性，請在 ACTION\_ 後方的 string 前面加上 android.intent.action.。

**android.intent.action.MAIN (constant value of ACTION\_MAIN)：**決定應用的入口 Activity，也就是我們啟動應用時首先顯示哪一個 Activity。

<category android:name="android.intent.category.LAUNCHER"/>：新增類別名稱至 intent filter。

標準類別在 Intent 類別中定義為 CATEGORY\_name 常數。此處的指派名稱可以從這些常數衍生，在 CATEGORY\_ 後方的 name 加上前置字元 android.intent.category.。

**android.intent.category.LAUNCHER (constant value of CATEGORY\_LAUNCHER)：**表示 activity 應該被列入系統的啟動器（launcher）（允許用戶啟動它）。Launcher 是安卓系統中的桌面啟動器，是桌面 UI 的統稱。

## 2. 請貼上自己的程式碼並附上註解

第一個頁面：

```
package com.example.app109511286_lab2;

import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {
    // 宣告物件變數
    EditText Name;
    EditText Student_ID;
    Button start;
    String strname, strid;

    // 初始化活動
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main); // 載入和顯示XML布局檔

        // 透過ID連結XML布局中的元件
        Name = (EditText) findViewById(R.id.Name);
        Student_ID = (EditText) findViewById(R.id.Student_ID);
        start = (Button) findViewById(R.id.start);

        // 建立物件變數的傾聽者 (Listener) - 點擊監聽器
        start.setOnClickListener(new View.OnClickListener() {
            @Override
            // 當 click button 時執行以下程式
            public void onClick(View view) {
                strname = Name.getText().toString(); // 得到plaintext中輸入的name
                strid = Student_ID.getText().toString(); // 得到plaintext中輸入的學號
                Intent intent = new Intent(); // 建新的Intent並初始化
                intent.setClass(MainActivity.this, MainActivity2.class); // 從此頁面跳到MainActivity2頁面
                Bundle bundle = new Bundle(); // 建新的Bundle放要傳送的数据
                bundle.putString("strname", strname); // 把name放到Bundle中
                bundle.putString("strid", strid); // 把學號放到Bundle中
                intent.putExtras(bundle); // bundle放到intent中傳送
                startActivity(intent); // 啟動由intent指定的MainActivity2頁面
            }
        });
    }
}
```

第二個頁面：

```

package com.example.app109511286_lab2;

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

public class MainActivity2 extends AppCompatActivity {
    // 宣告物件變數
    EditText input;
    TextView hinttextView;
    TextView guesstimetextView;
    TextView bestrecordtextView;
    Button restartbutton;
    Button sendbutton;
    Button quitbutton;
    int randNum=0, history=999; // randNum為要猜的數字，history為最佳紀錄
    int min=1, max=50, count=0; // min和max為上面hint顯示的數字範圍，count為猜測次數

    // 初始化活動
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2); // 載入和顯示XML布局檔

        Bundle bundle = this getIntent().getExtras(); // 得到intent傳來的bundle

        // 透過ID連結XML布局中的元件
        input = (EditText) findViewById(R.id.input);
        hinttextView = (TextView) findViewById(R.id.hinttextView);
        guesstimetextView = (TextView) findViewById(R.id.guesstimetextView);
        bestrecordtextView = (TextView) findViewById(R.id.bestrecordtextView);
        restartbutton = (Button) findViewById(R.id.restartbutton);
        sendbutton = (Button) findViewById(R.id.sendbutton);
        quitbutton = (Button) findViewById(R.id.quitbutton);

        randNum = (int) (Math.random() * 50 + 1); // 產生要猜的數字
        bestrecordtextView.setText("最佳紀錄 : " + history); // 設定最佳紀錄的textView

        // 建立物件變數的傾聽者 (Listener) - 點擊監聽器
        restartbutton.setOnClickListener(new View.OnClickListener() { // 重新開始
            @Override
            // 當 click button 時執行以下程式
            public void onClick(View view)
            {
                randNum = (int) (Math.random() * 50 + 1); // 重新產生數字
                min = 1; max = 50; count = 0; // 初始化min、max、count
                hinttextView.setText("請輸入1~50的數字"); // 初始化textView
                guesstimetextView.setText("猜測次數 : 0"); // 初始化textView
                input.setText(""); // 初始化edittext
            }
        });

        sendbutton.setOnClickListener(new View.OnClickListener() { // 送出答案
            @Override
            public void onClick(View view)
            {
                count++; // 猜測次數 + 1
                guesstimetextView.setText("猜測次數 : " + count); // 設定猜測次數的textView
                int in = Integer.parseInt(input.getText().toString()); // 得到edittext中輸入的數字
                if (in <= max && in >= min) // 在hint給的猜測範圍內
                {
                    if (in > randNum) // 猜太大，改max
                    {
                        max = in;
                        hinttextView.setText("請輸入" + min + "~" + max + "的數字");
                    }
                    else if (in < randNum) // 猜太小，改min
                    {
                        min = in;
                        hinttextView.setText("請輸入" + min + "~" + max + "的數字");
                    }
                    else if (in == randNum) // 猜對了
                    {
                        hinttextView.setText("猜對了");
                        if(count < history) // 比最佳紀錄好
                        {
                            history = count; // 設定最佳紀錄
                            bestrecordtextView.setText("最佳紀錄 : " + history); // 更新textView
                        }
                    }
                }
                else {hinttextView.setText("請輸入" + min + "~" + max + "的數字，請輸入正常值");} // 輸入超過範圍了，error
            }
        });

        quitbutton.setOnClickListener(new View.OnClickListener() { // 結束遊戲
            @Override
            public void onClick(View view)
            {
                Intent intent1 = new Intent(); // 建新的Intent並初始化
                intent1.setClass(MainActivity2.this, MainActivity3.class); // 從此頁面跳到MainActivity3頁面
                bundle.putInt("his", history); // 把得到的bundle再加上最佳紀錄的data
                intent1.putExtras(bundle); // bundle放到intent1中傳送
                startActivity(intent1); // 啟動intent1指定的MainActivity3頁面
            }
        });
    }
}

```

第三個頁面：

```
package com.example.app109511286_lab2;

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

public class MainActivity3 extends AppCompatActivity {
    // 宣告物件變數
    TextView id;
    TextView sname;
    TextView bestrecord;
    Button backbutton;
    String strname, strid;
    String his;

    // 初始化活動
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main3); // 載入和顯示XML布局檔

        Bundle bundle=this.getIntent().getExtras(); // 得到intent傳來的bundle
        strname = (String) bundle.getString("strname"); // 從bundle中得到strname的String
        strid = (String) bundle.getString("strid"); // 從bundle中得到strid的String
        his = String.valueOf((int) bundle.getInt("his")); // 從bundle中得到his的int，再轉成String

        // 透過ID連結XML布局中的元件
        id = (TextView) findViewById(R.id.id);
        sname = (TextView) findViewById(R.id.sname);
        bestrecord = (TextView) findViewById(R.id.bestrecord);
        backbutton = (Button) findViewById(R.id.backbutton);

        id.setText("學號： " + strid); // 設定學號的textview
        sname.setText("姓名： " + strname); // 設定名字的textview
        bestrecord.setText("最佳紀錄： " + his); // 設定最佳紀錄的textview

        // 建立物件變數的傾聽者 (Listener) - 點擊監聽器
        backbutton.setOnClickListener(new View.OnClickListener() { // 跳回第一個頁面
            @Override
            // 當 click button 時執行以下程式
            public void onClick(View v) {
                Intent intent1 = new Intent(); // 建新的Intent並初始化
                intent1.setClass(MainActivity3.this, MainActivity.class); // 從此頁面跳到MainActivity頁面
                startActivity(intent1); // 啟動intent指定的MainActivity頁面
            }
        });
    }
}
```

### 3. 心得

這次的實驗讓我學到如何讓 app 在不同頁面中轉換並傳送資料，可以把資料一個個丟進 intent 中，有或者可以把資料全部用 bundle 包起來，再用 intent 傳送 bundle。中間曾經畫面一直閃退，檢查過後才發現是取出 bundle 的程式碼寫錯了，還以為 Android Studio 又壞掉了。