

Android程式設計

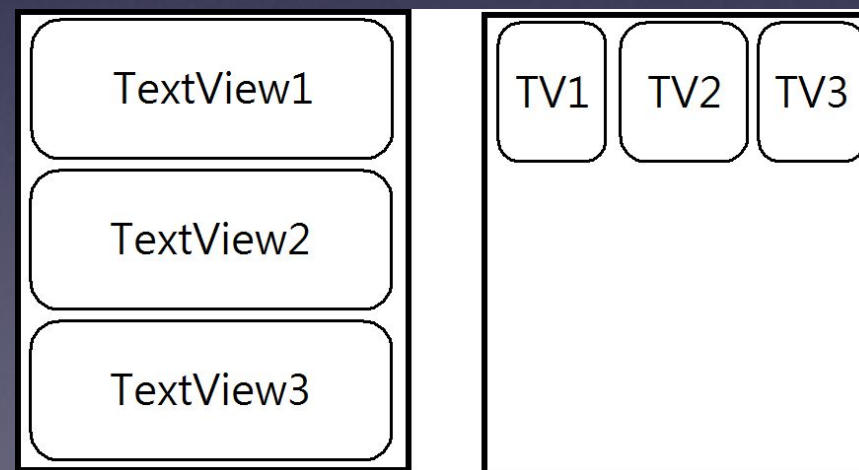
猜數字遊戲

王昱景 Brian Wang

brian.wang.frontline@gmail.com

LinearLayout

- LinearLayout 版面配置是將子介面元件排列成一列（垂直），或一欄（水平），一個接著一個排列成一直線，如下圖所示：



- 在版面配置資源的 XML 文件是使用 LinearLayout 元素來包含子元素的 View 元件，如下所示：

```
<LinearLayout  
    android:orientation="vertical"  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent">  
    .....  
</LinearLayout>
```



android:id

- android:id 屬性是在使用介面中找到指定介面元件的索引
- 單純顯示用途的介面元件不需要指定此屬性，如需要撰寫 Java 程式碼更改介面元件的屬性，或建立事件處理時，記得一定要指定此屬性值
- 例如：建立 Button 元件的事件處理方法

- 在 Java 程式碼可以使用類別常數來找出 Button 元件 button1，如下所示：

```
Button btn1 = (Button) findViewById(R.id.button1);
```

- findViewById() 方法參數是 R.id.button1 類別常數，從資源索引檔 R.java 可以取得 Button 元件 button1 的常數值為0x7f050000

建立事件處理方法

- 在 <Button> 標籤的 android:onClick 屬性指定事件處理方法名稱，以此例是 button1_Click() 方法，如下所示：

```
<Button android:id="@+id/button1"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:text="請按我"  
        android:onClick="button1_Click"/>
```

按鈕元件

- 按鈕（Button）元件是 Android 使用介面中十分重要的介面元件
- 它是實際執行功能的介面元件，也就是觸發事件的元件
- 在日常生活中的按鈕也隨處可見，例如：門鈴和遊戲控制器的按鈕，按一下可以響起門鈴聲，或發射子彈射擊
- 在 Android 使用介面上建立的按鈕可以分為三種：文字按鈕、圖形按鈕和切換按鈕，即 Button、ImageButton 和 ToggleButton 元件

Button

- Button 元件是一種文字按鈕，因為按鈕的標題是文字內容，可以觸發 Click 事件執行事件處理方法（即 onClick 事件處理方法），例如：在輸入資料後，按下按鈕顯示計算結果或更改屬性等操作
- 在版面配置資源的 XML 文件是使用 <Button> 標籤宣告在使用介面建立 Button 元件，如下所示：

```
<Button android:id="@+id/button1"  
        android:layout_width="wrap_content"  
        android:layout_height="100dp"  
        android:text="撲克牌1"/>
```


文字元件

- Android 使用介面的文字元件主要有兩種：TextView 和 EditText 元件，一個可以顯示文字內容；一個是用來輸入文字內容
- 對比應用程式的輸出入，TextView 是程式輸出；EditText 是程式輸入



TextView

- TextView 元件就是系統的標籤控制項，它是一種資料輸出元件，可以顯示程式執行結果
- 例如：按下 Button 元件後，在 TextView 元件顯示數學運算結果。在版面配置資源的 XML 文件是使用 <TextView> 標籤來宣告，如下所示：

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="身高(公分):"/>
```

EditText

- EditText 元件是 TextView 元件的子類別，可以讓使用者以鍵盤輸入程式所需的資料
- 例如：姓名、帳號和電話等，在版面配置資源的 XML 文件是使用 <EditText> 標籤來宣告，如下所示：

```
<EditText android:id="@+id/txtHeight"  
    android:layout_width="fill_parent"  
    android:layout_height="wrap_content"  
    android:inputType="number"/>
```

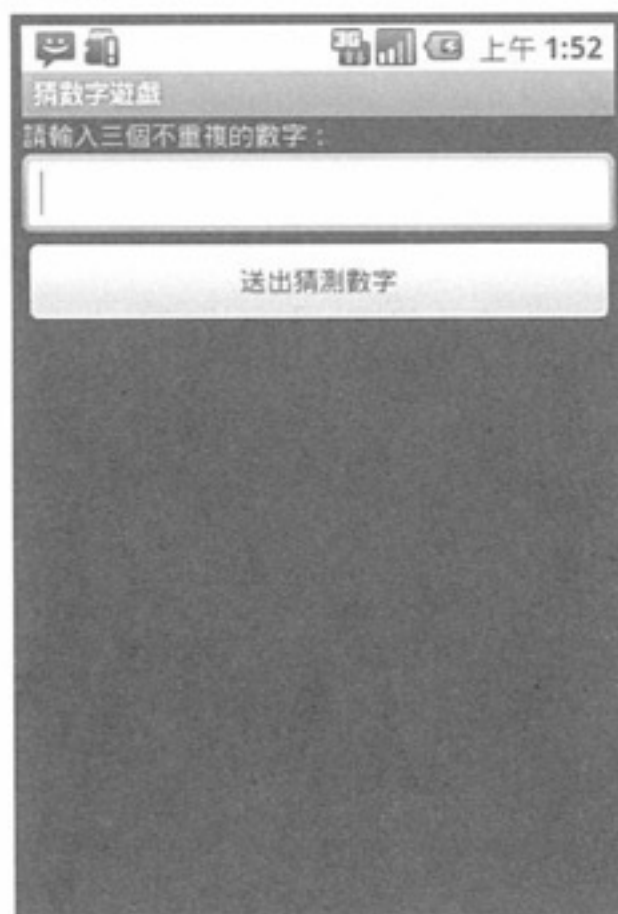
android:inputType

- android:inputType 屬性值的常用輸入類型說明，如下表所示：

屬性值	說明	屬性值	說明
none	唯讀	text	一般文字
textUri	URL網址	number	整數
numberSigned	有符號整數	numberDecimal	浮點數
phone	電話號碼	datetime	日期/時間
date	日期	time	時間
textMultiLine	多行文字	textEmailAddress	電子郵件地址
textPassword	密碼	textVisiblePassword	可見密碼

1. 題目說明：

請開啓 **COM.TQC.GDD02** 專案，設計「猜數字遊戲」。輸入三個不重複的數字，按下「送出猜測數字」按鈕，即可在下方顯示【幾 A 幾 B】的猜測狀況，請依下列題意完成作答。



<參考圖>

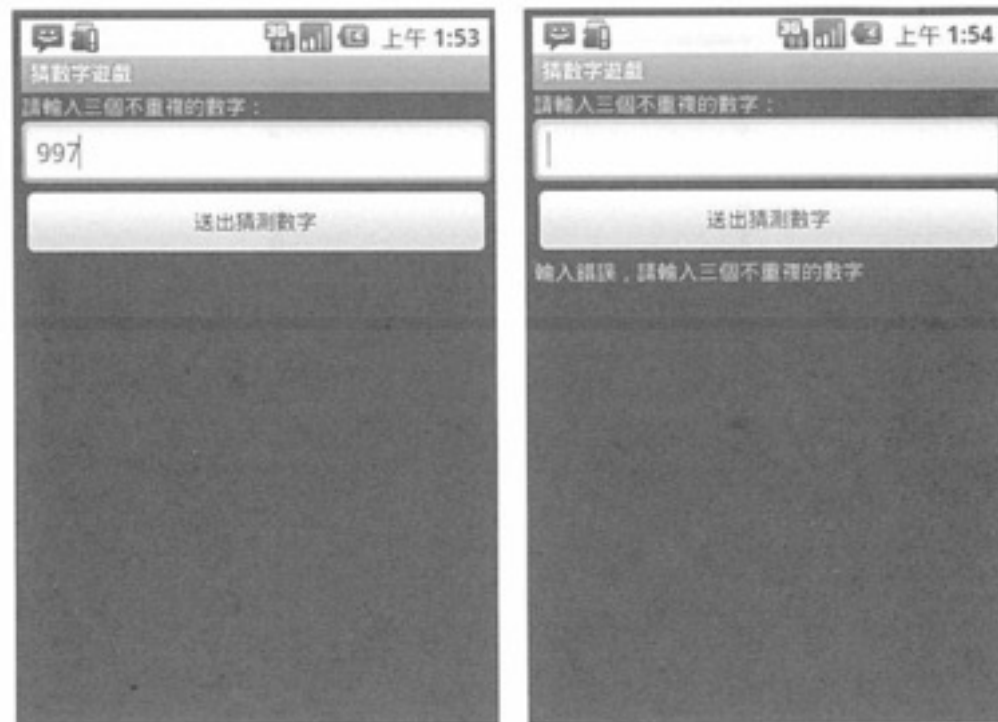
2. 設計說明：

- (1) 請開啓 main.xml 檔案，依參考圖利用 LinearLayout 製作出猜數字遊戲的主要畫面，並開啓 strings.xml 檔案，將字串變數設定於此。
- (2) 在 strings.xml 檔案中，找到 app_name 將值修改為【猜數字遊戲】。
- (3) 請設計一個 string 名稱為 title 的 TextView，string 值設定為【請輸入三個不重複的數字：】。

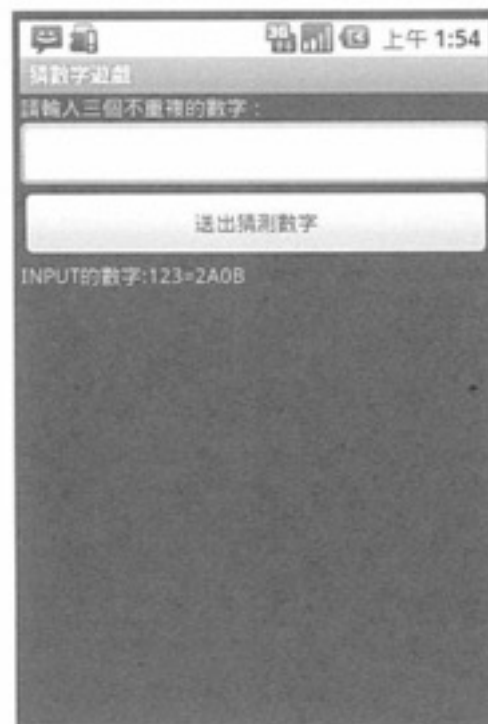
- (4) 請設計一個變數名稱為 `input_number` 的 `EditText`，限制輸入數值為整數且最多不超過三位數。
- (5) 請設計一個變數名稱為 `submit` 的 `Button`，其 `string` 名稱為 `btn`，`string` 值設定為【送出猜測數字】。
- (6) 請設計一個變數名稱為 `respond` 的 `TextView`，`text` 值預設為空。
- (7) 使用者輸入三個數字後，點選「送出猜測數字」按鈕，與電腦隨機抽取三個數字比對。若數字位置皆相同者以 A 表示，若數字相同但位置不相同者以 B 表示。
 - ❖ 例如：電腦預設數字為 123，我們猜測數字為 102，回覆值為 1A1B；猜測數字為 281，回覆值為 0A2B，以此類推。
- (8) 若輸入的三個數字有重複，變數名稱為 `respond` 的 `TextView`，其 `text` 值為【輸入錯誤，請輸入三個不重複的數字】，上方 `EditText` 清空。
- (9) 猜測過程中，點選「送出猜測數字」按鈕，變數名稱為 `respond` 的 `TextView`，其 `text` 值為【INPUT 的數字：xxx=nAmB】，xxx 代入所輸入的數字後，上方 `EditText` 清空，n 代入位置及數值皆正確（A）的總個數，m 代入數值正確，位置錯誤（B）的總個數。
- (10) 若猜對數值，變數名稱為 `respond` 的 `TextView`，其 `text` 值為【INPUT 的數字：xxx 恭喜你！答對囉~答案：xxx】，上方 `EditText`「不」清空。

3. 執行結果參考畫面：

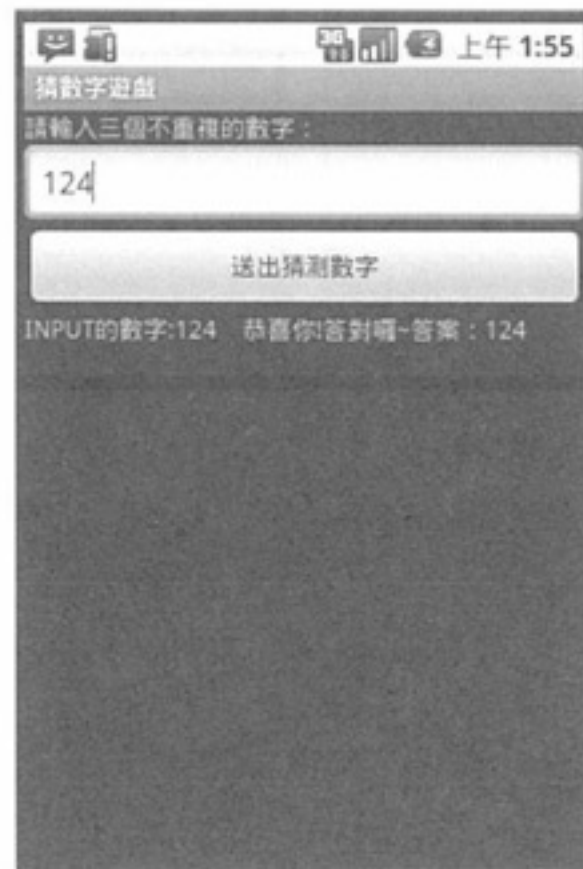
- (1) 若輸入的三個數值有重複，變數名稱為 `respond` 的 `TextView`，其 `text` 值為【輸入錯誤，請輸入三個不重複的數字】，上方 `EditText` 清空。



- (2) 猜測過程中，點選「送出猜測數字」按鈕，變數名稱為 `respond` 的 `TextView`，`text` 值為【INPUT 的數字：xxx=nAmB】，xxx 代入所輸入的數字後，上方 `EditText` 清空，n 代入位置及數值皆正確的總個數，m 代入數值正確，位置錯誤的總個數。



- (3) 若猜對數值，變數名稱為 `respond` 的 `TextView`，`text` 值為【INPUT 的數字：xxx 恭喜你！答對囉~答案：xxx】，xxx 代入所輸入的三個數字，上方 `EditText`「不」清空。



4. 自行測試程式是否達成下列結果：

- (1) 請開啓 main.xml 檔案，依參考圖利用 LinearLayout 製作出猜數字遊戲的主要畫面。
- (2) 開啓 strings.xml 檔案，將字串變數設定於此。
- (3) 變數名稱為 input_number 的 EditText，需限制輸入數值及限制輸入位數。
- (4) 變數名稱為 respond 的 TextView，text 值預設為空。
- (5) 若輸入的三個數值有重複，依設計說明(8)顯示提示訊息，上方 EditText 清空。
- (6) 於點選「送出猜測數字」的按鈕時，依設計說明(9)判斷比對猜測數字的正確性，顯示正確訊息。
- (7) 若猜對數值，依設計說明(10)顯示正確訊息，上方 EditText 不清空。

5. 評分項目：

項	目	配 分	得 分
(1)	畫面元件依 res/layout/main.xml 的定義正確顯示，各個物件及其屬性設定與參考圖相同	3	
(2)	字串變數設定於 strings.xml 檔案內	2	
(3)	變數名稱為 input_number 的 EditText，需限制輸入數值及限制輸入位數	3	
(4)	變數名稱為 respond 的 TextView，text 值預設為空	5	
(5)	若輸入的三個數值有重複，依設計說明(8)顯示提示訊息，上方 EditText 清空	5	
(6)	於點選「送出猜測數字」的按鈕時，依設計說明(9)判斷比對猜測數字的正確性，顯示正確訊息	10	
(7)	若猜對數值，依設計說明(10)顯示正確訊息，上方 EditText「不」清空	2	
總	分	30	

GuessNumber

- 開啟和執行 Android 專案
- 建立元件的版面配置
- 建立 Activity 活動類別


開啟和執行 Android 專案

- 請啟動 Eclipse IDE
- 建立 Android 專案
 - Project Name: GuessNumber
 - Build Target: Android 5.1.1
 - Package Name: tw.edu.vnu.guessnumber

New Android Application

New Android Application

Creates a new Android Application



Application Name: ⓘ

Project Name: ⓘ

Package Name: ⓘ

Minimum Required SDK: ⓘ

Target SDK: ⓘ

Compile With: ⓘ

Theme: ⓘ

💡 The application name is shown in the Play Store, as well as in the Manage Application list in Settings.

?

< Back

Next >

Cancel

Finish

New Android Application

Create Activity

Select whether to create an activity, and if so, what kind of activity.

☒ Create Activity

Blank Activity

Blank Activity with Fragment

Empty Activity

Fullscreen Activity

Master/Detail Flow

Navigation Drawer Activity

Tabbed Activity

Empty Activity

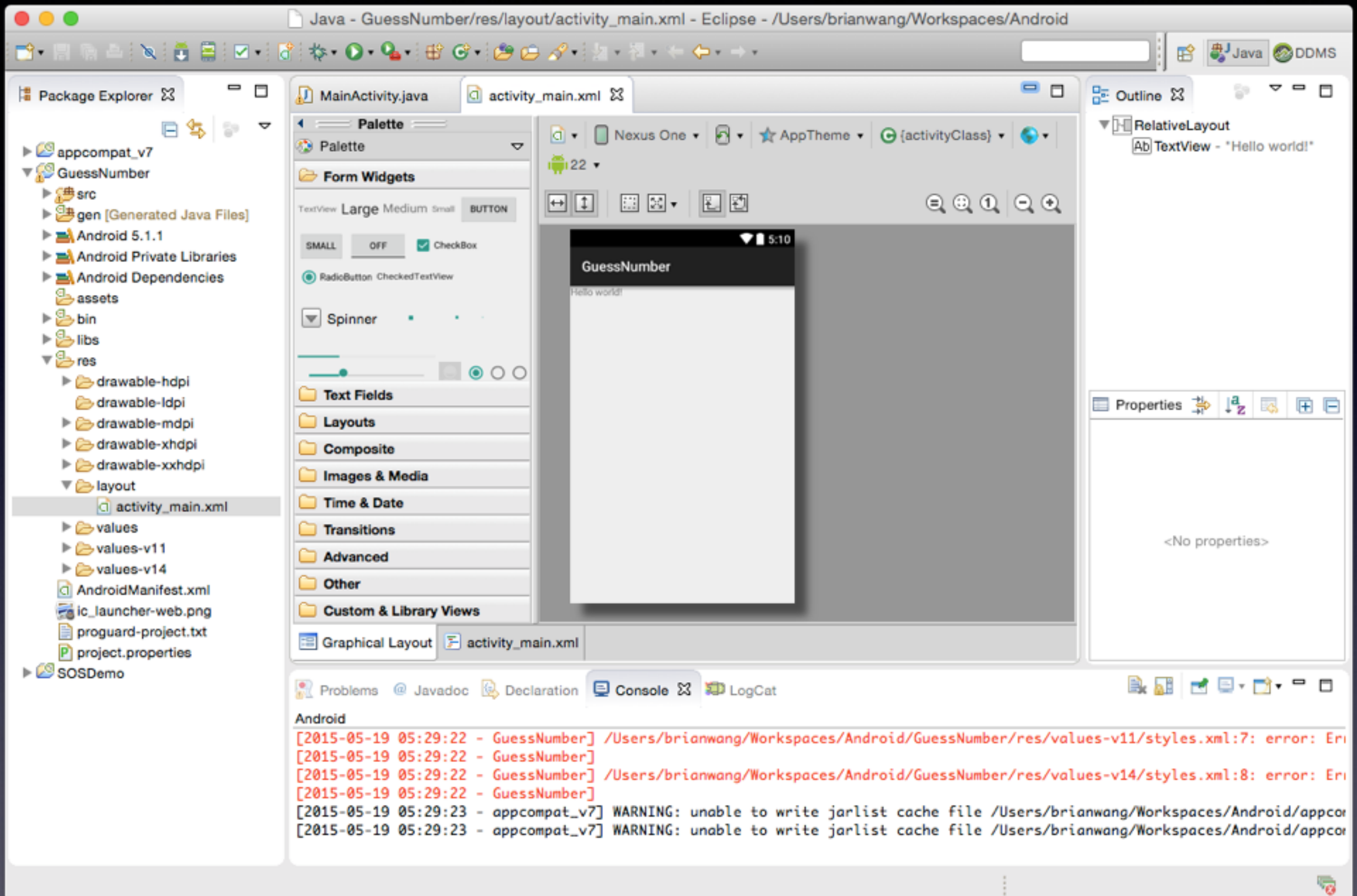
Creates a new empty activity

< Back

Next >

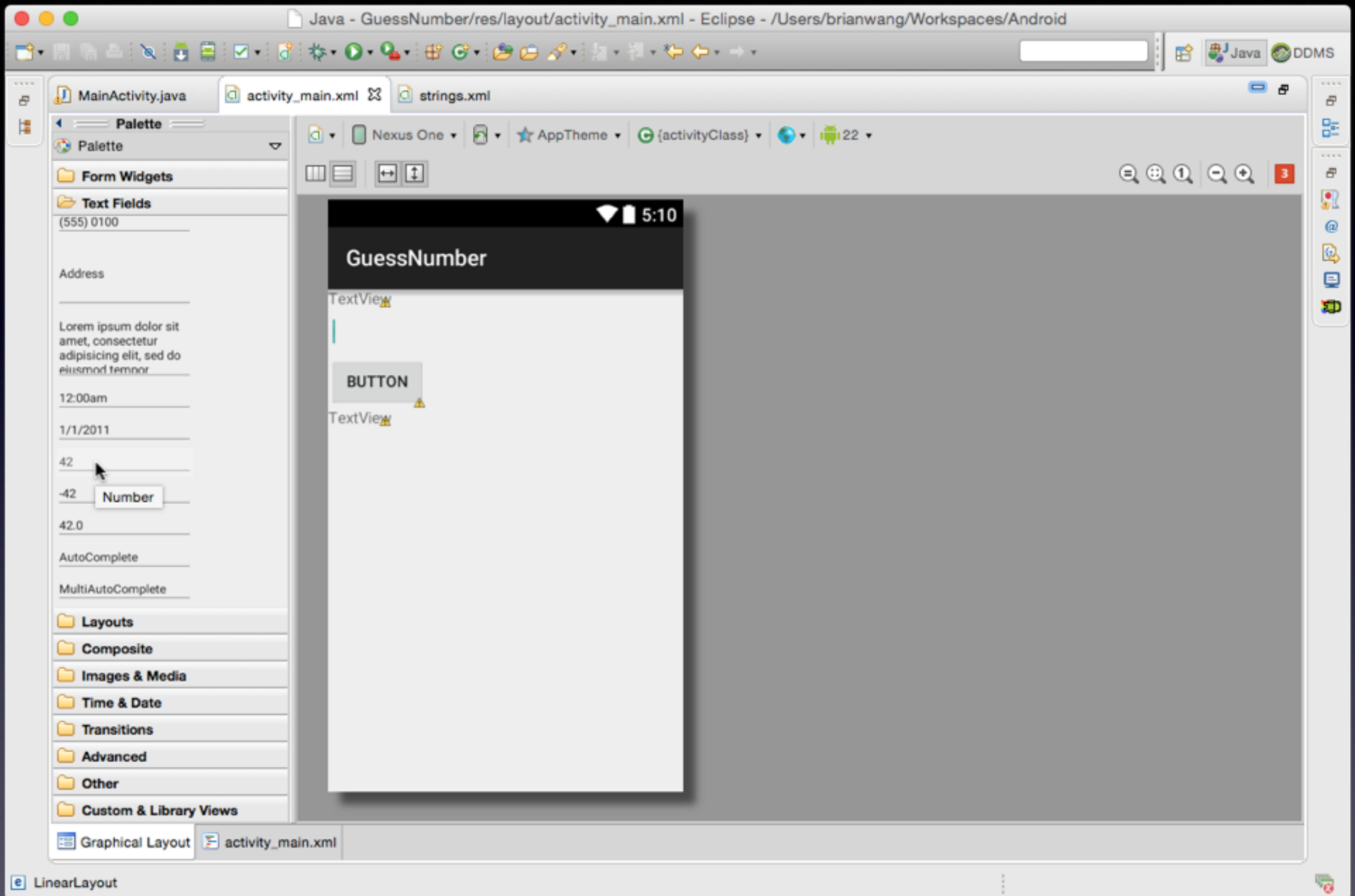
Cancel

Finish

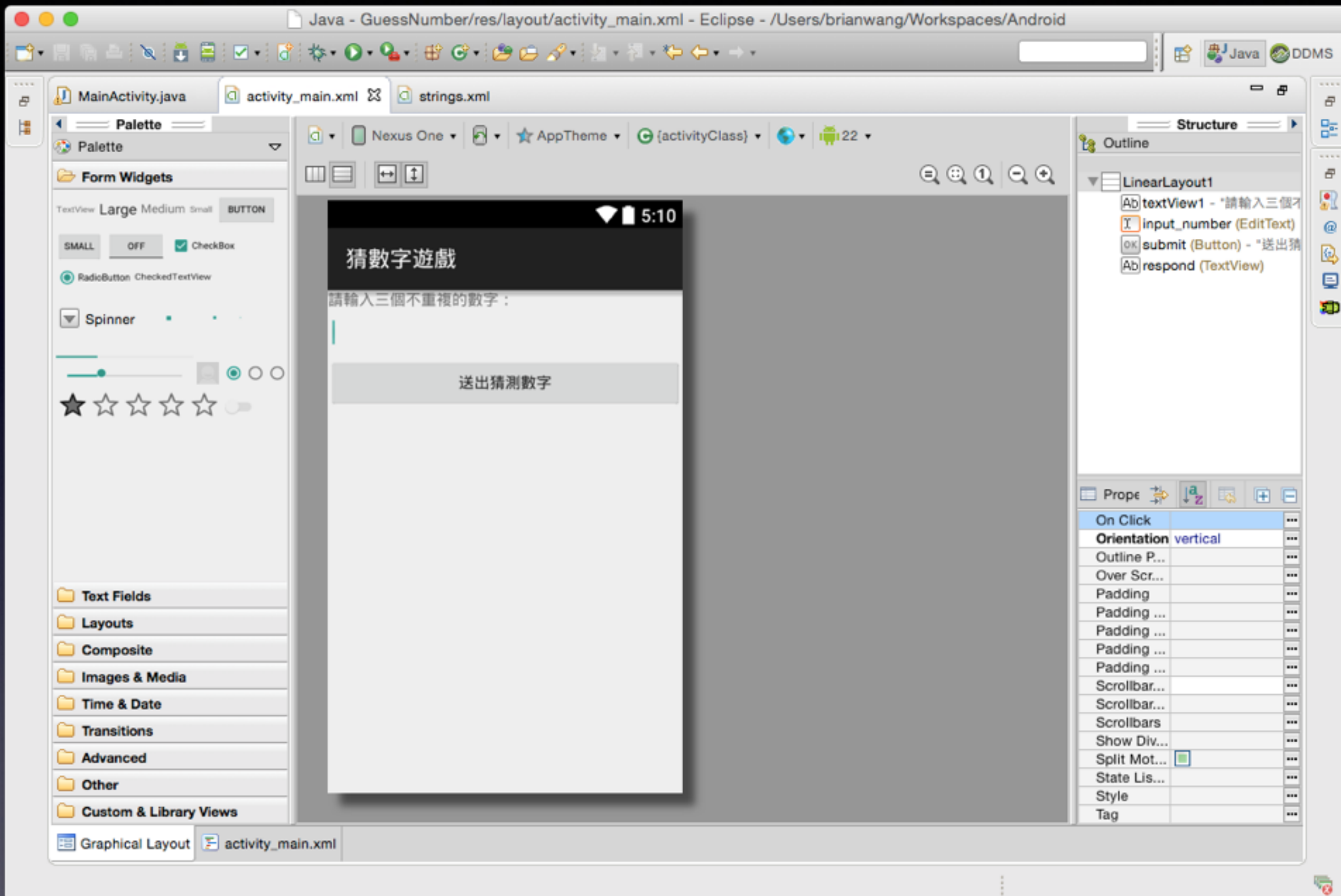


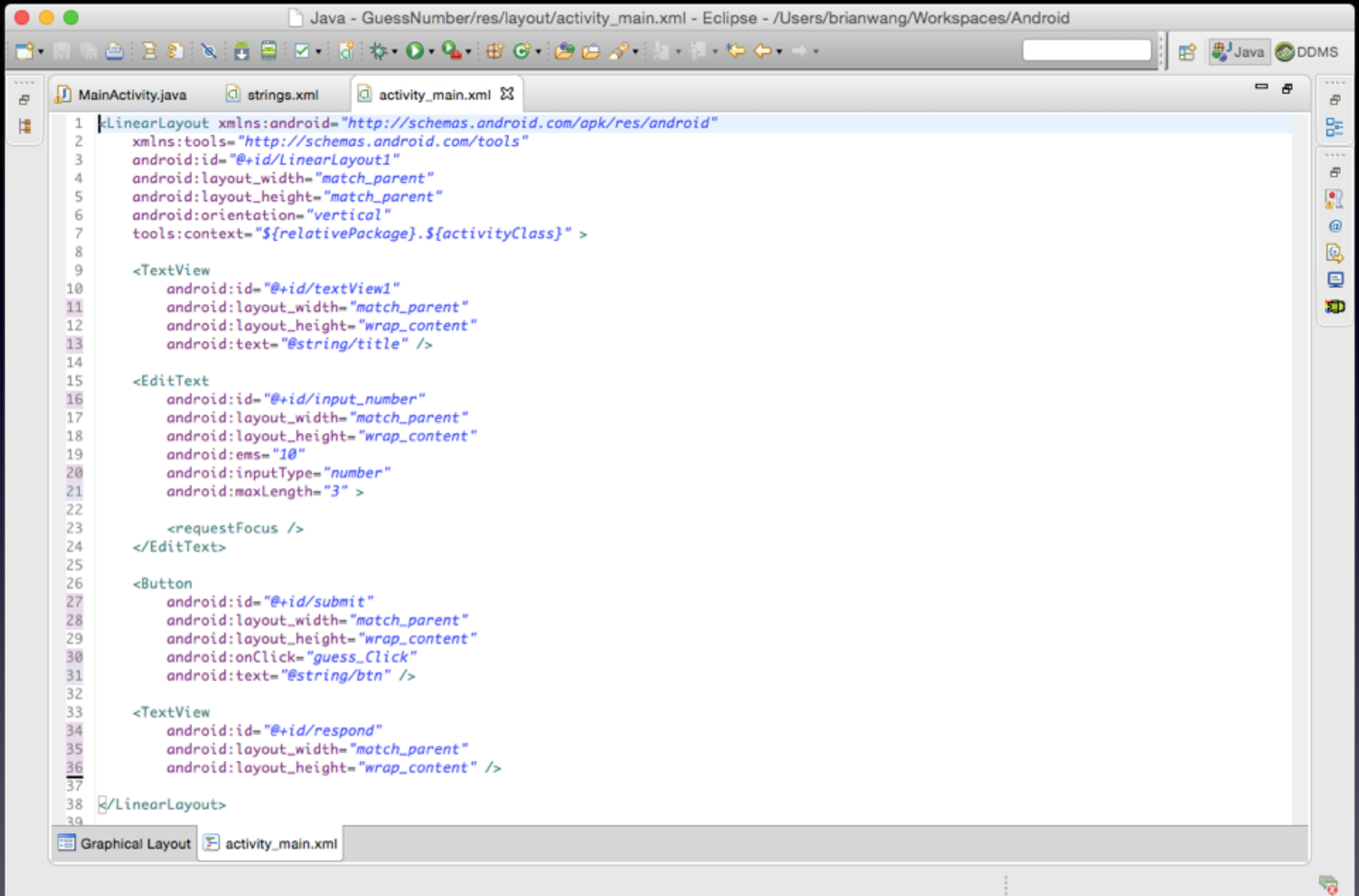
建立元件的版面配置

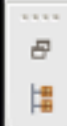
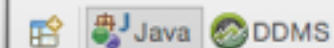
- 刪除 “Hello World” TextView
- 將 RelativeLayout 改為 LinearLayout (Vertical)
- 在 LinearLayout 垂直編排 TextViwe、
EditText (Number) 、Button 和 TextViwe 元件



- 修改 string.xml 的 app_name 為 “猜數字遊戲”
- 第一個 TextView 將 Layout Width 改為 Match Parent，並新增 title 的 string 值為 “請輸入三個不重複的數字：”
- 將 EditText 的 id 設定為 input_number，並設定 maxLength 為 3
- 將 Button 的 Layout Width 改為 Match Parent，id 設定為 submit，onClick 設定為 guess_Click，並新增 btn 的 string 值為 “送出猜測數字”
- 第二個 TextView 將 Layout Width 改為 Match Parent，id 設定為 respond，並清空 text







MainActivity.java

strings.xml



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <resources>
3
4     <string name="app_name">猜數字遊戲</string>
5
6     <string name="title">請輸入三個不重複的數字:</string>
7
8     <string name="btn">送出猜測數字</string>
9
10    <string name="error">輸入錯誤，請輸入三個不重複的數字</string>
11
12 </resources>
13
```



Resources

strings.xml

建立 Activity 活動類別

- 宣告成員變數
- onCreate() 方法
- getRandomNumber() 方法
- showMsg() 方法
- guess_Click() 事件處理方法

宣告成員變數

```
14 public class MainActivity extends Activity {  
15  
16     private final static int NUMBER_LENGTH = 3;  
17  
18     private EditText inputText;  
19     private TextView respond;  
20  
21     private List<String> numList;  
22     private String result;  
23 }
```

onCreate() 方法

```
24 @Override
25 protected void onCreate(Bundle savedInstanceState) {
26     super.onCreate(savedInstanceState);
27     setContentView(R.layout.activity_main);
28
29     inputText = (EditText) findViewById(R.id.input_number);
30     respond = (TextView) findViewById(R.id.respond);
31
32     getRandomNumber();
33 }
34
```

getRandomNumber() 方法

```
79 private void getRandomNumber() {  
80     List<String> list = new ArrayList<String>();  
81  
82     for (int i = 0; i < 10; i++) {  
83         list.add(String.valueOf(i));  
84     }  
85  
86     numList = new ArrayList<String>();  
87  
88     for (int i = 0; i < NUMBER_LENGTH; i++) {  
89         int index = (int) (Math.random() * 10);  
90         numList.add(list.get(index));  
91         result += list.get(index);  
92         list.remove(index);  
93     }  
94 }  
95
```


showMsg() 方法

```
96 private void showMsg(String msg, boolean isClear) {  
97     respond.setText(msg);  
98  
99     if (isClear) {  
100         inputText.setText("");  
101     }  
102 }
```

guess_Click() 事件處理方法

```
35 public void guess_Click(View view) {  
36     String inputNumber = inputText.getText().toString().trim();  
37  
38     if (inputNumber.length() == 0) {  
39         showMsg(this.getString(R.string.error), true);  
40         return;  
41     }  
42  
43     List<String> guessNumberList = new ArrayList<String>();  
44     Set<String> set = new HashSet<String>();  
45  
46     for (int i = 0; i < inputNumber.length(); i++) {  
47         String s = inputNumber.substring(i, i + 1);  
48         guessNumberList.add(s);  
49         set.add(s);  
50     }  
51 }
```

```
52     if (set.size() != 3) {
53         showMsg(this.getString(R.string.error), true);
54         return;
55     }
56
57     int a = 0;
58     int b = 0;
59
60     for (int i = 0; i < guessNumberList.size(); i++) {
61         int index = numList.indexOf(guessNumberList.get(i));
62
63         if (index == -1) {
64             continue;
65         } else if (index == i) {
66             a++;
67         } else {
68             b++;
69         }
70     }
71
72     if (a == NUMBER_LENGTH) {
73         showMsg("INPUT的數字：" + inputNumber + "恭喜你!答對囉~答案：" + result, false);
74     } else {
75         showMsg("INPUT的數字：" + inputNumber + "=" + a + "A" + b + "B", true);
76     }
77 }
78
```