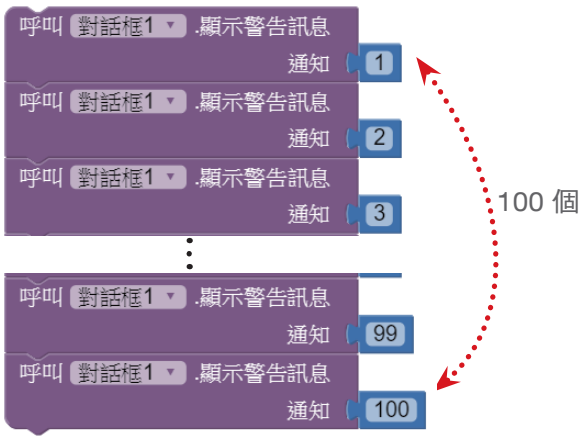
5 迴圈

學習重點

- 5-1 迴圈結構—使用「對於任意 數字」
- 5-2 迴圈結構—使用「當 滿足條件」

手機或平版的中央處理器（CPU）每秒鐘可執行幾億個指令，中央處理器適合用於運算大量的資料，也適合使用重複結構重複執行程式以獲得結果。迴圈結構是重複執行程式的結構，程式中設定迴圈結構的起始值、增減值與終止值。解決問題的過程中經常會使用迴圈結構，例如：搜尋、找最大值、找最小值、累加等，使用迴圈結構才能善用中央處理器的運算能力。假設寫一個程式能夠答數，產生列出 100 個視窗，上面有數字從 1 到 100，若不使用迴圈結構，需寫 100 個「  」後面串接數字 1 到 100，如左圖。

產生 100 個視窗上面寫著數字從 1 到 100 的程式碼



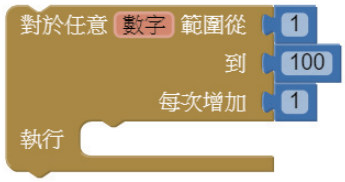
產生 100 個視窗上面寫著數字從 1 到 100 的程式碼

相同功能的程式使用迴圈結構可以簡化程式碼，如左圖。

App Inventor 的迴圈結構有「對於任意 數字」與「當 滿足條件」兩種，迴圈結構會給定迴圈變數初始值，再測試迴圈變數是否到達終止值，若測試迴圈變數未達終止值，則進行迴圈程式拼塊，迴圈程式拼塊執行結束後，迴圈變數進行遞增或遞減，會再次測試迴圈變數是否到達終止值，若未到終止值就繼續執行；超過終止值就跳出迴圈，執行迴圈的下一行程式。這兩種迴圈結構都在「內建方塊」的「流程控制」內。

5-1 迴圈結構－使用「對於任意 數字」

「對於任意 數字」迴圈結構用於已知重複執行次數的迴圈結構，迴圈結構中指定迴圈變數的初始值、終止值、遞增或遞減值，迴圈變數將由初始值變化到終止值，迴圈執行一次後，迴圈變數將依照遞增或遞減值變化。

「對於任意 數字」程式語法	程式範例 (印出 1 到 100)
	

說明

「對於任意 數字」內迴圈變數「數字」由起始值 (從) 變化到終止值 (到)，每執行一次迴圈程式拼塊，迴圈變數就會遞增 (減) 值 (每次增加)，重複執行迴圈內程式拼塊，直到超過終止值 (到) 後停止執行。本範例迴圈變數「數字」由 1 變化到 100，每次遞增 1，經由「對話框」物件的「顯示警告訊息」函式顯示「數字」到螢幕。

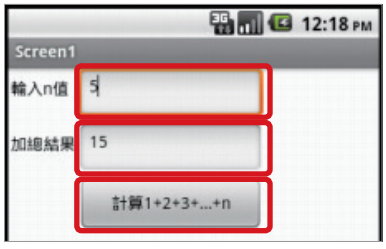
5-1-1 加總

檔案名稱：ch05\Sum.aia

請寫一個程式，給定一個正整數 n，求計算 1+2+3+…+(n-1)+n 的結果。

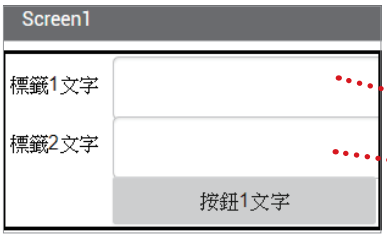
(一) 預覽執行結果

在 n 值欄位輸入「5」，點選「計算 1+2+3+…+n」，輸出加總結果為「15」。



(二) 建立使用者介面

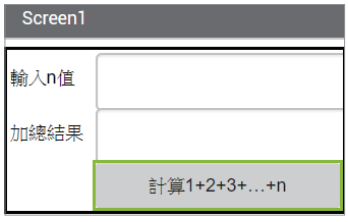
在 Screen1 新增一個表格配置，在表格配置元件內加入兩個標籤、兩個文字輸入盒與一個按鈕。

Screen1 的畫面	「元件清單」視窗中顯示所有元件間關係
	

（三）屬性設定

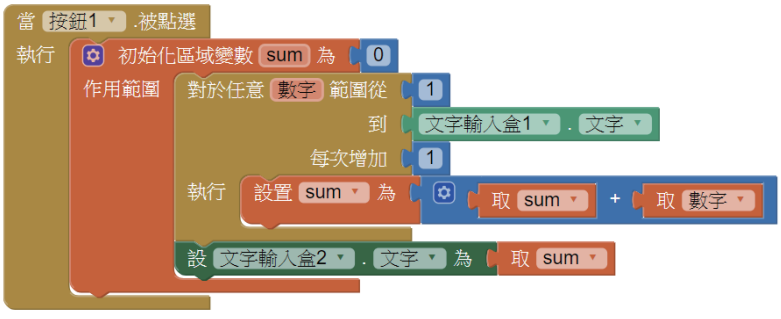
物件	屬性	屬性值
表格布局 1	行數 (因為將 row 翻譯成行數，所以實際上是設定列數)	3
標籤 1	文字	輸入 n 值
標籤 2	文字	加總結果
按鈕 1	文字	計算 1+2+3+…+n

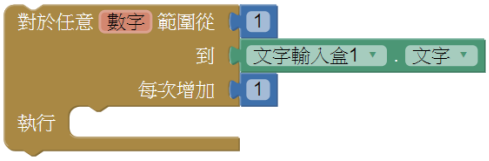
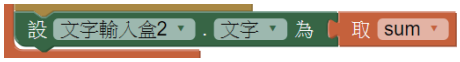
設定後，結果如圖。



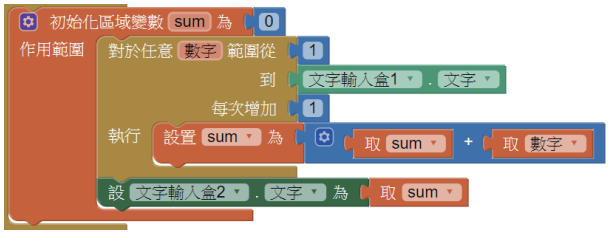
（四）建立程式拼塊與程式解說

在 Screen1 新增以下程式拼塊。

程式拼塊 與 程式解說


	(1) 當按鈕 1 按下時，區域變數 sum 初始化為 0。
	(2) 迴圈變數「數字」從 1 開始，每次遞增 1 變化到文字輸入盒 1 所輸入的 n 值為止。
	(3) 累加變數「數字」到變數 sum。
	(4) 將迴圈計算加總結果顯示到文字輸入盒 2。

使用「  」進行加總的原理，舉例說明如下表，在 App Inventor 2 中「  」的右邊「  」先運算，再將相加結果回存到左邊「  」。



數字	sum 加總過程	sum 加總後
數字 =1	sum=0+1	sum=1
數字 =2	sum=1+2	sum=3
數字 =3	sum=3+3	sum=6
數字 =4	sum=6+4	sum=10
數字 =5	sum=10+5	sum=15

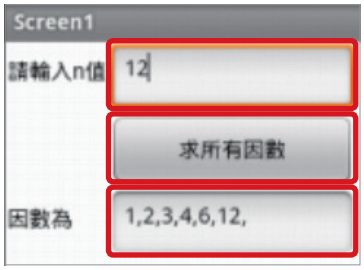
5-1-2 求因數

檔案名稱：ch05\Factor.aia

請寫一個程式，輸入一正整數 n，列出該數的所有因數。

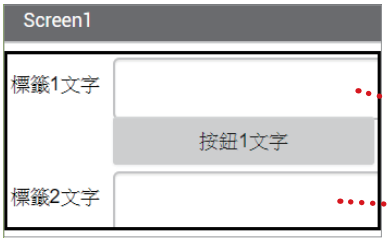
（一）預覽執行結果

在 n 值欄位輸入「12」，點選「求所有因數」，輸出所有因數結果為「1,2,3,4,6,12,」。



（二）建立使用者介面

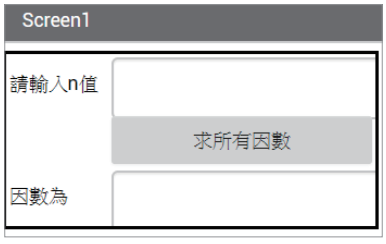
在 Screen1 新增一個表格配置，在表格配置元件內加入兩個標籤、兩個文字輸入盒與一個按鈕。

Screen1 的畫面	「元件清單」視窗中顯示所有元件間關係
	

（三）屬性設定

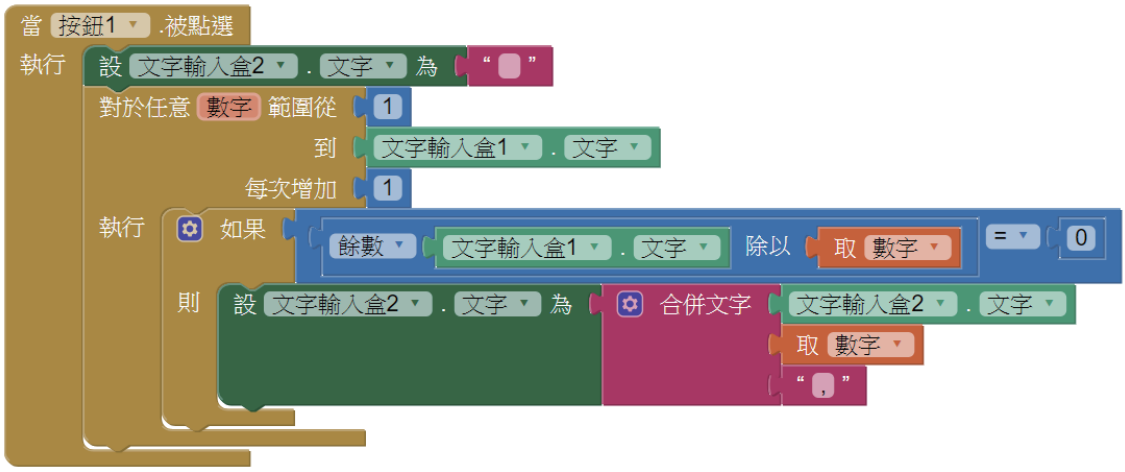
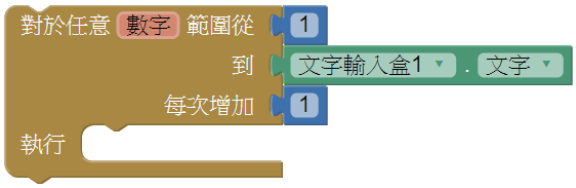
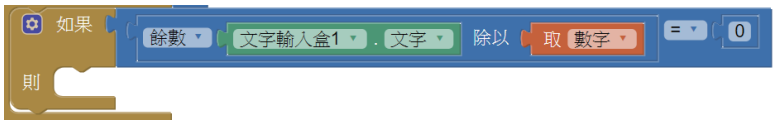
物件	屬性	屬性值
表格配置	行數 (因為將 row 翻譯成行數，所以實際上是設定列數)	3
標籤 1	文字	請輸入 n 值
標籤 2	文字	因數為
文字輸入盒 2	允許多行	True 或打勾
按鈕 1	文字	求所有因數

設定後，結果如圖。



（四）建立程式拼塊與程式解說

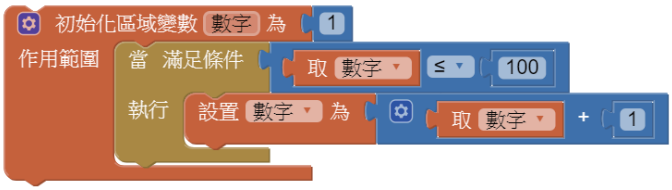
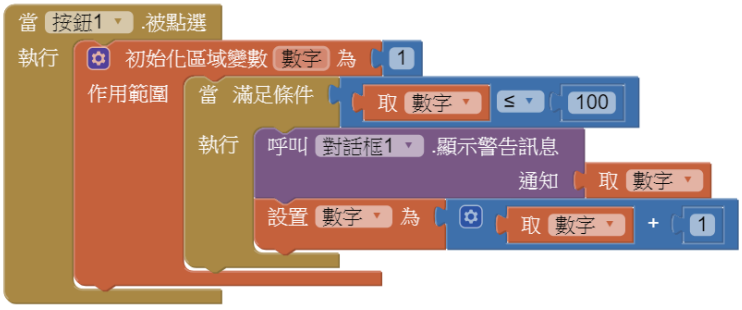
在 Screen1 新增以下程式拼塊。

程式拼塊 與 程式解說	
	
	(1) 當按鈕 1 按下時，初始化文字輸入盒 2 為空字串。
	(2) 迴圈變數「數字」由 1，每次遞增 1，直到到達文字輸入盒 1 所輸入的值，意即迴圈變數「數字」的值分別為 1、2、3、…、到文字輸入盒 1 所輸入的值為止。
	(3) 測試文字輸入盒 1 所輸入值是否被「數字」所整除。
	(4) 可以整除表示「數字」是文字輸入盒 1 輸入值的因數，將「數字」串接到文字輸入盒 2 並加上「,」。

5-2 迴圈結構－使用「當 滿足條件」

「當 滿足條件」迴圈結構與「對於任意 數字」迴圈結構都是迴圈結構，「當 滿足條件」迴圈結構用於執行次數不固定的迴圈結構，由迴圈中測試條件的結果決定是否繼續執行，測試條件為真時繼續執行迴圈；當測試條件為假時結束迴圈。為何需要不固定執行次數的迴圈？某些問題的解決過程無法確定執行的次數，例如：猜數字遊戲，假設有兩人（小明與小華）玩猜數字遊戲，小明先想一個數，小華去猜，小明就小華所猜數字回答「猜大一點」或「猜小一點」，直到小華猜到小明所想數字，這樣的猜數字遊戲就是不固定次數的問題，這樣的問題適合使用「當 滿足條件」迴圈結構。

「當 滿足條件」迴圈結構「滿足條件」後面所接條件測試，若測試結果為真時會不斷做迴圈內動作，直到測試結果為假時跳出「當 滿足條件」迴圈。

「當 滿足條件」迴圈語法 (「數字」由 1 變化到 100，每次加 1)	說明
	「當 滿足條件」迴圈外需使用「初始化區域變數 變數名稱 為」宣告迴圈變數並設定初始值，「當 滿足條件」迴圈變數由起始值變化到終止值，終止值由「滿足條件」後面的條件判斷決定，每重複執行一次程式，迴圈變數就會遞增(減)值，遞增(減)值設定在「設 變數名稱 為」，重複執行迴圈內程式拼塊，直到超過終止值後停止執行。
程式範例（印出 1 到 100）	
	

5-2-1 求小於 n 的級數和

檔案名稱：ch05\Series.aia

請寫一個程式輸入 n，求最小的 k，使得「1+2+3+4+⋯+(k-1)+k >= n」。

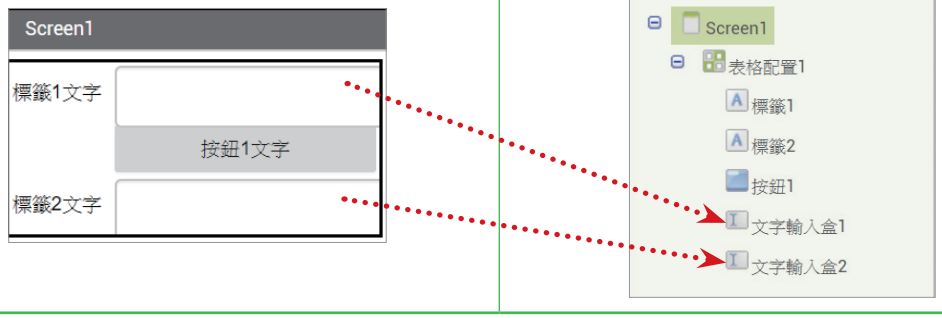
（一）預覽執行結果

在請輸入 n 值欄位輸入「10」，點選「求最小 k 值」，輸出 k 值為「4」，如右圖。

（二）建立使用者介面

在 Screen1 新增一個表格配置，在表格配置元件內加入兩個標籤、兩個文字輸入盒與一個按鈕。

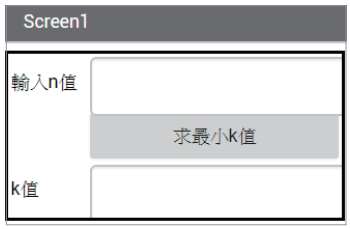
Screen1 的畫面	「元件清單」視窗中顯示所有元件間關係



（三）屬性設定

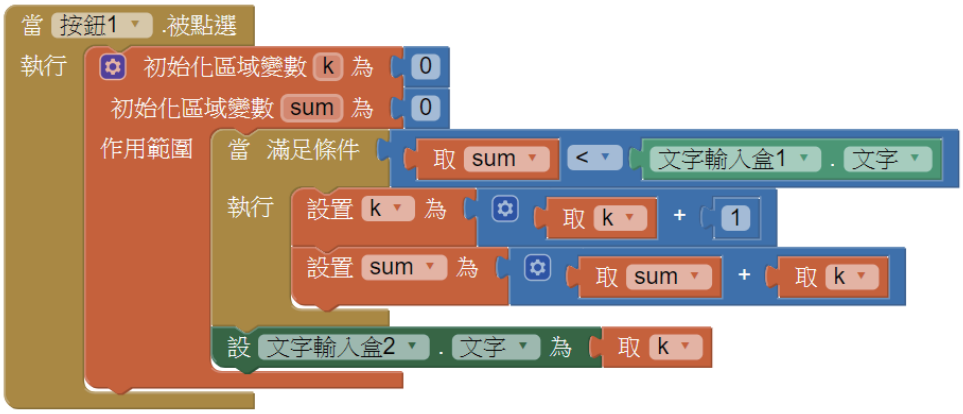
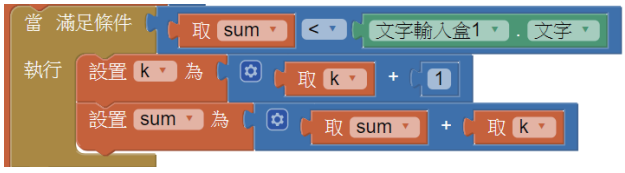
物件	屬性	屬性值
表格配置 1	行數 (因為將 row 翻譯成行數，所以實際上是設定列數)	3
標籤 1	文字	輸入 n 值
標籤 2	文字	k 值
按鈕 1	文字	求最小 k 值

設定後，結果如圖。

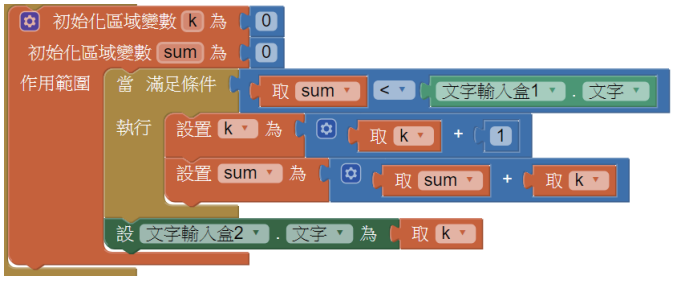


（四）建立程式拼塊與程式解說

在 Screen1 新增以下程式拼塊。

程式拼塊 與 程式解說	
	
	(1) 當按鈕 1 按下時，初始化區域變數 k 為 0，區域變數 sum 為 0。
	(2) 利用「當 滿足條件」迴圈計算 sum(級數和) 是否小於文字輸入盒 1 的輸入值，若是則繼續「當 滿足條件」迴圈，k 值遞增 1，累加 k 到 sum。
	(3) 將 k 值顯示到文字輸入盒 2。

程式碼在 App Inventor 2 中，k 值與 sum 值的變化如下表。



k 值	sum 相加過程	sum 相加後
k=1	sum=0+1	sum=1
k=2	sum=1+2	sum=3
k=3	sum=3+3	sum=6
k=4	sum=6+4	sum=10
…	…	…

EXERCISE

課後練習

一、選擇題

（　　）1. 下列何者適合用於重複結構？

(A) 如果 則　(B) 關閉畫面　(C) 開啟畫面　(D)「對於任意 數字」。

（　　）2. 求以下程式執行完畢，文字輸入盒印出幾個「*」？

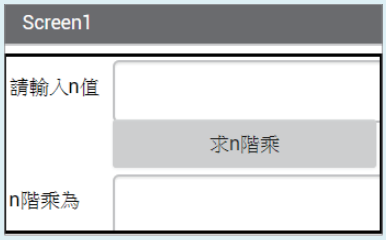
(A)3　(B)4　(C)5　(D)6。

（　　）3. 求以下程式完畢，sum 值等於？　(A)5　(B)10　(C)15　(D)20。

二、程式實作

階乘計算 (檔案名稱：ch05\ex_Fac.aia)

請輸入正整數 n，求 n 的階乘，n 階乘等於 1*2*3…*(n-1)*n。
版面配置如下圖。



1. 參考程式區塊

2. 程式執行結果

在請輸入 n 值欄位輸入「10」，點選「求 n 階乘」，則顯示「3628800」。