

目錄

1-1 何謂變數.....	2
變數方塊	2
變數可記錄的類型	2
區域變數	3
1-2 對話框的運用	5
1-3 標準 BMI 計算的畫面佈置.....	7
1-4 標準 BMI 計算的程式拼貼.....	8
1-5 成績計算的畫面佈置.....	9
1-6 成績計算的程式拼貼.....	10
1-7 清除資料的程式拼貼.....	12
2-1 比較運算的說明	14
數字類	14
文字類	14
邏輯類	15
2-2 及格與否的單向判斷式語法	16
2-3 多向判斷式的語法說明	18
想一想	18
2-4 BMI 多向判斷式語法	20
2-5 只能輸入 0-100 的成績單(一).....	23
2-6 只能輸入 0-101 的成績單(二).....	24
2-7 終級密碼畫面佈置.....	27
2-8 終級密碼的程式拼貼.....	28

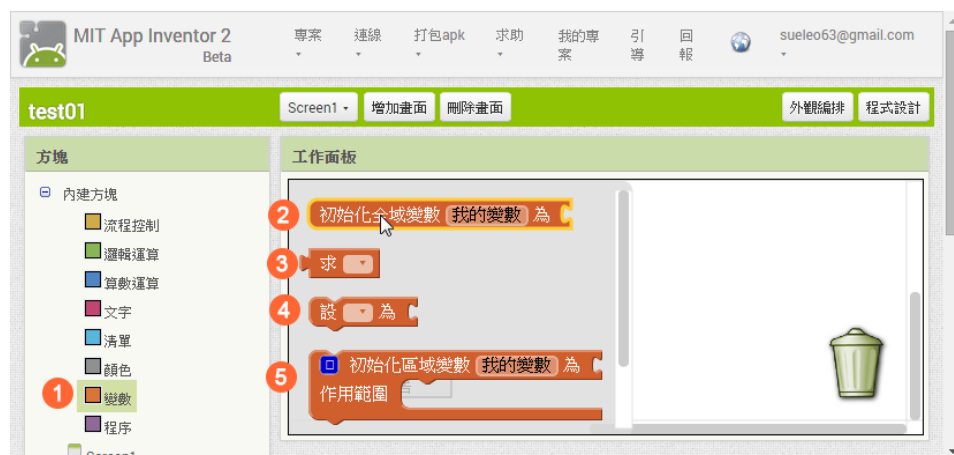
1-1 何謂變數

在程式中有些資料會隨著程式的運算產生變化，爲了要記錄這些會經常變化的資料，我們會設定一個代號來記錄這些會變化的資料，而這個代號就稱爲變數。

變數在程式中又分爲「全域變數」和「區域變數」二類。

變數方塊

1. 在內鍵方塊中點選「變數」
2. 設定「初始化全域變數」
3. 取得變數值
4. 指定變數值
5. 設定「初始化區域變數」



變數可記錄的類型

1. 宣告「score」爲數字類型的全域變數,起始值爲「200」
2. 宣告「name」爲文字類型的全域變數,起始值爲「leo」
3. 宣告「flag」爲邏輯類型的全域變數,起始值爲「true」



區域變數

1. 設定第一個區域變數的名稱
2. 點選擴充項目的圖示
3. 可以增加區域變數



1. 利用拖曳法增加區域變數
2. 完成所需要的區域變數設定



1-2 對話框的運用

有些元件一開始看不到，例如對話方塊

1. 拖曳對話方塊到手機中
2. 會在下方有個對話方塊



完成手機元件上的佈置：



先設定二個「全域變數」

初始化全域變數 name 為 " " (empty string)

初始化全域變數 yer 為 " " (empty string)

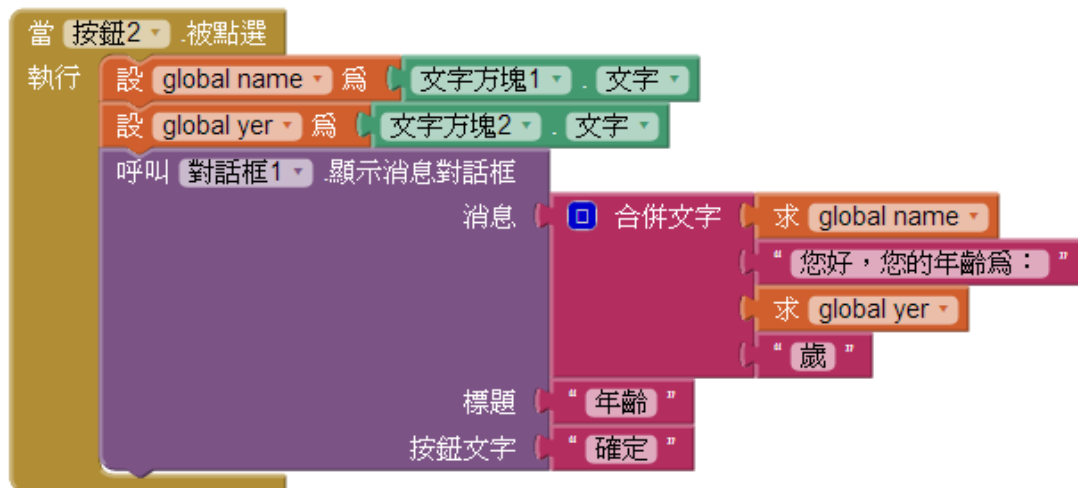
「name」是爲了記錄「文字方塊 1 的文字」

「yer」是爲了記錄「文字方塊 2 的文字」

完成第一個按鈕的語法，主要是使用「短暫訊息」



完成第二個按鈕的語法，主要是使用「對話方塊」



點第一個按鈕可以看到訊息視窗



點第二個按鈕看到對話框



1-3 標準 BMI 計算的畫面佈置

身體質量指數(BMI)是一種以身高為基礎來測量體重是否符合標準的指數。對全身的體脂肪而言，BMI 是值得信賴的指數，它與疾病及死亡的風險具關連性。

【BMI 測試】

世界衛生組織建議以身體質量指數 (Body Mass Index, BMI) 來衡量肥胖程度，其計算公式是以體重 (公斤) 除以身高 (公尺) 的平方。

國民健康署建議我國成人BMI應維持在18.5 (kg/m²) 及24 (kg/m²) 之間，太瘦、過重或太胖皆有礙健康。

研究顯示，體重過重或是肥胖 (BMI ≥ 24) 為糖尿病、心血管疾病、惡性腫瘤等慢性疾病的主要風險因素；而過瘦的健康問題，則會有營養不良、骨質疏鬆、猝死等健康問題。



成人肥胖定義	身體質量指數(BMI) (kg/m ²)	腰圍 (cm)
體重過輕	BMI < 18.5	
健康體位	18.5 ≤ BMI < 24	
體位異常	過重：24 ≤ BMI < 27 輕度肥胖：27 ≤ BMI < 30 中度肥胖：30 ≤ BMI < 35 重度肥胖：BMI ≥ 35	男性：≥ 90 公分 女性：≥ 80 公分

$$\text{BMI} = \frac{\text{體重(公斤)}}{\text{身高}^2(\text{公尺}^2)}$$

以經知道 BMI 的公式，我們就可以利用 AppInventor 來寫一個 BMI 計算的程式，先完成下圖的畫面配置：



1-4 標準 BMI 計算的程式拼貼

接下來開始拼貼 BMI 的程式：

先設定三個「全域變數」

初始化全域變數 height 為 0

初始化全域變數 weight 為 0

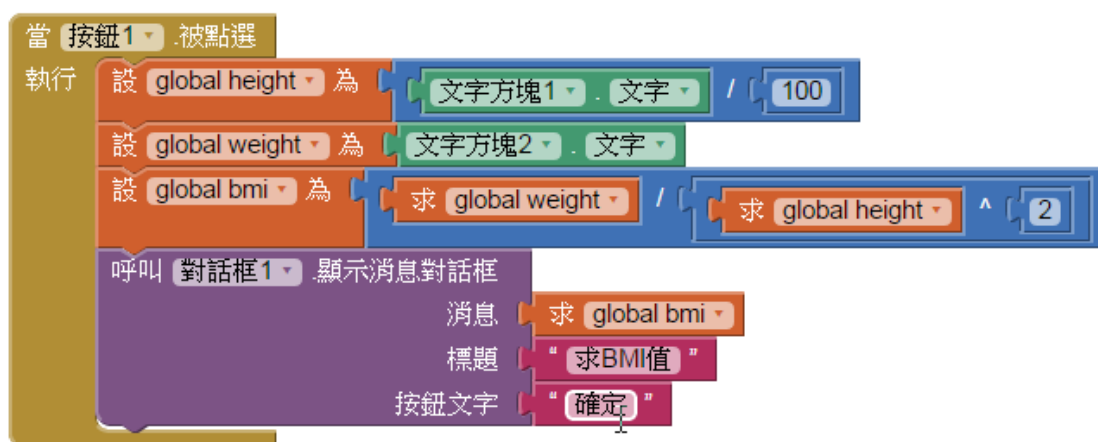
初始化全域變數 bmi 為 0

「height」是爲了記錄輸入的身高數值

「weight」是爲了記錄輸入的體重數值

「bmi」是爲了記錄計算的結果

當按下按鈕 1 時就要出現計算的結果，語法如下：



執行結果如下：



1-5 成績計算的畫面佈置

這個案例要利用 AppInventor 來寫一個成績計算的程式，先完成下圖的畫面配置：



1-6 成績計算的程式拼貼

接下來開始拼貼成績計算的程式：

先設定五個「全域變數」



「chin」是爲了記錄國文的分數

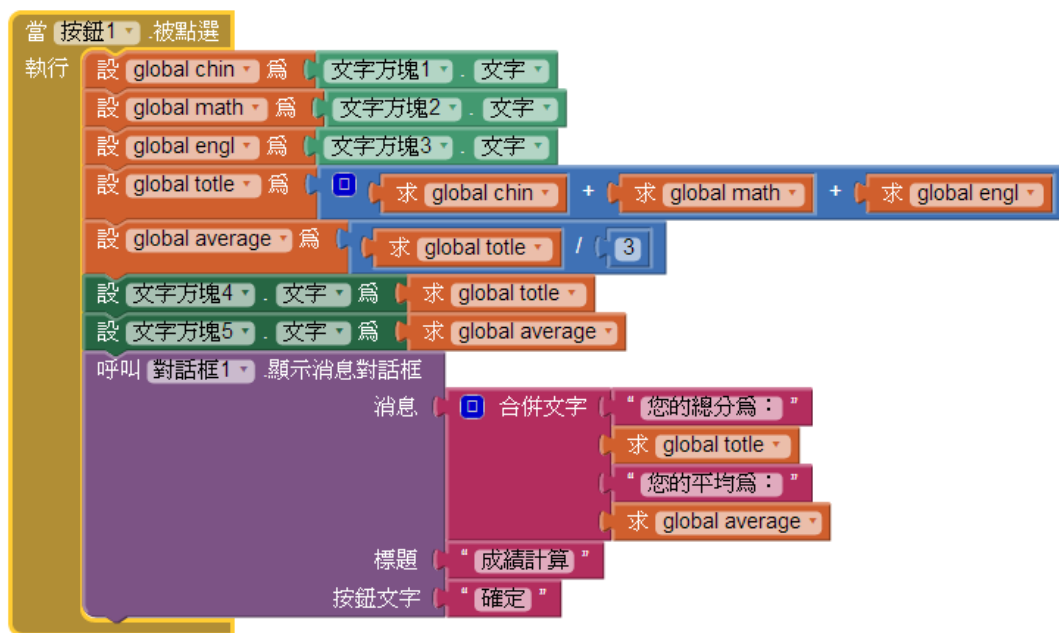
「math」是爲了記錄數學的分數

「engl」是爲了記錄英文的分數

「totle」是爲了記錄加總的計算結果

「average」是爲了記錄加總的計算結果

當按下按鈕 1 時就要出現計算的結果，語法如下：

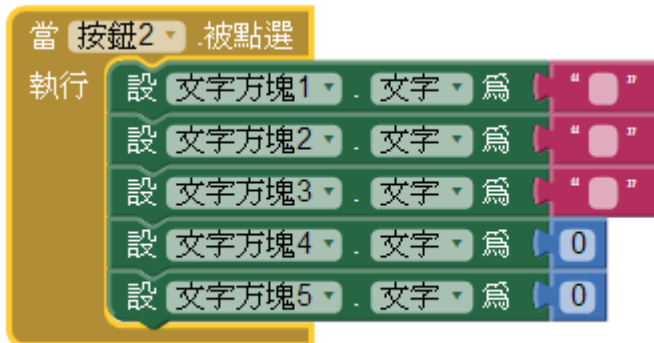


執行結果如下：



1-7 清除資料的程式拼貼

最後完成「清除資料」的按鈕語法，這個按鈕就是要讓所有的文字方塊都回到預設值，所以語法也變的很簡單，如下圖：



執行的結果，當文字方塊原本是有資料的，按下「清除資料」的按鈕



資料就會清空回到預設值了。



2-1 比較運算的說明

之後的程式都需要使用到條件判斷，而在使用條件判斷之前，一定要了解比較運算式，因為條件判斷是否成立，就是比較運算的工作，而比較運算有三大類：

數字類

數字類的比較運算主要是以數字的大小來做比較，可以由內鍵方塊「算數運算」找到「比較方塊」



文字類

文字類的比較運算主要是以文字是否相同來做比較，可以由內鍵方塊「文字」找到「比較文字方塊」



邏輯類

這在同時要判斷多個條件時經常用到，可以由內鍵方塊「邏輯運算」中找到許多功能的邏輯方塊



2-2 及格與否的單向判斷式語法

首先說明條件判斷的語法：

當符合條件要執行我們指定的程式

3. 加入條件

4. 符合條件要執行的程式



一開始先建立格與否的畫面設計：



我們希望使用者在文字方塊 1 中輸入自己的分數，當按下及格判斷按鈕時，如果及格標籤 2 就會顯示「你及格了」，如果不及格就不做任何反應，因為只有及格才做反應，所以稱為「單向判斷」的語法。切換到程式設計畫面，在按鈕 1 被點選時要執行條件判斷，所以在內鍵方塊的「流程控制」中找到「如果」的條件判斷方塊。



1. 加入條件為「文字方塊的文字大於等於 60」
2. 如果條件成立就執行「設定標籤 2 的文字為〔你及格了〕」



執行的結果：

如果輸入 50，則條件不成立，所以不做任何的反應



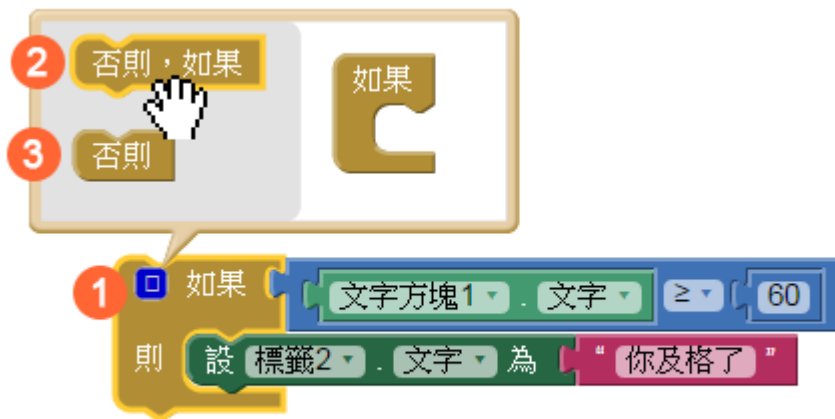
如果輸入 60，則條件成立，所以設定標籤 2 的文字變成「你及格了」



2-3 多向判斷式的語法說明

多向判斷式代表當條件成立時可以執行某個語法，而不成立時可以執行另一個語法，有不同的方向可以執行，而要設定多向判斷式的語法，使用的還是「如果」這個判斷式，只是要再增加方塊：

1. 點選「如果」語法的藍色按鈕
2. 加入「否則，如果」可以再做第二個條件判斷
3. 加入「否則」代表當條件不成立就要執行「否則」的語法



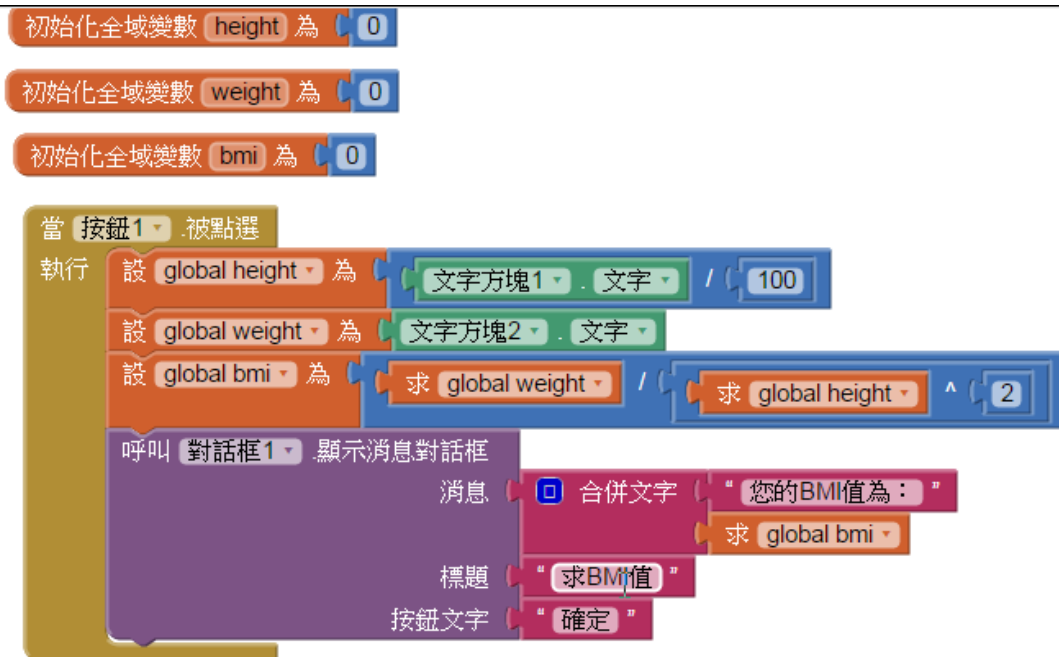
所以可以將剛剛的及格判斷做修改：

3. 加入條件為「文字方塊的文字大於等於 60」
4. 如果條件成立就執行「設定標籤 2 的文字為〔你及格了〕」
5. 否則的試就執行「設定標籤 2 的文字為〔你不及格〕」



想一想

這是當初的 BMI 計算的程式語法：



如果要加入條件判斷：

1. 當 BMI 大於等於 24，要出現「你過重了」
2. 當 BMI 小於 18.5，要出現「你過輕了」
3. 否則的話就顯示「標準」

該如何修改程式

2-4 BMI 多向判斷式語法

這是當初的 BMI 計算的程式語法：

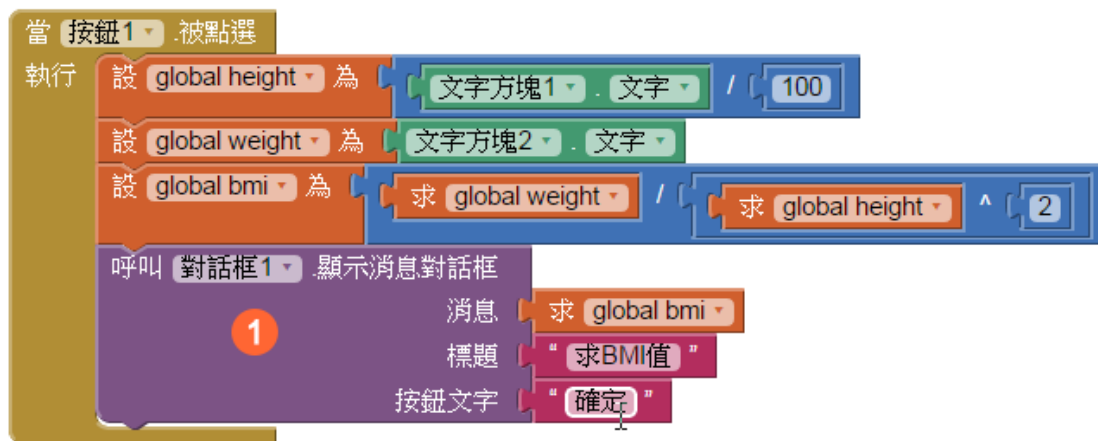


如果要加入條件判斷：

4. 當 BMI 大於等於 24，要出現「你過重了」
5. 當 BMI 小於 18.5，要出現「你過輕了」
6. 否則的話就顯示「標準」

該如何修改程式

這個程式只要修改「呼叫對話框 1」的語法即可



將「呼叫對話框 1」的語法換成「如果」的判斷式，但要加入多向式的判斷：

1. 輸入條件比較 BMI 大於等於 24
2. 條件成立出現「你過重了」
3. 第一個條件不成立再做第二個條件判斷，輸入條件為比較 BMI 小於 18.5
4. 條件成立出現「你過輕了」
5. 如果前二個條件都不成立就直接顯示「標準」



修改好的語法如下：

1. 輸入條件比較 BMI 大於等於 24
2. 條件成立出現「你過重了」
3. 第一個條件不成立再做第二個條件判斷，輸入條件為比較 BMI 小於 18.5
4. 條件成立出現「你過輕了」
5. 如果前二個條件都不成立就直接顯示「標準」

當 按鈕1 被點選

執行



2-5 只能輸入 0-100 的成績單(一)

在之前的成績計算也沒有限制只能輸入 1-100 之間的分數，所以必須使用條件判斷為它設定只能輸入 0-100 之間的數字，而這次我們希望在輸入完分數馬上判斷是否有輸入錯誤，那麼就必須將判斷式寫在當「文字方塊失去焦點時」

這裡以輸入國文分數的文字方塊做個說明：



但這個寫法有個問題，就是最後一個輸入的分數，並不會被加入計算，因為當我們按下「成績計算」時它並沒有失去焦點，所以計算出來的結果並不正確，我們必須使用另一個方式來撰寫。



2-6 只能輸入 0-101 的成績單(二)

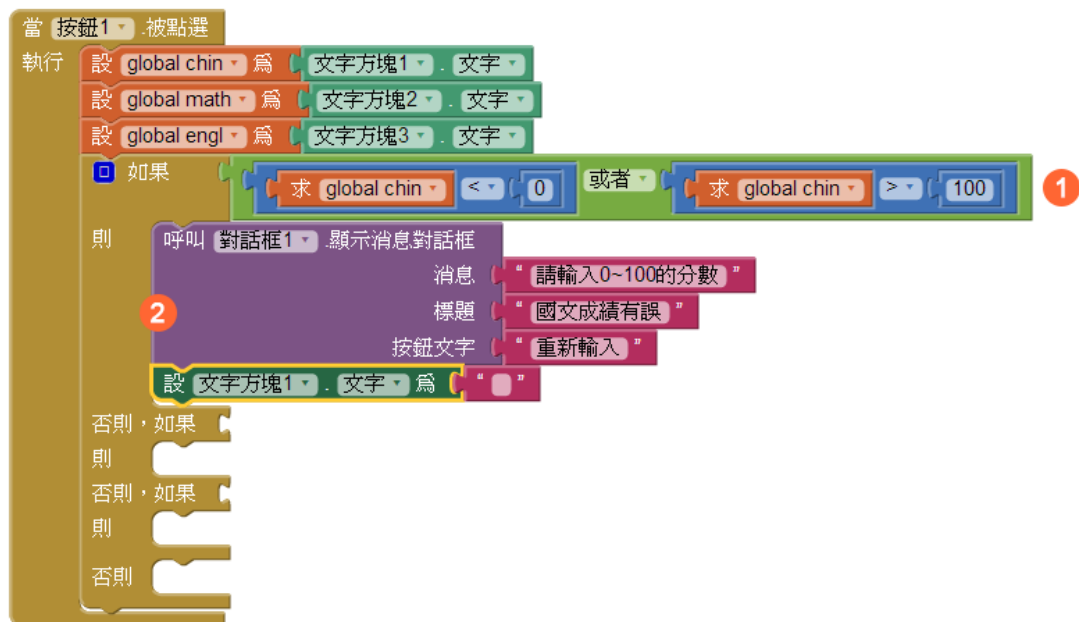
前一階段將判斷式寫在「當文字方塊失去焦點的時候」這種方式並不適合，所以我們還是將語法寫在按鈕上，只是要使用「多向判斷式」來撰寫語法：

1. 將三個文字方塊所寫的分數放到三個變數中
2. 先判斷國文的分數是否小於「0」或者大於「100」
3. 出現對話方塊，說明「國文分數有錯誤」，並且清除「文字方塊 1」的「文字」
4. 如果國文分數沒有問題，再判斷數學的分數是否小於「0」或者大於「100」
5. 出現對話方塊，說明「數學分數有錯誤」，並且清除「文字方塊 2」的「文字」
6. 如果國文和數學分數沒有問題，再判斷英文的分數是否小於「0」或者大於「100」
7. 出現對話方塊，說明「英文分數有錯誤」，並且清除「文字方塊 3」的「文字」
8. 如果國文、數學和英文的分數都沒有問題，就執行加總和平均的計算

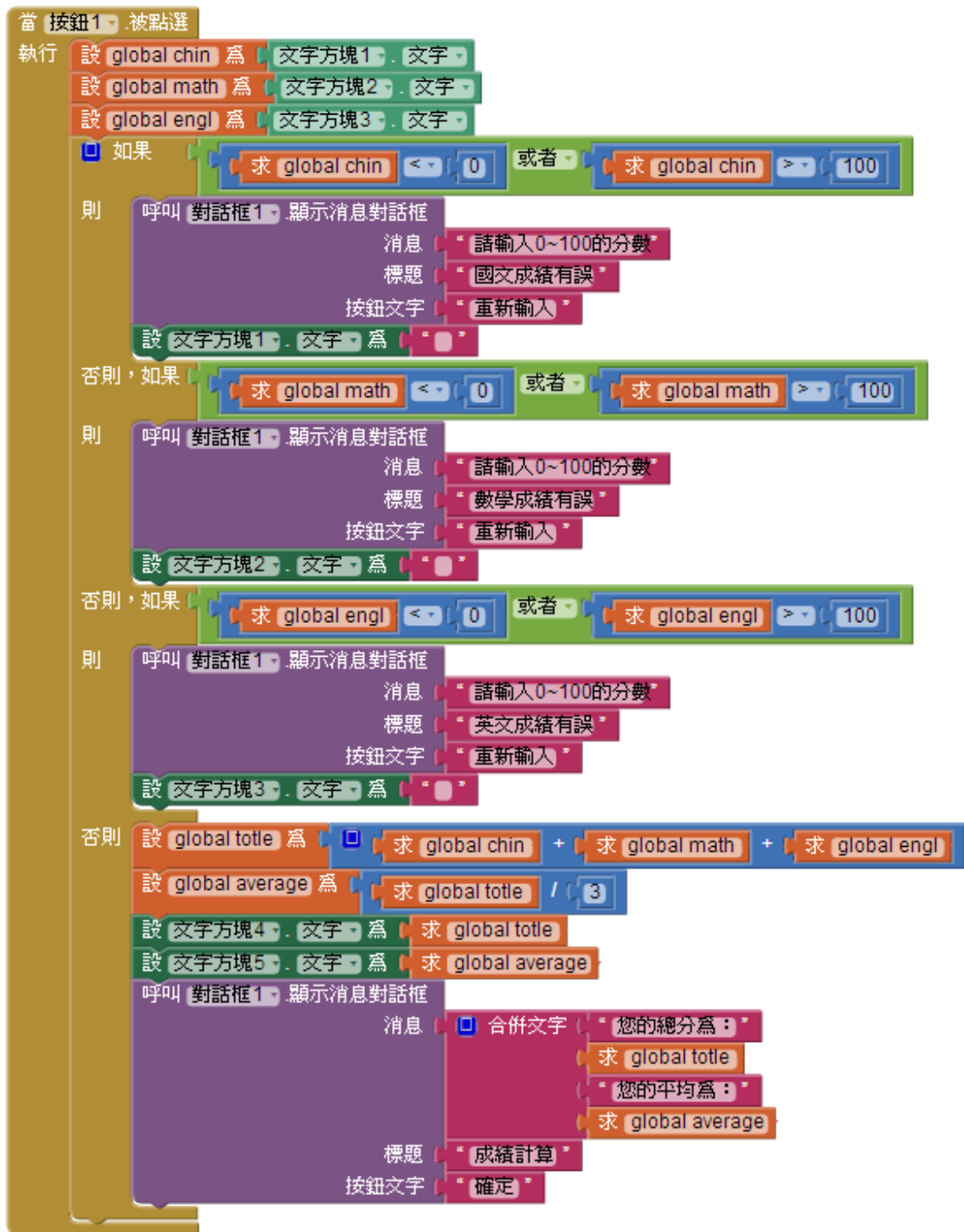


了解架構後，接下來寫入第一個條件判斷：

1. 就是判斷國文分數是否有小於「0」或者大於「100」
2. 如果有的話就呼叫對話方塊，並顯示想要的文字內容，最後將「文字方塊 1」的「文字」清空



完整的語法如下：



這就可以讓成績計算的 App 更加完整了

2-7 終級密碼畫面佈置

這次的畫面要有一部份是不顯示的，在某些特殊情形下，它才可以顯示，而不顯示的資料可以利用表格布局來設計，因為只要讓表格布局隱藏就可以了，所以終級密碼的佈置如下圖：

1. 勾選「顯示隱藏元件」，這樣設定隱藏的元件才會顯示
2. 以表格布局，放入設定終級密碼的元件，一開始就顯示
3. 以表格布局，放入猜終級密碼的元件，一開始就隱藏
4. 再玩一次的按鈕，一開始要隱藏，等猜對了才顯示



2-8 終級密碼的程式拼貼

接下來進行終級密碼的程式拼貼，首先要建立四個變數：

1. password：是記錄終級密碼的，預設值為「0」
2. guess：是記錄參與者所猜的密碼，預設值為「0」
3. min：是記錄目前密碼區間的最小值，預設值為「0」
4. max：是記錄目前密碼區間的最大值，預設值為「100」



接下來當「按鈕 1 被按下時」，代表已經設定了終級密碼，所以要顯示「表格布局 2」，並且將「表格布局 1」隱藏起來。

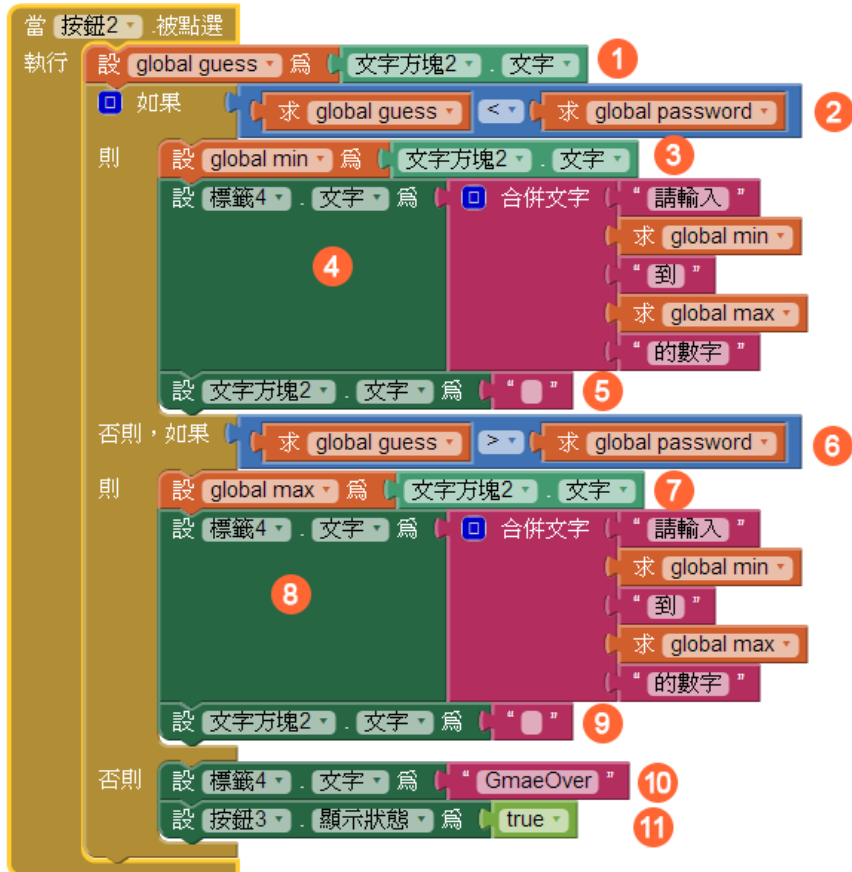
1. 將設定好的終級密碼記錄在「password」的變數中
2. 顯示「表格布局 2」
3. 隱藏「表格布局 1」



接下來就要開始猜密碼了，所以當「按鈕 2 被按下時」，就要由程式來判斷有沒有猜中：

1. 將玩遊戲所猜的密碼記錄到「guess」的變數中
2. 判斷「猜的密碼」有沒有小於「終級密碼」
3. 有的話，先將猜的密碼記錄給「min」
4. 設定標籤 4 要出現提示文字，例如：「請輸入 30 到 100 的數字」
5. 將剛剛的猜的數字清除掉
6. 如果剛剛第一個條件不成立，就要做第二個條件判斷，判斷「猜的密碼」有沒有大於「終級密碼」
7. 有的話，先將猜的密碼記錄給「max」
8. 設定標籤 4 要出現提示文字，例如：「請輸入 0 到 80 的數字」
9. 將剛剛的猜的數字清除掉
10. 如果小於不成立，大於也不成立，那麼就一定是猜中了，所以設定標籤 4 要出現「Game Over」文字

11. 讓「再玩一次」的按鈕出現



最後當按下再玩一次按鈕時，要讓所有的資料都回到預設值：

1. 設定顯示「表格布局 1」，並且將「表格布局 2」和「再玩一次」的按鈕隱藏起來
2. 設定四個變數都回到預設值
3. 設定「文字方塊 1」、「文字方塊 2」、「標籤 4」的文字都清空

