**Lab 2**

**畫面設計與元件使用**

本節目的：

* 了解Android的畫面設計方式。
* 了解Android中主要的三種Layout布局。
* 了解如何使用Android布局元件。
  1. **版面配置**

一個基本的APP至少會具備一個畫面來與使用者互動，在Android中我們將透過「Xml」語法去描述一個畫面的版面布局，這類檔案我們稱之為「Layout」。

圖2-1左下方的「Design/Text」中，Design是設計畫面的圖形介面，Text代表相對應的程式畫面的Xml。

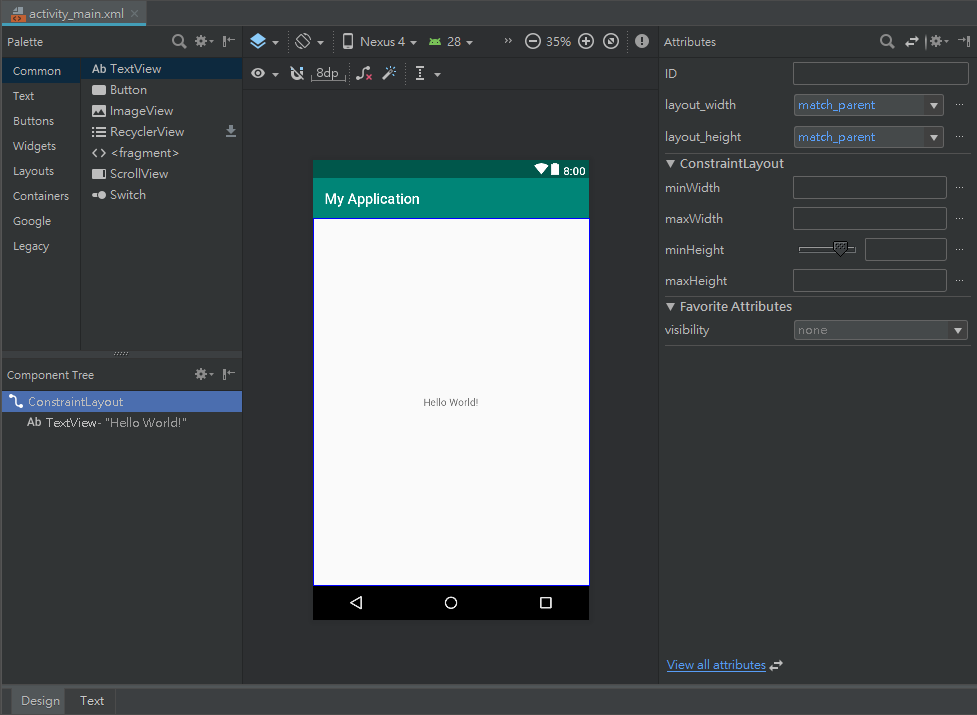


圖2-1 Android Studio布局畫面

* + 1. **設計畫面介紹**

Layout檔會被放置在「res/layout」目錄之下，將其打開後，可見到圖2-2中的預覽畫面，左側為調色盤/元件盤（palette），我們可以從中挑選Layout或是元件直接放到中間的預覽畫面中或是圖2-4的元件樹中。

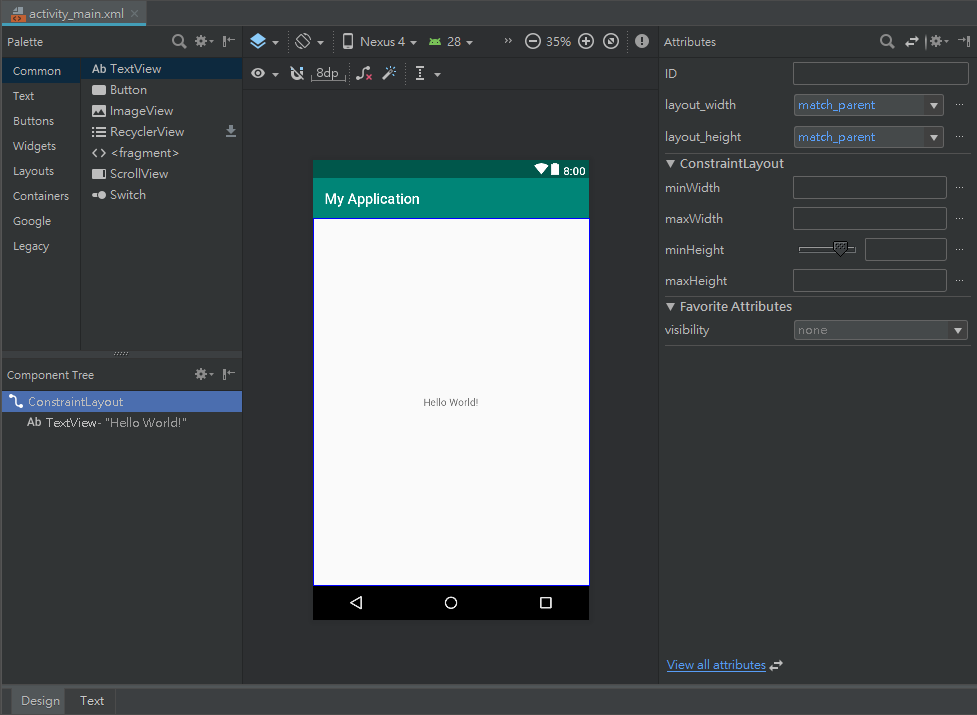


圖2-2 元件盤位於布局畫面左上方

已經在預覽畫面中的元件可直接點擊，圖2-3右方的欄位會顯示所點擊到的元件的屬性表，可以透過屬性表直接更改該元件的相關資訊內容。

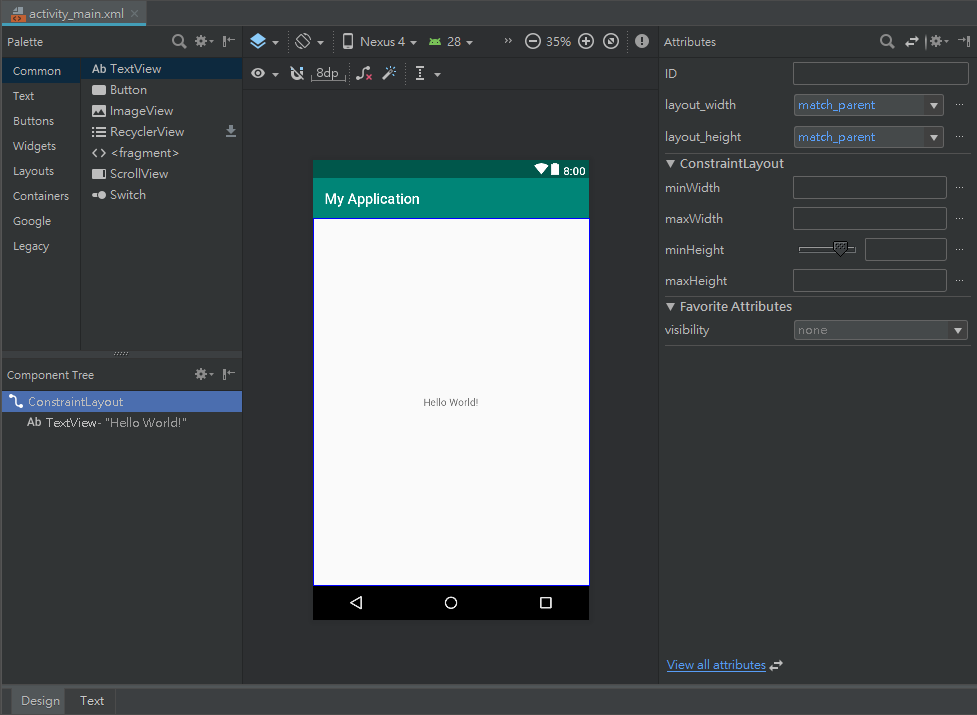


圖2-3 元件屬性表位於布局畫面右上方

圖2-4左下角會顯示畫面對應的元件樹（Component Tree），元件樹可以讓我們知道元件之間的定位關係，透過樹狀關係來描述其位置。

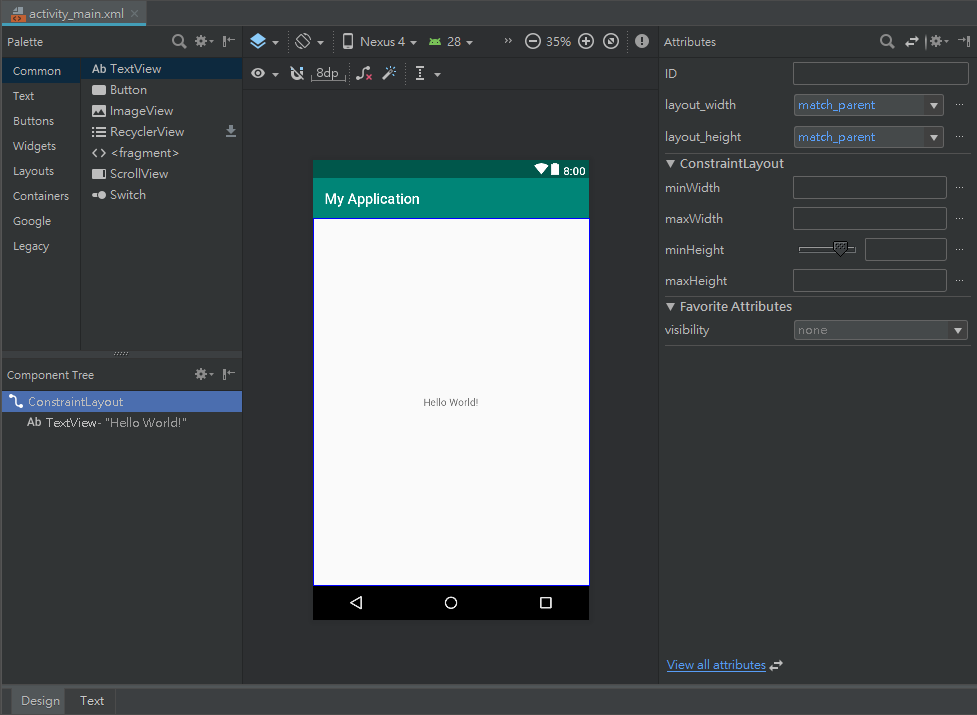


圖2-4 元件樹位於元件盤的下方

元件盤的元件可直接拖曳到預覽畫面或元件樹中，而當畫面的設計較複雜時則建議直接拖曳至元件樹中，元件樹可以明確的表示元件層級位置，要避免直接拖到預覽畫面，造成元件的層級位置擺放錯誤，細節的層級關係會在後面做介紹。

說明

由於Layout檔是「Xml」格式，可以透過圖2-5左下角的「Text」按鈕切換查看Xml程式碼，「Design」按鈕切換查看設計畫面的圖形介面。

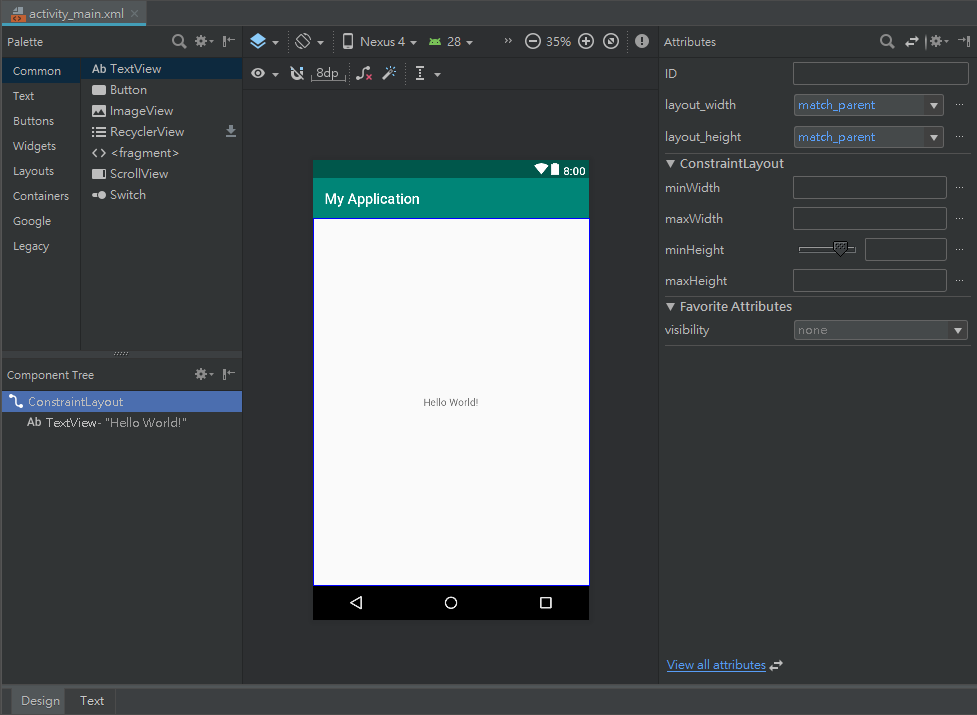
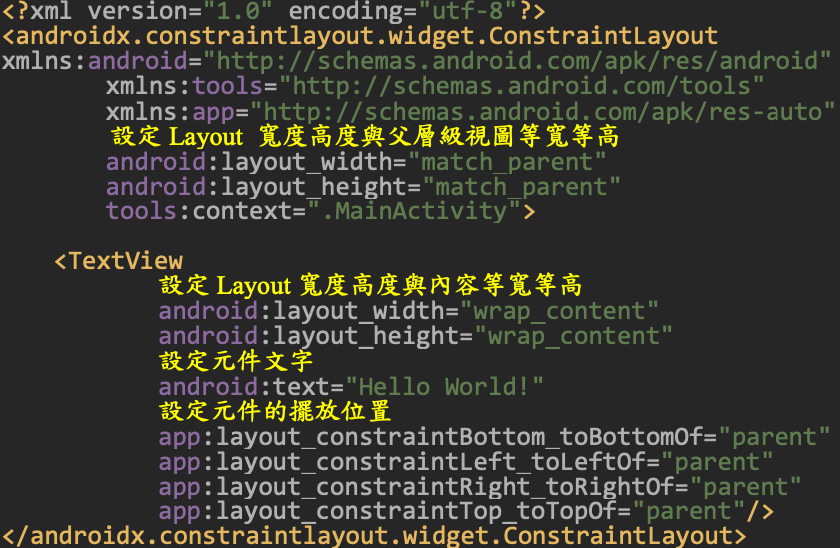


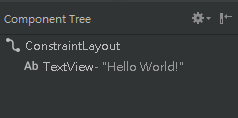
圖2-5 按鈕切換布局模式

切換至「Text」後，我們可以看到其原始碼格式。



* + 1. **Layout－版面布局**

在畫面編排時最重要的是決定每個元件在畫面上的位置。若要使元件被layout控制，我們要把元件放到layout內，形成父子層級的關係，如圖2-6所示。



影響排列方式

父層級

子層級

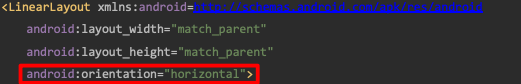
圖2-6 元件的層級排序

子層級的元件排列方式就會被父層級的layout所影響，當然子層級也可以擺放新的layout，這樣「孫」層級就會同時受到子層級與父層級影響，而基本的layout有四種，LinearLayout、FrameLayout、RelativeLayout、ConstraintLayout，RelativeLayout已被ConstraintLayout取代，因此以下介紹三種layout方式。

* LinearLayout：顧名思義就是依照順序逐一排列介面元件，也是最常被使用的布局方式。依照方向（orientation）的不同LinearLayout分為垂直排列（vertical）與水平排列（horizontal）兩種呈現方式，透過orientation屬性切換，如圖2-7所示。

|  |  |
| --- | --- |
|  |  |

圖2-7 垂直布局vertical（左）與水平布局horizontal（右）



Orientation可以決定LinearLayout是vertical或horizontal，如果沒有此屬性預設是vertical。

而編排布局上除了三種布局方式之外，每個畫面元件的屬性中也可以用layout:width與layout:height設定本身的寬與高。



width與height可以寫一個固定數值強制定義大小，由於裝置大小不同，建議要能以裝置尺寸做動態調整。

* + match\_parent：元件的寬度與高度擴展到最大，但最大只能等同父層級的大小，如圖2-8所示。

|  |  |
| --- | --- |
|  |  |

圖2-8 最大化按鈕顯示

* + wrap\_content：元件的寬度與高度依據內容自動調整，而內容定義包含文字、圖片或子層級，如圖2-9所示。

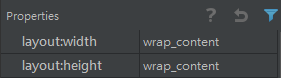


圖2-9 最小化按鈕顯示

如圖2-9所示，按鈕的大小是依照ButtonA的文字內容，自動調整成大小。

* FrameLayout：FrameLayout是以堆疊方式呈現，如果使用這種介面布局檔，子層級的元件皆會重疊，重疊順序會依照元件樹，下面的元件會覆蓋上面的元件。圖2-10中，ButtonB覆蓋於ButtonA上，可看到元件樹中ButtonB排列在ButtonA下。

|  |  |
| --- | --- |
|  |  |

圖2-10 按鈕B覆蓋於按鈕A上

* ConstraintLayout：ConstraintLayout是Android Studio 2.2中新加入的布局，結合了FrameLayout與RelativeLayout的特性，可以有效地解決布局層級過多的問題。ConstraintLayout採用堆疊的方式呈現，定位上需要明確描述參考的對象，以及與該對象的具體距離單位量，否則元件會以畫面左上角作為基準。與傳統Layout不同，ConstraintLayout非常適合使用圖形化的方式來編輯界面。

Step1 設定邊框範圍，上下左右的padding各為16dp，如圖2-11所示，實線框是ConstraintLayout的尺寸，上方程式碼設定ConstraintLayout中內容到ConstraintLayout邊緣的距離（實線到虛線的距離為16dp），虛線框的範圍為可放置元件的區域。

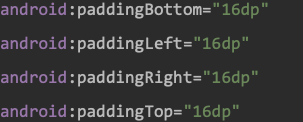




圖2-11 ConstraintLayout布局

**16dp**

Step2 點擊ButtonA後可以看見元件四周圍的基準點，透過滑鼠拖曳基準點，可讓元件對齊畫面邊緣或是其它元件，設定ButtonA對齊畫面上緣與左側，如圖2-12所示。



圖2-12 拖曳按鈕A對齊畫面

**100dp**

**200dp**

左圖2-12的實線框代表ButtonA的位置，設定靠左面和靠上面，上方為程式碼設定。

Step3 將ButtonB的基準點對齊ButtonA的右側與下緣，並設定距離16dp，如圖2-13所示。

圖2-13的實線框代表ButtonB的位置，下方為程式碼設定：

圖2-13 拖曳按鈕B對齊按鈕A

* marginTop設定距離上緣基準16dp。
* marginStart設定距離左側基準16dp。
* ButtonB的**上緣對齊**ButtonA的下緣。

**16dp**

* ButtonB的**左側對齊**ButtonA的右側。



* + 1. **Widget－視窗元件**

了解如何編排元件的位置後，下一步要了解元件盤中的元件（Widget），元件可於圖2-14左側的原件盤中選擇。

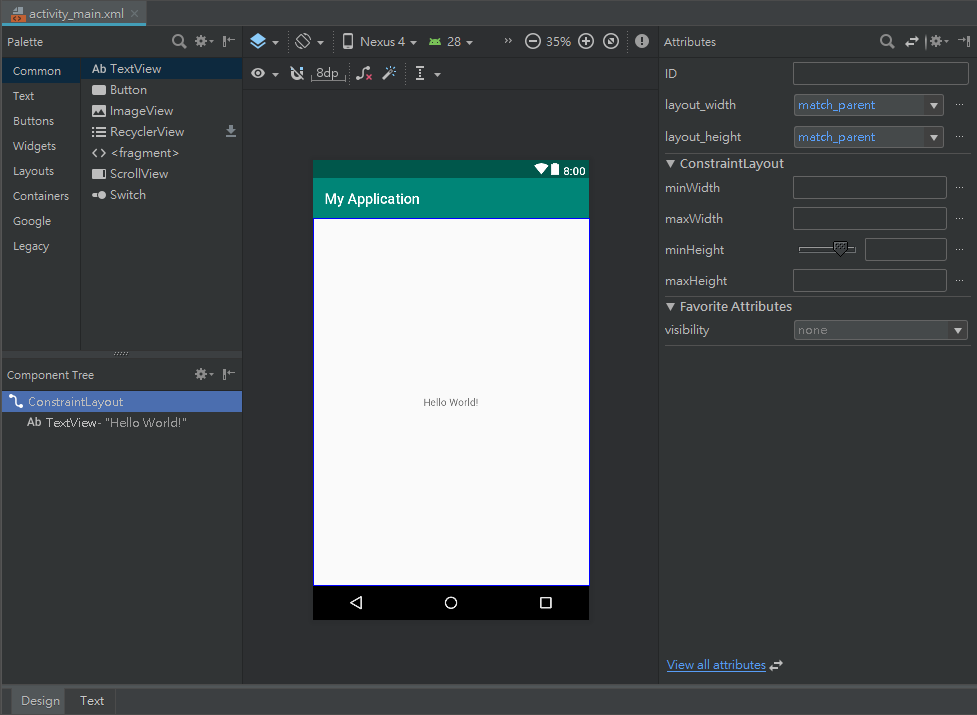


圖2-14 元件盤

下列簡單介紹四種常用的元件：

* TextView：顯示文字的文字元件，如圖2-15所示。

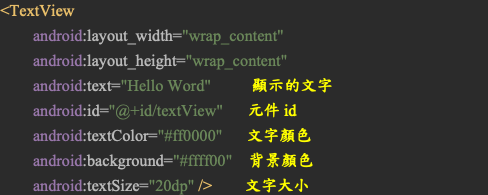




圖2-15 TextView範例

* Button：觸發點擊事件的按鈕元件，自帶有額外有按下的動畫，如圖2-16所示。





圖2-16 Button範例

* EditText：可輸入訊息的輸入元件。當EditText被點擊後會自動彈出小鍵盤讓使用者對「android:text」做輸入，如圖2-17所示。





圖2-17 EditText範例

* RadioGroup與RadioButton：單選框的群體與單選元件。一個RadioGroup中可放置多個RadioButton，如圖2-18所示。

checked屬性決定該RadioButton為開啟或是關閉狀態（預設為false）。若透過使用者去點擊其中一個RadioButton並將其變為true（開啟），而其他的RadioButton會被應用程式給轉為false（關閉）。





圖2-18 RadioGroup範例

* 1. **猜拳遊戲畫面設計**
* 使用ConstraintLayout實作圖2-19的畫面布置。

|  |  |
| --- | --- |
|  |  |

圖2-19 猜拳遊戲預覽畫面（左）與布局元件樹（右）

觀察布局與畫面之間的關聯性，可以注意到元件的擺放是受到layout影響的。因此需要去理解layout的用法。

說明

第三層：由Linear layout垂直排序

* + 1. **元件布局與排版**

Step1 開啟activity\_main.xml檔，由於預設Layout會有一個「Hello World!」文字的TextView元件，此處需要我們手動將它刪除，如圖2-20所示。



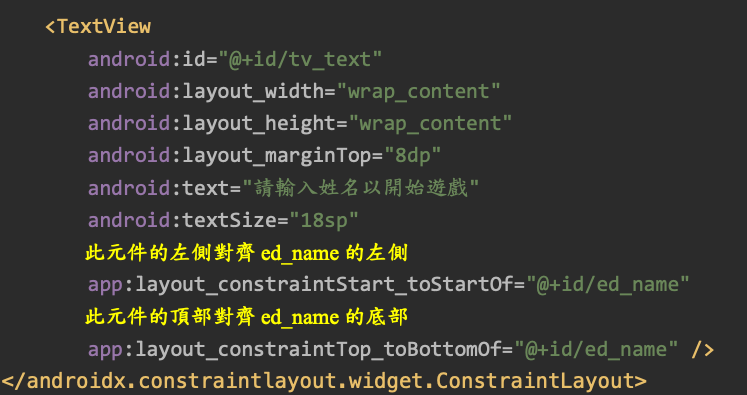
圖2-20 移除預設的TextView

Step2 在左上方搜尋框輸入「Text」，將TextView與EditText拉入Component Tree中，並將EditText對齊畫面上緣與左側，而TextView則對齊EditText左側與下緣，如圖2-21所示。



圖2-21 加入顯示文字與輸入框

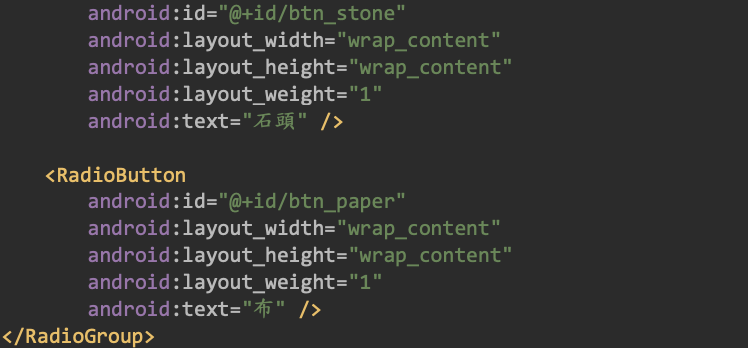
元件放置完成後，系統會產生對應的Xml，之後額外可修改id與增加屬性。



Step3 在TextView下方加入RadioGroup元件，並在RadioGroup中增加三個RadioButton，如圖2-22所示。



圖2-22 加入RadioButton



Step4 將Button與TextView放入RadioGroup的下方，如圖2-23所示。



圖2-23 放入猜拳鈕與狀態文字





Step5 完成後的畫面，如圖2-24所示。



圖2-24 猜拳遊戲實機畫面