

Beginner's guide to Construct 2

感謝你選擇 Construct2！讓我們開始製作你的第一個 HTML5 遊戲。

我們將製作“ Ghost Shooter” 的遊戲 Demo，所以你知道我們的目標是：讓主角看著滑鼠，然後用方向鍵移動、用滑鼠射擊怪物。您將學習做一個簡單的遊戲所需要的所有事情 - 從 layers 到 event 系統！

其他 Tutorials

有個另外的初學者教學：[How to make a platform game](#)，這個教學目標在做出一個跳躍和使用一些 platformer，而不是做一個由上往下看的射擊遊戲。

我們建議你可以用兩邊所學到的東西來做出一個遊戲。

這裡也有個：[How to make an Asteroids clone in under 100 events](#) by

Kyatric 的 Tutorial，內容比較困難，但是也很詳細

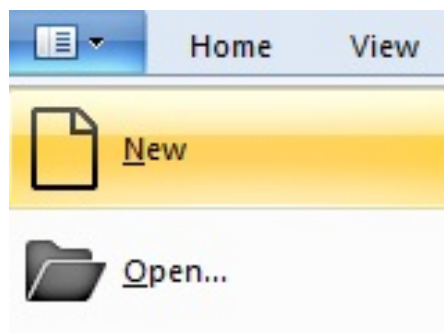
安裝 Construct 2

如果你還沒下載的話，請到官方網站下載最新的 Construct2，它目前只能在 Windows 上面運行，不過你做出來的遊戲可以在任何平台執行，像是 Mac、Linux、iPad 等等。

Construct2 同時也便於攜帶，你可以把它裝在 USB 隨身碟裡面，然後帶在身上。

Getting started

現在你安裝也開啟了 Construct2，請點擊 File>New。



在 New Project 視窗裡面，你不用改變任何選項，只要按下 Create Project 就好。Construct2 會建置一個簡單的專案並且生成.capx 檔。現在你看到一個空的 Layout – 你可以在這邊編輯和設定你的遊戲，想像 Layout 就像是一個遊戲的關卡或者選單場景，在接下來的文章裡面我們會稱他為 room、scene 或者 frame。

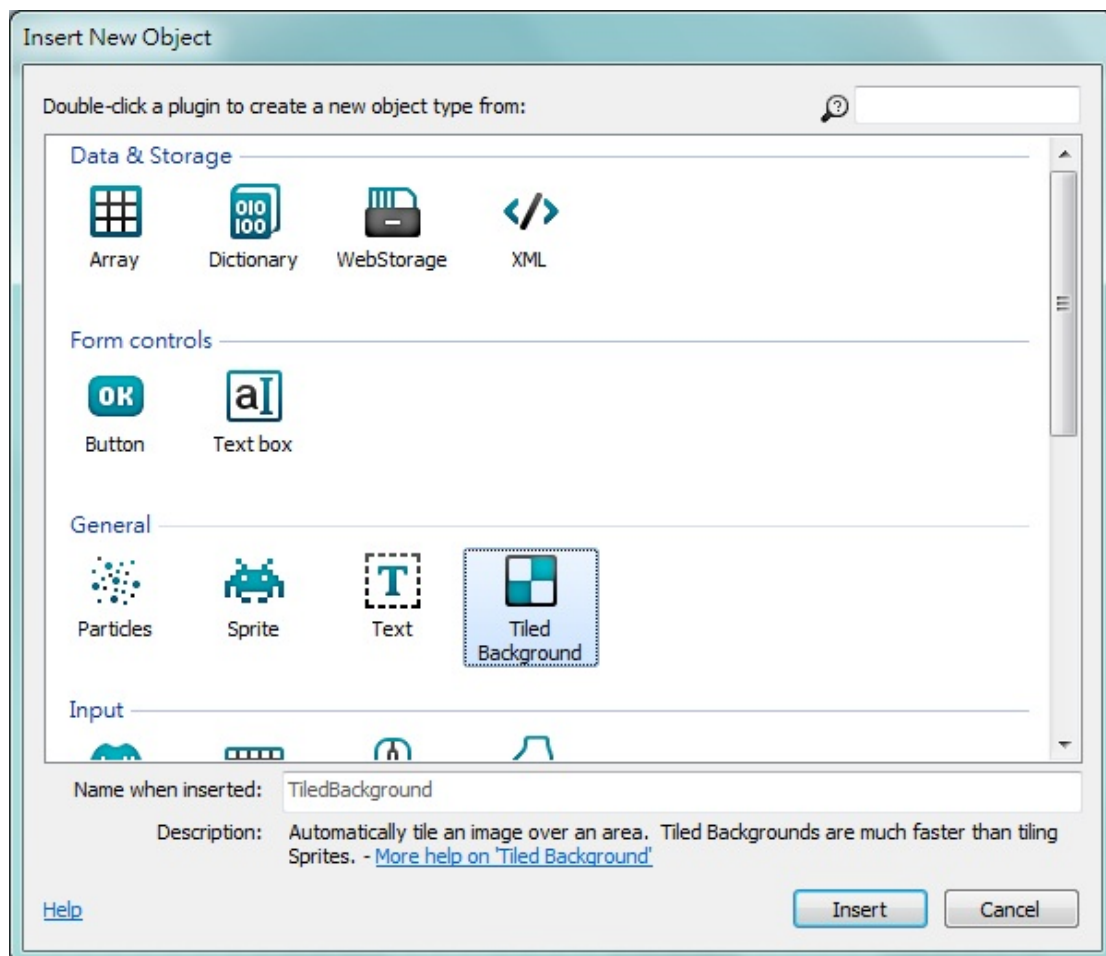
創製物件

Tiled Background

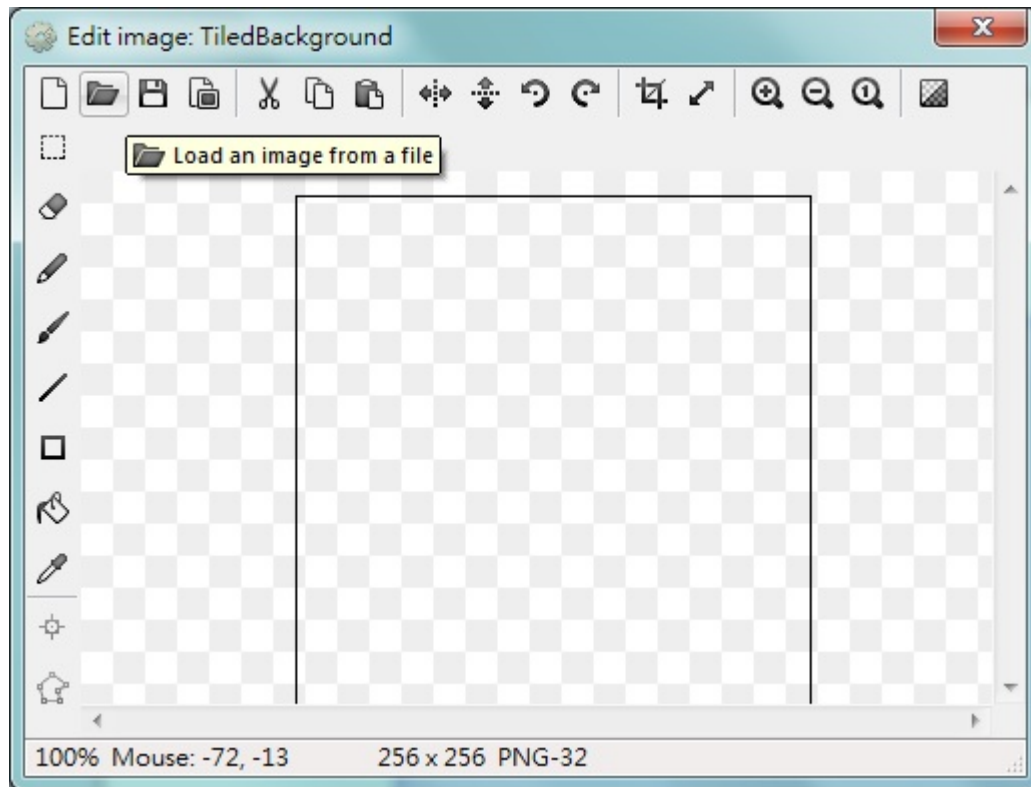
首先第一步我們需要先做出一個背景，而 Tiled Background 可以幫我們完成這件事情，首先找到背景的 texture：



然後雙擊你的 layout 空白的地方(你也可以按右鍵選擇 Insert New Object)，他會跑出一個新的視窗，請雙擊 Tiled Background。



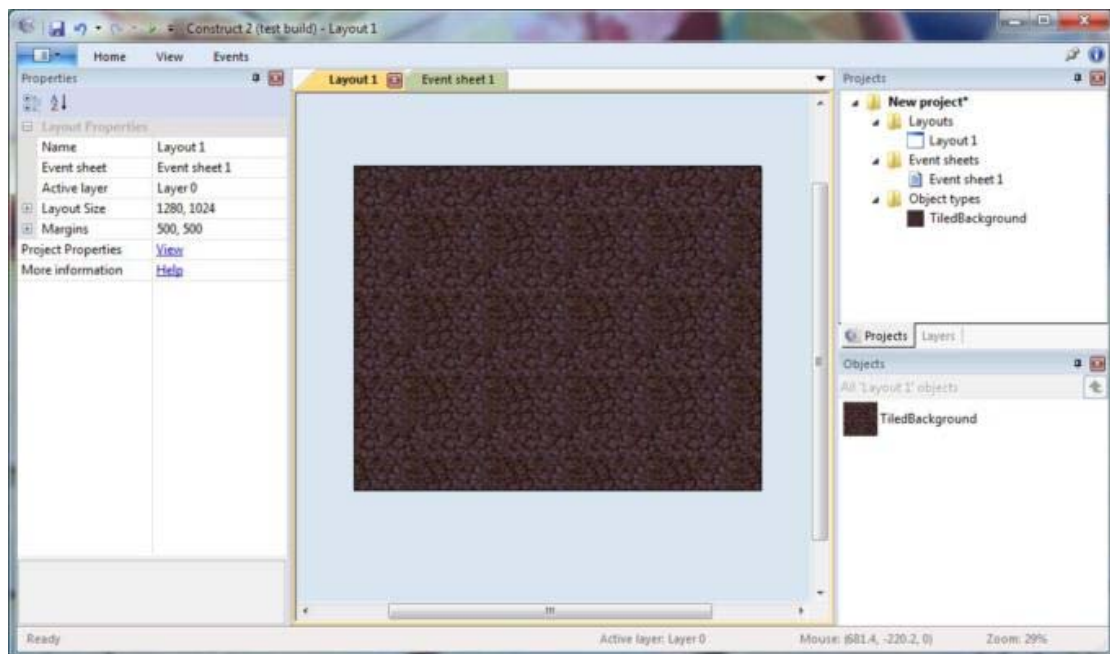
接下來選擇你要鋪上背景的地方，雙擊 layout 中間的位置，接下來會開啟 Texture editor 視窗，好匯入你的 texture。接下來從你的資料夾找到你的 Texture，並且匯入。



接下來關掉 texture editor 的視窗，如果系統有問你要不要存檔記得一定要存！現在你應該看到你的背景物件在你的 layout 上面。讓我們來改變他的大小好覆蓋整個 layout，請點一下你的背景物件，然後在左邊的屬性欄中設定他的大小和位置，位置設定在 0,0，大小設定在 1280,1024。

Object Type Properties	
Name	TiledBackground
Plugin	Tiled Background
Common	
Layer	Layer 0
Angle	0
Opacity	100
Position	0, 0
Size	1280, 1024
Instance variables	
Edit variables	Add / edit

現在我們來看整個 layout，按住 Ctrl+滑鼠滾輪來放大縮小，你也可以按住空格或者按住滾輪來左右平移整個 Layout。你的整個畫面應該要像這樣：



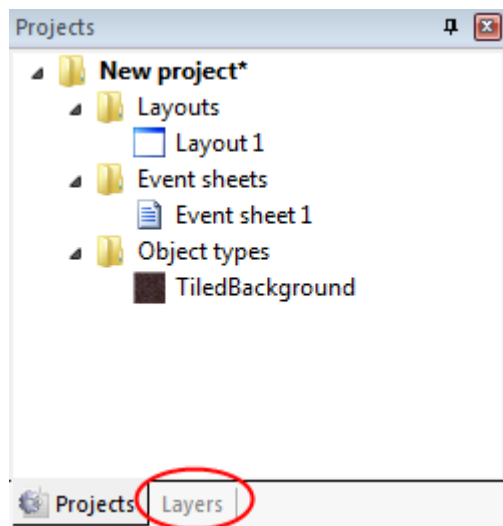
按下 Ctrl+0 或者點擊 view-zoom 到 100%好看到整個 1:1 的 layout。

增加一個 layer

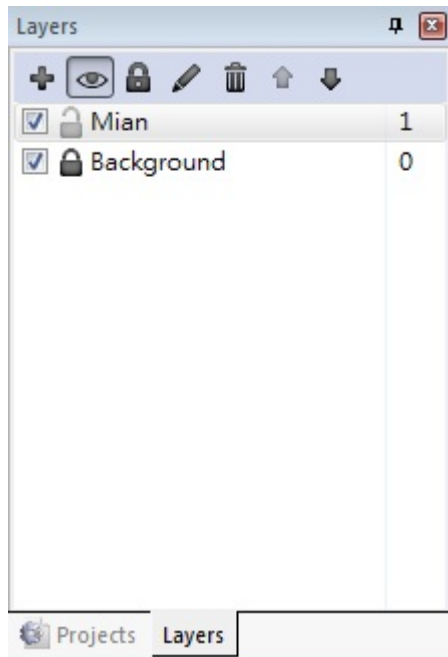
OK，現在我們要來增加其他更多的物件，但是我們必須小心別選到背景，或者直接將它鎖起來。要做到這點就得使用圖層系統(layering system)。

我們的遊戲場景可以同時包含好幾個圖層，你可以想像這就像是 Photoshop 的圖層概念，這樣可以輕鬆的安排出場的對象。圖層同時可以隱藏、鎖定等等效果。想像在遊戲中，我們要讓其他物件在背景物件上面行動，就可以使用圖層來做到。

點擊右邊的 Layers：



你應該會看到有個 layer 0 在我們的清單裡面，將它重新命名成 Background，因為它是我們的背景圖層，然後點擊綠色的十字圖案好增加一個新的圖層，我們叫他 Main。然後點擊黃色的鎖頭圖案來鎖定我們的背景圖層，這樣你就不會選擇背景的物件了，如果你有需要更改任何背景圖層上面的項目的話，在點一次黃色鎖頭來解鎖就可以了。



現在確認選到的圖層是 Main 圖層，選擇對的圖層是很重要的，因為你新增加的物件都會直接跑到你所選擇的圖層上，所以請隨時留意你選擇的圖層有沒有正確。

增加輸入物件

將你的焦點拉回到 Layout，雙點 Layout 來增加一個新的 Object，這次選擇 Mouse 物件，然後再做一次選擇 Keyboard 物件。

P.S.這些物件不會出現在 layout 裡面，它們是隱藏的，會自動在項目範圍內運作，現在在我們的所有 layout 中都可以接受滑鼠和鍵盤的輸入。

遊戲物件

這次我們來增加我們的遊戲物件，這邊是你需要的 Texture，把它存下來吧。

Player:



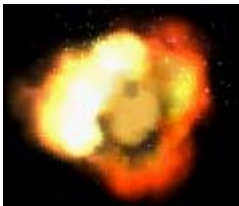
Monster:



Bullet:



Explosion:



每一個物件我們都用 Sprite Object 來建置，它只會顯示一個 Texture，可以移動、調整大小、旋轉。Sprite 物件是遊戲的主要組成。

接下來請依照步驟將上面的四個物件用 SpriteObject 加入：

1. 雙點 Layout 開啟 insert a new object 視窗。
2. 雙點 Sprite object。
3. 當滑鼠變成十字時，點擊 layout 的某個地方，系統應該會提醒你是 Main 圖層。

4. Texture 編輯器應該會跳出來，按下 open 的圖示，然後選擇你的 Texture。

5. 關掉 Texture 編輯器，存檔，你應該會看到你的物件在 layout 上面了！

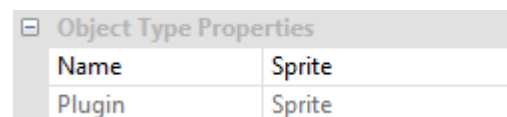
P.S.另外一個快速的方法是直接將圖片拖進去 Layout 裡面，系統會自動幫你建立一個 SpriteObject，不過請記住不要一次選擇所有圖片拖進去。

將子彈和怪物拖到 layout 框框的外面，因為遊戲一開始不會看到他們。

這些物件應該會被叫做 Sprite、Sprite2、Sprite3、Sprite4。

這樣的名稱會很容易搞混，所以請重新將它們命名成 Player、Monster、Bullet

和 Explosion，你可以選擇這些物件並且從屬性籃中的 Name 更改。



Object Type Properties	
Name	Sprite
Plugin	Sprite

增加 Behaviors

Behaviors 在 Construct2 裡面是包裝好的功能。例如說，你可以增加一個

Platform behaviors 到一個物件上，然後增加 Solid behaviors 到地板物件上，

然後你就可以在一個平台上面跳躍。

你可以做同樣的事件，但是會花很多時間，而且也沒什麼意義。不管怎麼樣，我

們都已經準備好了！

所以讓我們來看看有哪些 behaviors 是我們可以使用的：

8 Direction movement：這可以讓你用方向鍵來移動主角。

Bullet movement：這可以讓一個物件從你的對象前方做一個簡單的移動，這非常適合拿來製作子彈的功能。

Scroll to：使遊戲時的視點維持在一個物件上。

Bound to layout：這可以使你的物件不會離開你所設定的範圍。

Destroy outside layout：這可以使你的物件離開範圍後遭到刪除。好比說子彈，當物件飛離整個場景後他並不會消失，而是會一直存在者並且繼續飛下去，這樣會消耗你的效能，所以我們必須刪除這類超出範圍就不會再回來的物件。

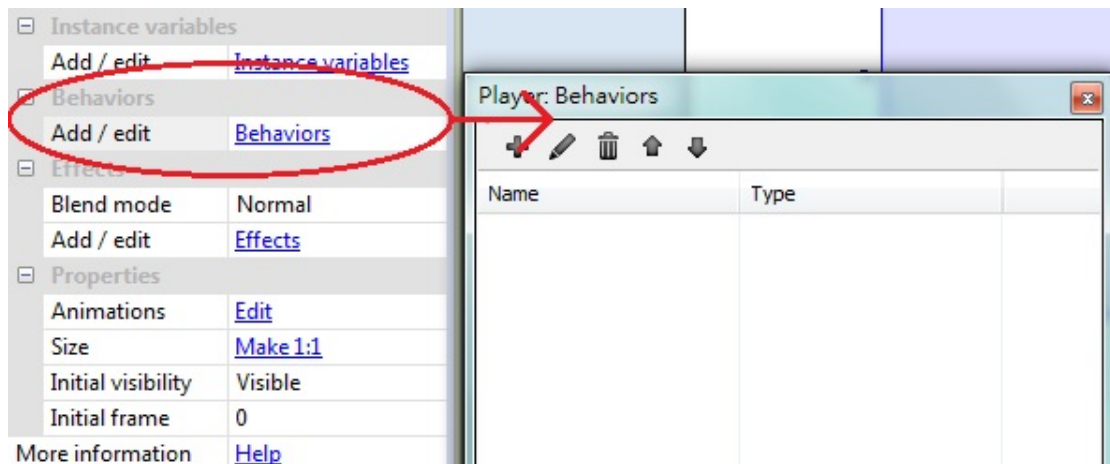
Fade：讓一個物件淡出，通常這個功能使用在特效上面。

讓我們來增加一些 behaviors 給需要的物件。

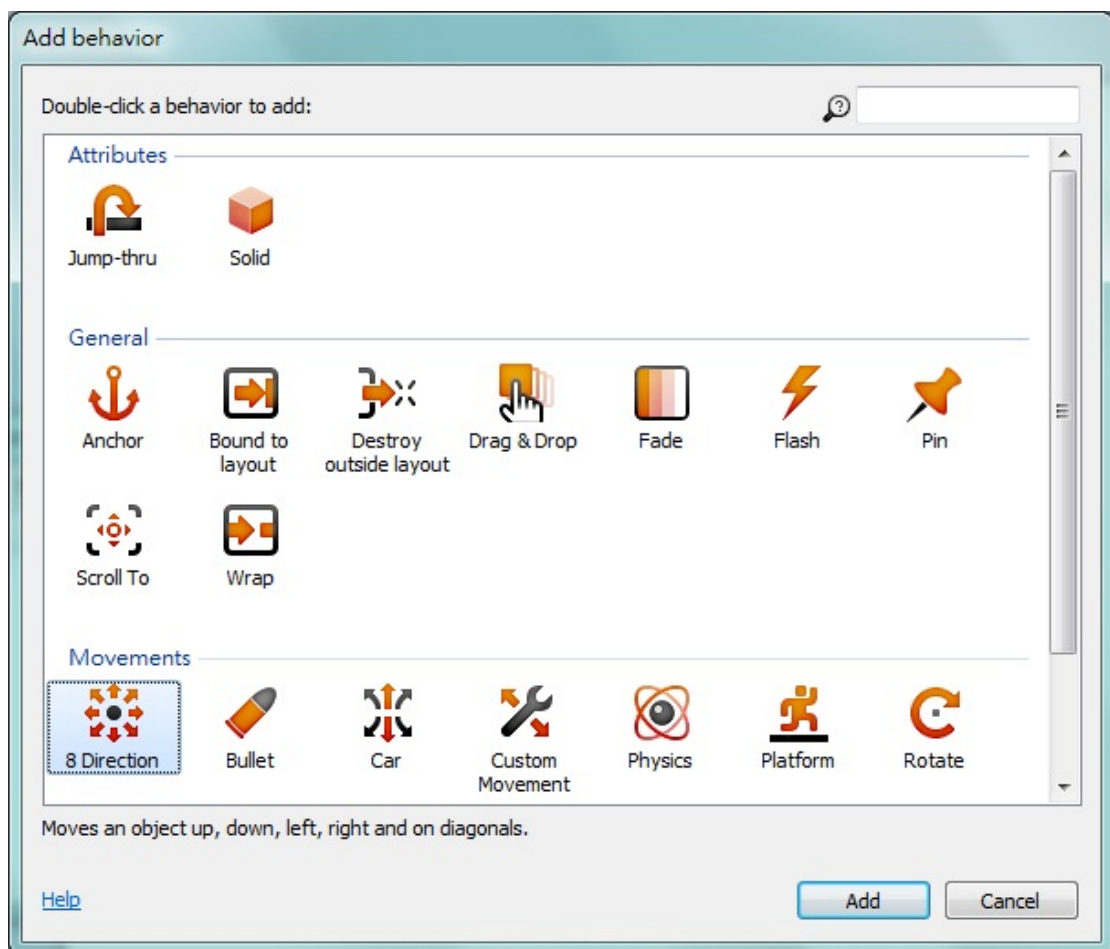
如何增加一個 behaviors

我們增加一個 8 Direction movement behaviors 給我們的 Player。

選擇 Player，然後再屬性欄裡面找到 Behaviors 的欄位，點擊 Add/Edit，接著它會跳出 Behaviors 視窗。

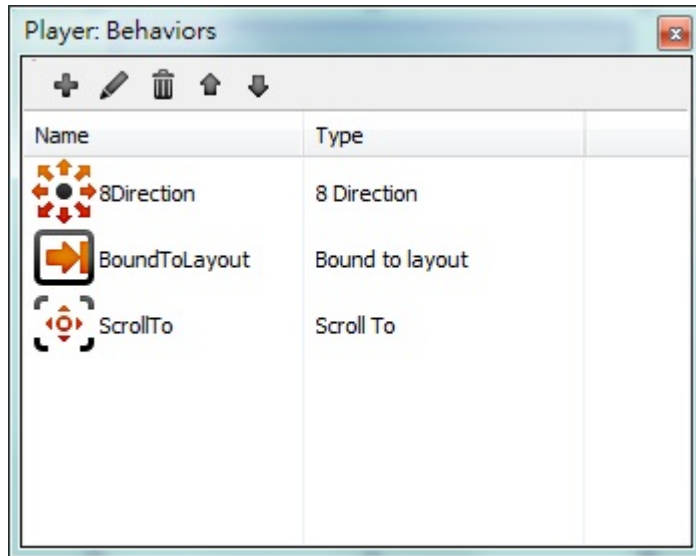


按下路色的十字圖案並且雙擊 8 Direction movement 來增加。

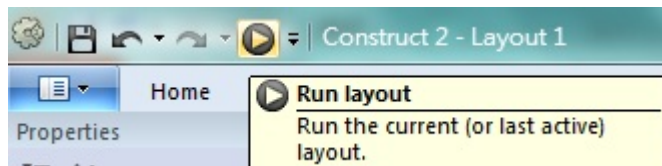


以同樣的步驟分別再加入 Scroll To 讓視點維持在 Player 身上·另外增加 Bound to Layout 防止 Player 走出整個場景。

接下來你的 Behaviors 視窗應該會像這樣：



關掉 behaviors 視窗，然後運行遊戲試試看！



你需要一個可以兼容 HTML5 的瀏覽器，像是最新的 FireFox 或者 Chrome 或者是 Internet Explorer9。

現在，你的畫面應該會跟著你的 Player，你可以使用你的方向鍵來移動你的 Player，然後你走不出整個場景外面，這都要歸功於 Behaviors 的功能。

接下來我們將使用 EventSystem 來增加一些自訂義的功能。

增加其他 Behaviors

我們可以用相同的方法幫其他物件增加 behaviors，選好你的物件點擊 Add/Edit

打開 behaviors 視窗，然後增加一些 behaviors 到其他物件上：

-增加 Bullet movement 和 Destroy outside layout 到 Bullet 物件上。

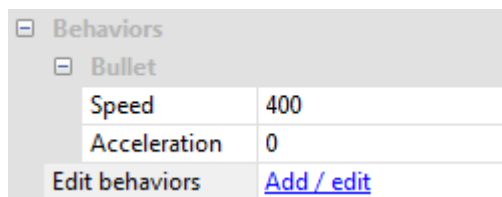
-增加 Bullet movement 到 Monster 物件上。

-增加 Fade 到 Explosion 物件上。

假設你試著運行遊戲，你會看到 Monster 會像自彈射出去那樣快速的移動過去，

讓我們放慢步調，選擇 Monster 物件，這時屬性欄裡面應該會有一些新的屬性

可以設定：



Behaviors	
Bullet	
Speed	400
Acceleration	0
Edit behaviors	Add / edit

這可以讓我們來調整它的速度，請將速度改變成 80。

接著選擇 Bullet 物件改變它的速度為 600，還有 Explosion 物件，改變它的 Fade

Out 的時間為 0.5 秒。

增加更多的 Monster

按住 Ctrl 然後點擊並且拖動 Monser 物件，你會發現你拖了另外一個 Monster

物件出來，這是非常簡單的複製方式。

這類複製出來的物件的 ObjectType 本質上是 Classes 物件，在事件系統中就是

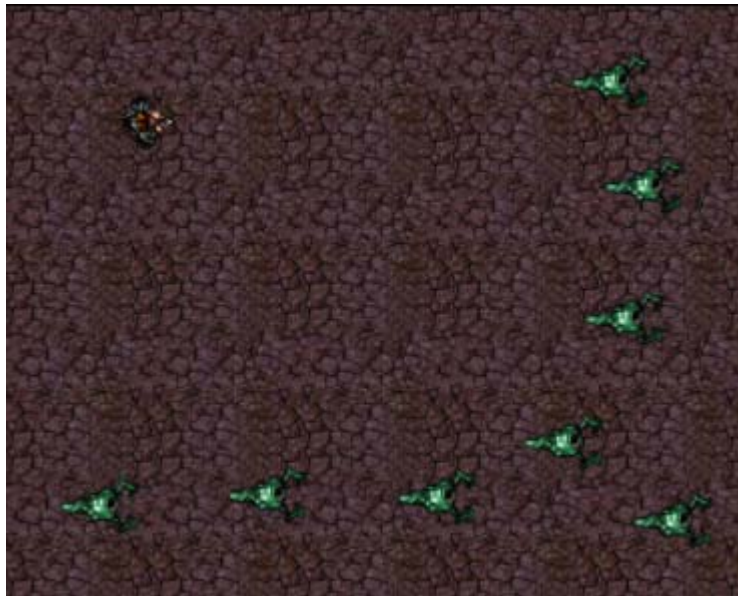
主要處理的物件類型。例如說：一個子彈打到怪物的事件。這實際上是再說子彈碰到所有怪物的事件，而不是單一個獨立的事件。

為了提升效能，這種 `ObjectType` 可以共用 `Texture` 和 `behaviors`。

例如說當玩家在玩線上遊戲時，畫面上有十幾隻相同的怪物，而玩家們不用下載幾十個相同的怪物，它們只需要下載一個，然後由系統去複製就好。

我們以後會介紹更多的對象類型和實際例子。

現在用 `ctrl` 來拖動複製 `Monster` 物件，請不要讓怪獸太接近 `player`，然後排成大概像下圖這樣：



接下來是時候來使用 `Event System` 添加一些自訂義的功能。

Event

首先在頂部的分頁中點選 **Event** 切換到事件表編輯器，你可以在這邊編輯對應遊戲中不同部分的不同 **Event** 或者組織，**Event** 裡面也可以包括其他 **Event**，所以你可以設定多層次事件。

關於 **Event**

在文字畫的空表單中，Construct2 跑一輪所有 **Event** 表單為 1 per tick。大部分的機器通常一秒會跳 60 per，所以 Construct 通常會運行事件表每秒 60 per 每次運行都會刷新一次場景。

Event 的執行順序會由上到下依序執行，也就說比較上面的 **Event** 會率先執行。

條件、動作、子事件

Event 要執行是必須包含條件的，例如：空白鍵是否有按下？

在所有條件都滿足時，所有的事件都會運行，例如：生出一個子彈物件。當動作執行後子事件也會跟著執行，這樣可以測試更多的條件、動作、子事件。

使用 **Event System** 就可以建置複雜的功能，並且運用在遊戲中。

我們再來說明一下，總之 Event 基本上是這樣的：

條件是否通過？

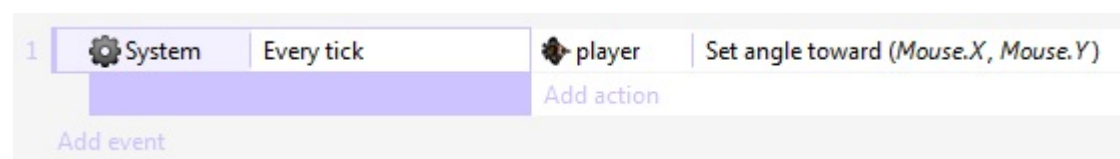
-->Yes：運行 Event 的動作。

-->No：執行下一個 Event。(不包含子事件)

這是一個有點過於簡單化的方法，Construct2 有許多不同的 Event 幫助你去組合不同的功能，就目前而言這是相當讚的！

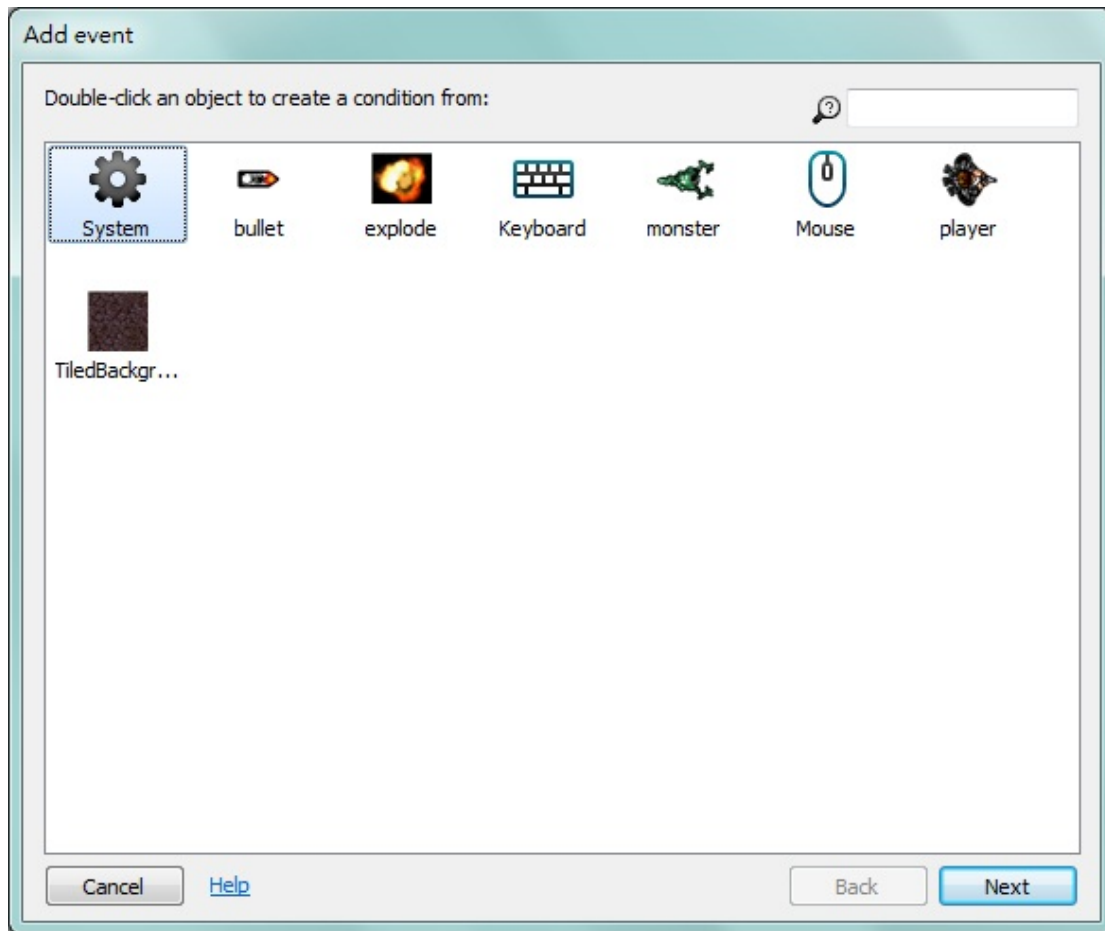
你的第一個 Event

我們要讓 Player 物件一直朝著滑鼠的方向看，會有點像這個樣子：



記住每運行一次 Event 場景都會刷新一次，所以我們要 Player 物件一直盯著滑鼠看的話就必須每次都執行，這樣它不論如何都會一直看著滑鼠。

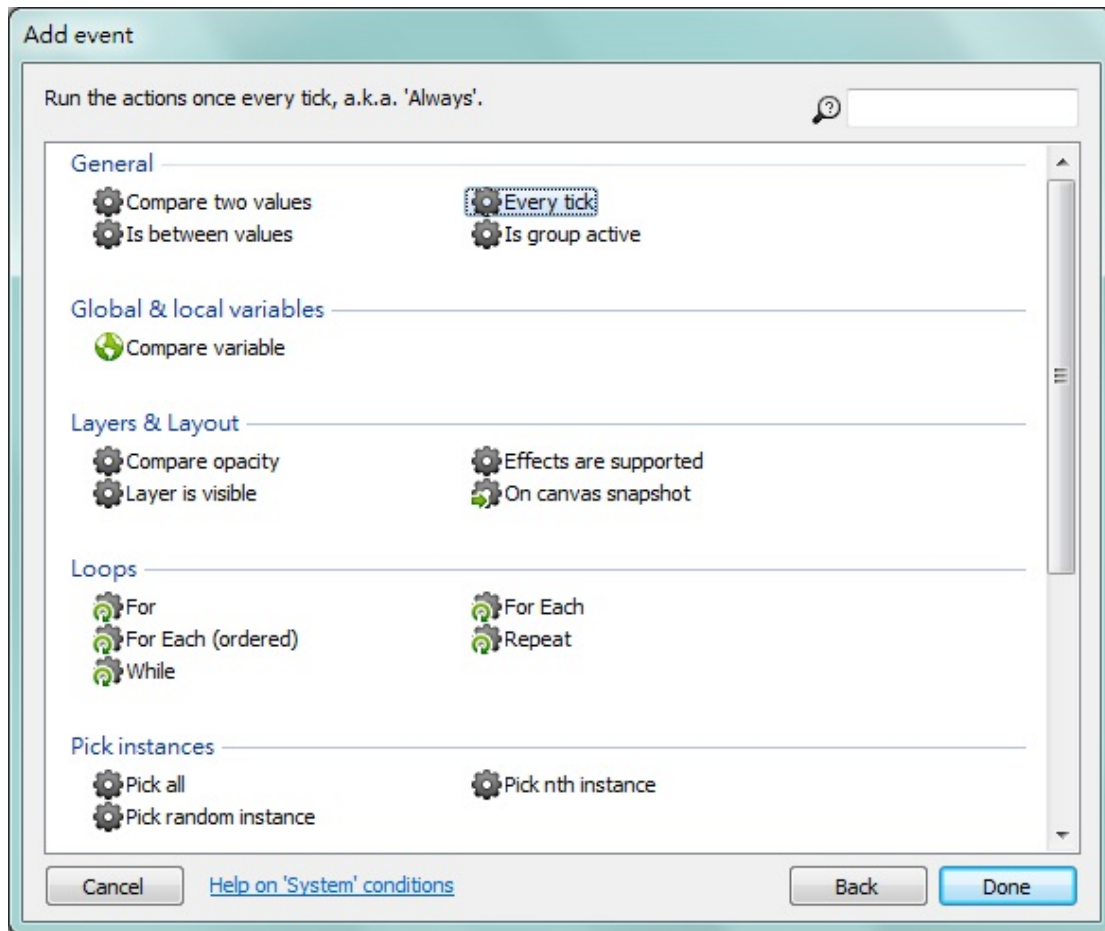
我們開始製作 Event 吧，雙擊 Event 頁面的空白處，這時會跳出一個增加條件的視窗。



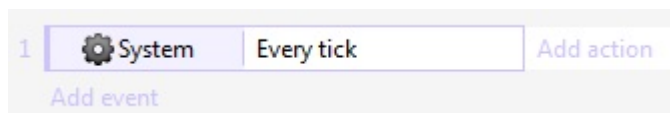
不同的物件會有不同的條件和動作來決定他們可以做甚麼。另外還有一個

System · 它是 Construct2 的內建功能。

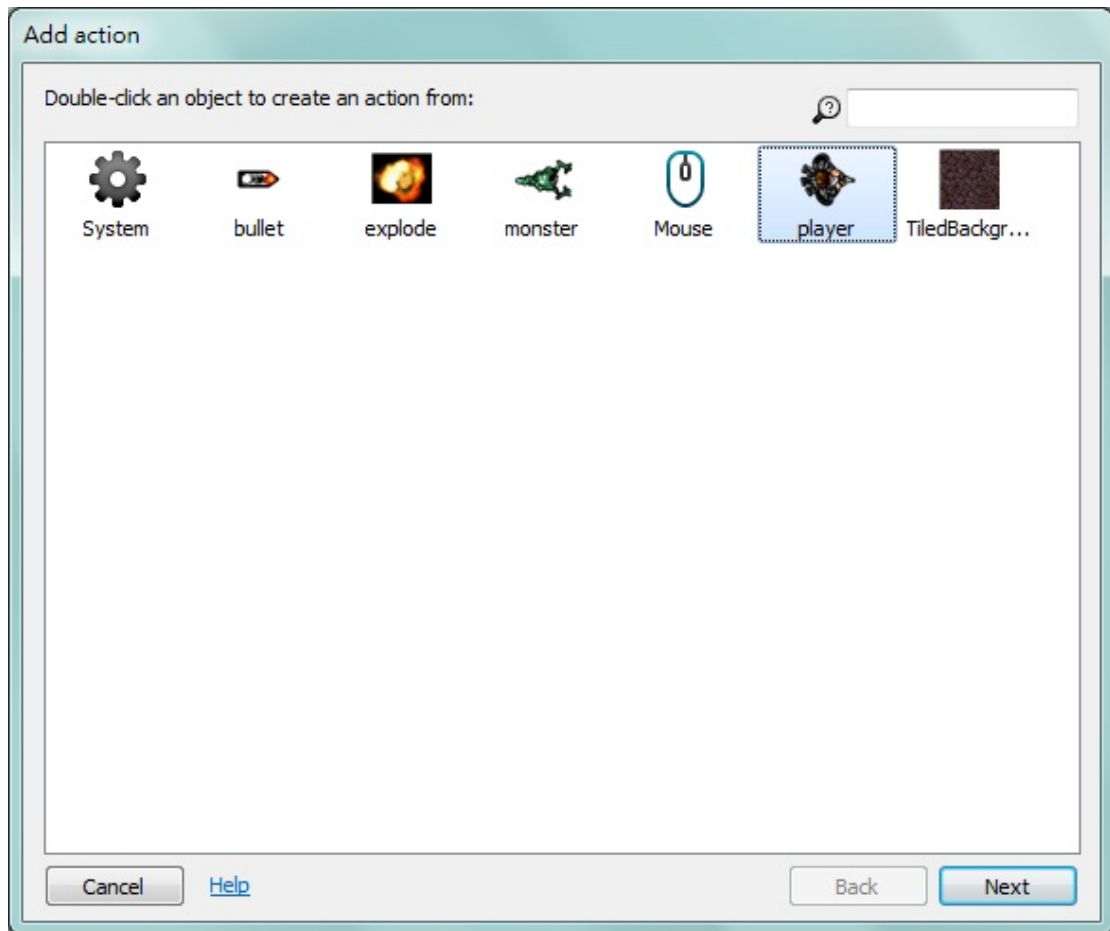
雙擊 System · 接下來會列出所有的 System 的條件：



雙擊 Event tick 條件來加入，接著這個視窗會關閉，並且建置出 Event，但是目前還沒有動作，所以你的畫面應該像這樣：

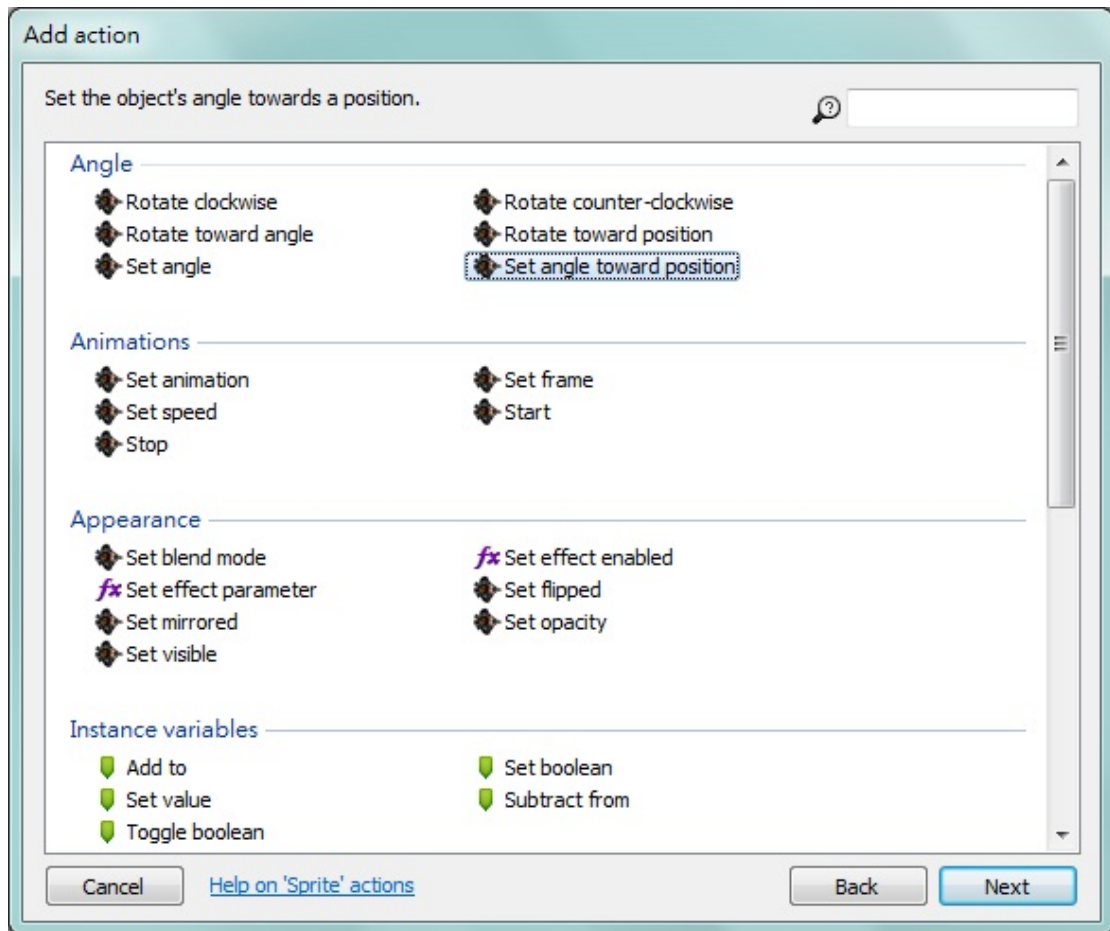


現在我們要增加動作使 Player 物件看著滑鼠，點擊 Add action 來增加事件(請確定你選擇的是 Add action，而不是 Add event)，Add action 視窗會長這個樣子：



這有點像是 Add event 視窗，不過這次是 Add action 視窗。

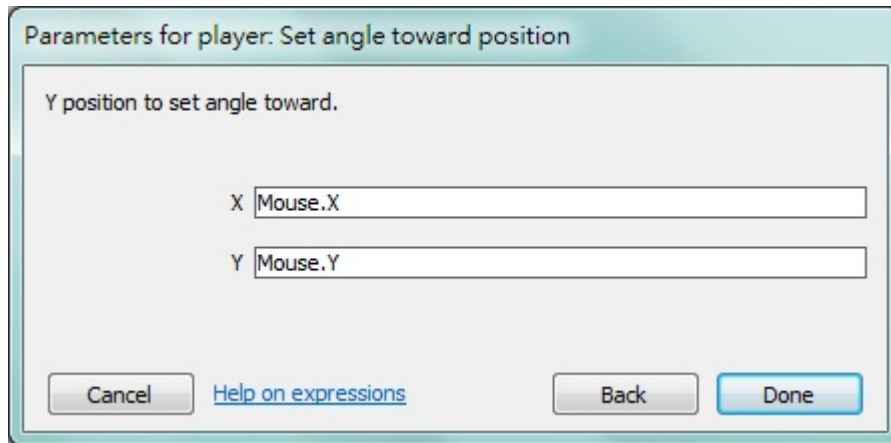
請雙擊 Player 物件，接著他會出現 Player 物件的操作列表。



注意，之前增加的 8-direction movement 仍舊有效，雖然現在我們不用去擔心這問題。

我們不用設定旋轉的角度，所以我們選擇 Set angle towards position，這可以讓物件的前方朝著你所訂的座標看，請雙擊它。

接下來要讓系統知道 Player 所需要看的座標外位置(X,Y)。



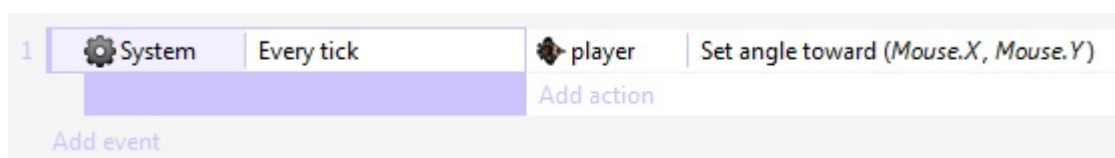
我們需要提供滑鼠的座標給他好讓 Player 物件可以看著它。

這邊我們輸入 Mouse.X 給 X，輸入 Mouse.Y 給 Y，這種寫法稱為表達式。他們就像是計算的項目，例如你也可以輸入 Mouse.X+100(雖然這種特殊的例子比較不常用到)，這樣的方式可以讓你使用任何對象的任何數據，或任何計算制定出行動中的參數和條件。

這種 Construct2 隱藏的參數相當強大且靈活。

如果你收掉一個錯誤訊息說“ Mouse is not an object name” ，請回到前面的章節確認你有沒有增加到 Mouse Object。

設定好後按下 Done 你的動作就會被添加進去了，它應該會長這樣：



這就是你的第一個 Event，試著運行遊戲看看，Player 物件應該會一直看著滑鼠。

增加遊戲功能

如果每個 Event 都要詳細敘述的話那這會是一個相當攏長的教學，讓我們下一個事件的說明稍微簡短一點。記住，增加一個 Event 的步驟是：

1. 雙擊插入一個新的 Event 或點擊 Add action Link 來增加一個 action。
2. 雙擊物件增加條件/動作。
3. 雙擊你需要的條件/動作。
4. 輸入參數，如果有需要的話。

讓 Player 可以射擊

當玩家點擊滑鼠應該要可以射出子彈，這需要讓子彈在 Player 的前方產生出來並且直線射出去。

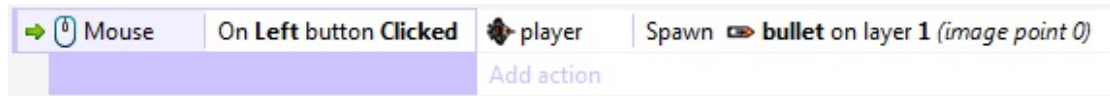
條件：Mouse>click left button

動作>Player 物件> Spawn another object > 物件選擇 Bullet 物件。

Layer 選擇 1(Main 圖層就是圖層 1，記住，Construct2 的數值都次從 0 開始)。

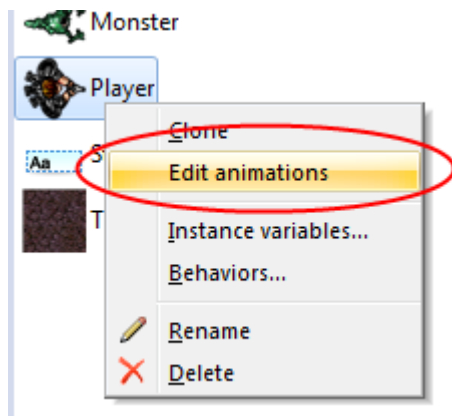
把 Image point 設定成 0

你的 Event 現在應該看起來樣這樣：

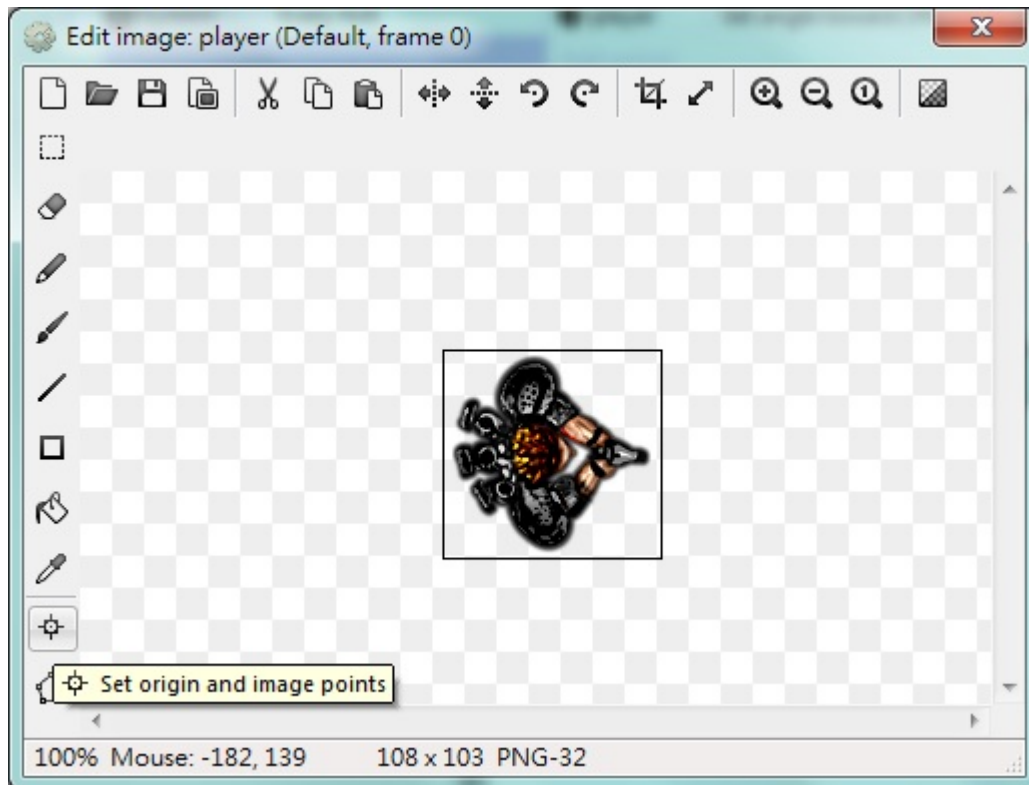


現在你試著運行遊戲，你應該可以發射出子彈了，可是子彈應該會從 Player 物件的正中心產生出來，而不是從他的槍射出，現在我們要修正這個錯誤。

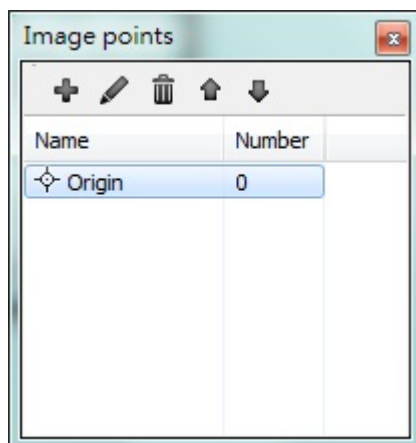
在右邊的 Project 視窗裡面找到 Player 物件，並且按下右鍵>Edit animaitons。



它會跳出圖像編輯器的視窗，點擊 Set origin and image points



然後會打開 Image points 視窗。



請注意物件的原點(Origin)顯示在圖片的中心點·如果要旋轉 object 它會照著這個紅點旋轉。

我們要添加另外一個圖像的點來表示槍·按一下十字圖示來添加·這時會出現一

個新的點-這就是我們圖像的新點，將這個點移到的前方：



關掉圖形編輯器，然後再次打開你剛剛設定的射擊 Event，接著把它的 image point 改變成 1。

接下來運行遊戲，你的子彈應該會從槍的前方射出了，不過你的子彈上面還沒有任何 Event。

接下來增加一些 Event 做子彈好殺死 Monster。

條件：Bullet > On collision with another object > Monster

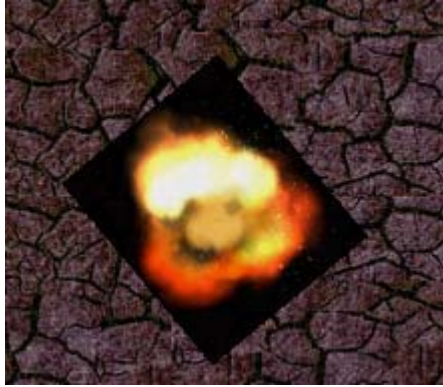
動作：Monster > Destroy

動作：Bullet > 產生物件 > Explosion, Layer1

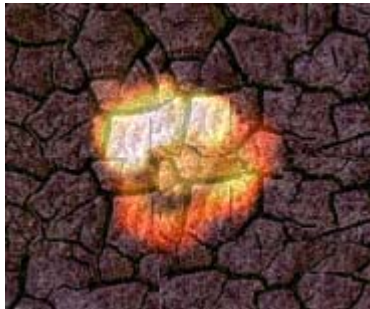
動作：Bullet > Destroy

Explosion 特效

運行遊戲然後試著射擊 Monster，你會看到 explosion 有旁邊很大的黑色區塊。



你可能早就猜到會有這種情況，不過別擔心，選擇 Explosion 物件將它的屬性欄拉到下面，Effect 的欄位設定成 Additive 之後再運行一次遊戲。

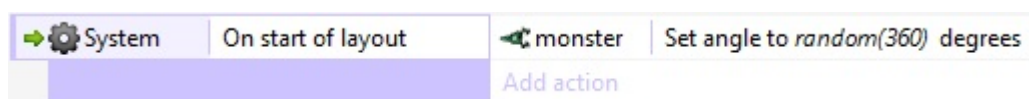


讓 Monster 變的聰明一點

現在 Monster 分布在場景的右側，為了讓遊戲有趣一點，我們讓怪物在一開始會隨機選擇一個方向前進。

條件：System -> On start of Layout。

動作：Monster -> Set angle -> random(360)。



不過這樣設定 Monster 還是會跑出場景外，得讓他們保持在場景內部，我們要做的是指定 Player 作為 Monster 的目標還有，限制他們別跑出場景外。

條件：Monster > Is outside layout。

動作：Set angle toward position > Player.X - Player.Y.

運行遊戲，放著一段時間你會發現 Monster 還是會乖乖的待在場景裡面，然後他們會朝各種方向前進，這是很難的 AI，可是 Construct2 裡面的設定就是這麼簡單。

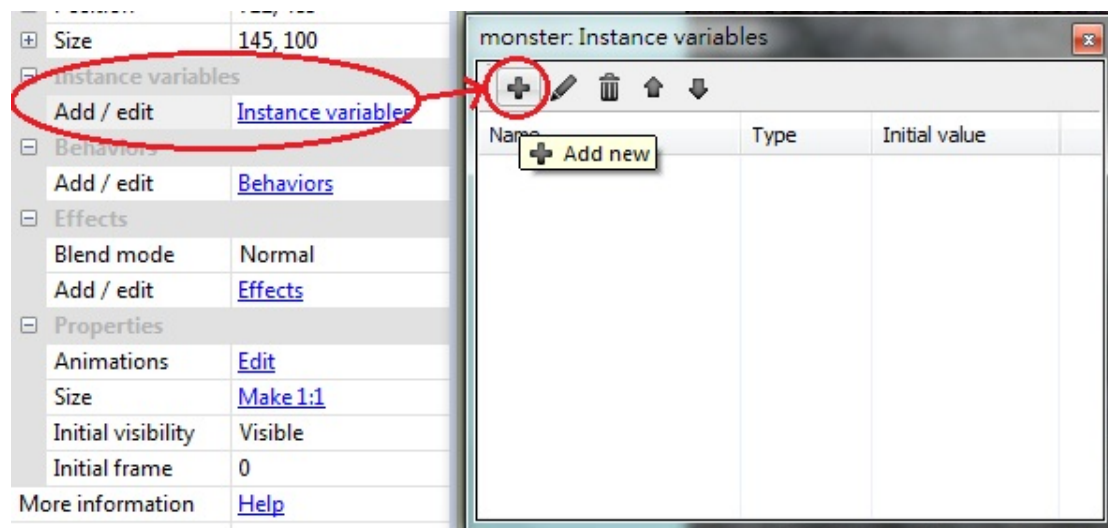
現在我們希望 Monster 要被子彈攻擊五次才會死亡，要怎麼做呢？

我們可以設定一個 Health 的數值來當作 Monster 的血量，我們可以使用 instance variables。

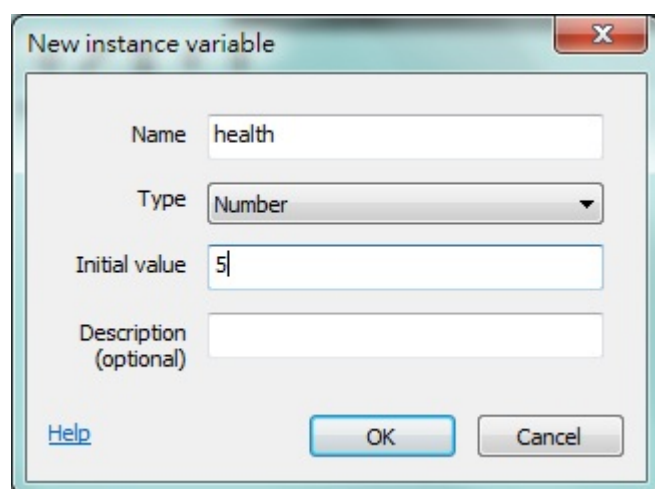
Instance variables

Instance variables 可以設定 Monster 的血量，一個變數是一個簡單的數值，可以改變，他們是分別儲存的，也就說每個 Monster 都可以有自己的血量。

我們來增加 health 變數來當作 Monster 的血量。在 Project 視窗點擊 Monster，讓左邊的屬性視窗變成 Monster 的。這時點擊 Edit variables 欄位的 Add/edit。

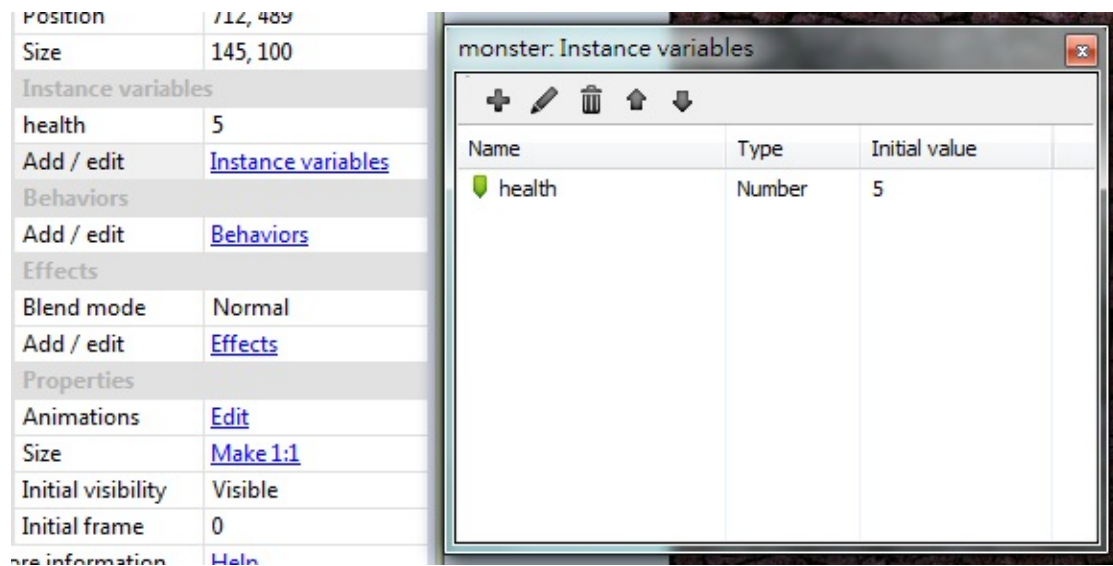


接著會跳出 Instance variables 視窗，這時候點擊綠色的十字按鈕增加變數。



在這個視窗裡面最上面填入 health，中間選擇 Number，最下面打入數字 5 設定好後就按下 OK。這時就設定好血量了，不過 Bullet 的 Event 還沒改變，你需要把它改成打到一次 health 的值就減 1 才行，然後當 health 值歸零時才刪除 Monster 物件。

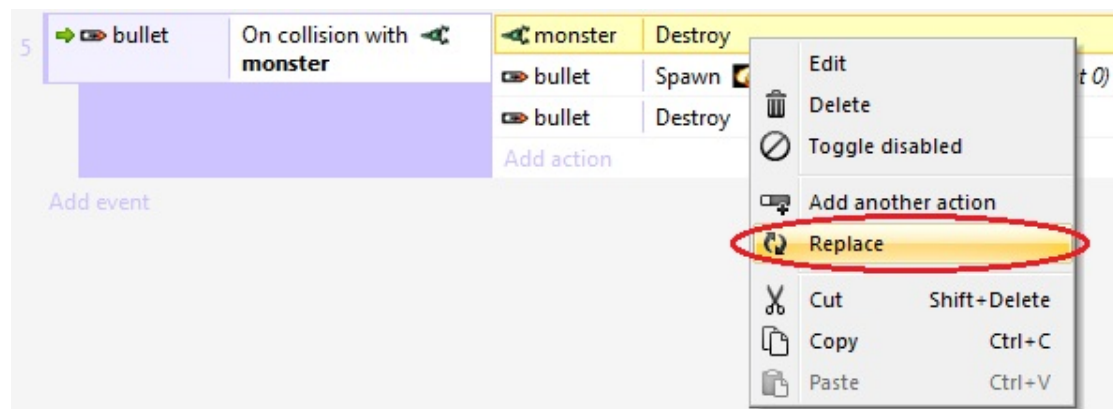
設定好血量的變數後你的畫面應該會像這樣：



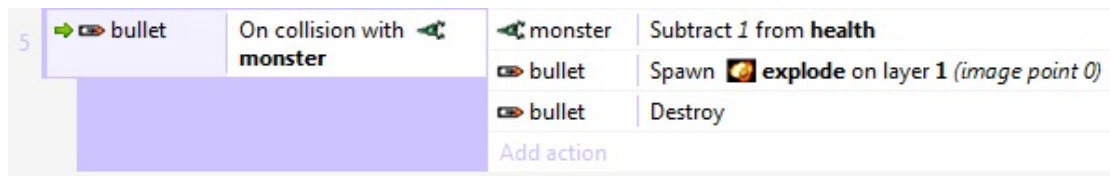
改變 Events

把畫面切到 Event 的分頁，然後找到 Bullet - on collision with Monster 的事件。

在 destroy monster 的動作上按下右鍵然後選擇 Replace。



出現視窗後，這一次我們要改變我們的動作，選擇 Monster -> Subtract from (in the Instance variables category) -> 變數"health"，然後再 Value 的地方輸入 1，按下 Done 這時你的畫面應該會變成這樣：



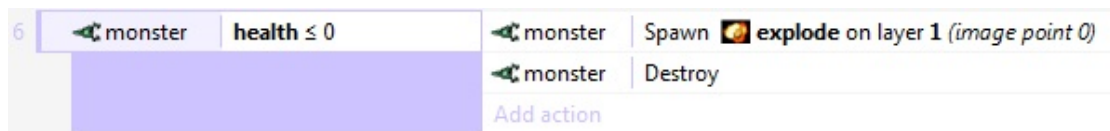
這時我們設定好減少血量後該來設定怪物死亡的 Event 了。

條件： Monster -> Compare instance variable -> Health, Less or equal, 0

動作： Monster -> Spawn another object -> Explosion, layer 1

動作： Monster -> Destroy

設定好後的畫面應該會像這樣子：



為甚麼要設定小於等於 0 呢？

這是程式設計的一個基本概念，現在我們設定是子彈打到怪物一次怪物的血量會減 1，但是當你有不同種類的武器時，怪物扣血的量就會不一樣，有可能會是減 2，那這樣怪物的血量變數就有可能會出現 -1、-3 這樣的數值，如果你是設定剛好等於 0 的話條件是就不會通過，所以我們會設定成小於等於 0，這樣怪物的血量不論是歸零還是小於 0 都算死亡。

現在試著運行遊戲，你應該要攻擊怪物五次怪物才會死亡。

保存分數

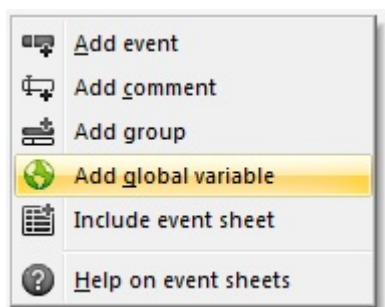
接下來我們要設定一個分數機制，我們需要一個新的變數。

你可能會想說我們可以把變數放在 Player 上面，這是個不錯的點子，可是你要知道在物件上面的變數是跟著物件的，也就是說物件一但被 Destroy 的話變數也會跟著消失。

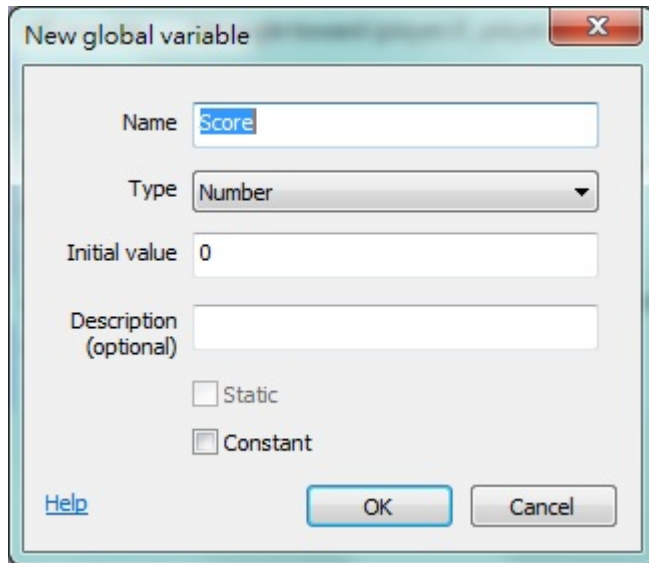
所以我們要一個不會受到物件消失影響的變數，要做到這點我們可以可使用 global variable。

這有點像是個 instance variable，不過 global variable 是個直接存在 Event 的變數，所以不會因為場景上的物件增減而消失。

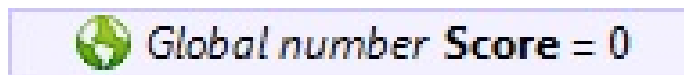
在 Event 分頁的空白處按下右鍵，之後選擇 Add global variable 來創建 global variable。



在最上面打入 Score 中間選擇 Number 最下面數值打入 0 然後按下 OK。

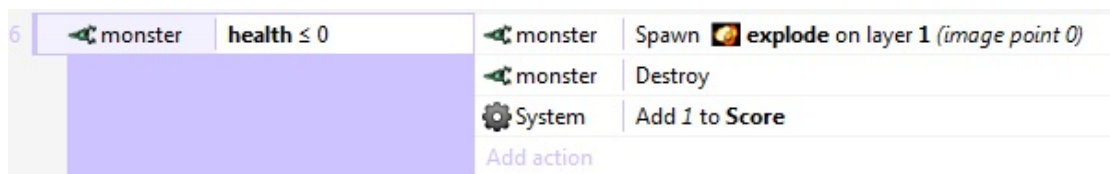


現在 Global variable 已經加入到 Event 裡面了。



接著我們回到 Monster: health less or equal 0 的 Event 再來增加一個動作。按下 Add action，然後選擇 System -> Add to (under Global & local variables) -> Score, value 1。

現在你的畫面應該像這樣：



現在你只要擊殺掉怪物分數就會往上增加了！

不過你在遊戲中看不到這個分數，現在我們來讓分數顯示在畫面上。

建置一個 heads-up display(HUD)

Head –up display(HUD)是一個介面顯示器，他可以用來顯示玩家的血量或者分數或者其他在遊戲中的數值。

我們先看到 layers 視窗，這邊我們新增一個圖層好讓 HUD 使用，選擇圖層後屬性欄應該會顯示圖層的屬性，請將 Parallax 設定成 0,0。

接著雙擊場景來建置新的物件，這次我們選擇 Text 物件，請將他放在場景的畫面的左上角，如果他的文字是黑色的話會很不容易看到，所以我們看到屬性欄裡面，把他改成的屬性改成粗體(bold)、斜體(italic)、黃色(yellow)，然後選擇一個較大的字體大小，並且重新調整他的寬度好容納整個文字。

你現在的畫面應該像這樣：



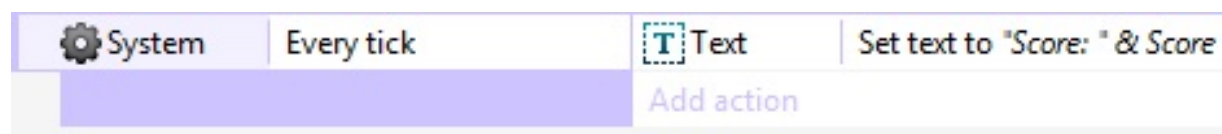
這時切換到 Event 分頁，我們必須一直更新玩家的分數，所以條件我們選擇 Every tick 來建置 Event，然後加入動作 Text > Set text。

"Score: " & Score

這邊開頭說的" Score: " 代表的是會一直顯示不會變動的文字，後面的 Score

則是在事件中的變數，我們要把這兩個東西一起放入屬性欄的 Text 中。

設定好後畫面應該像這樣：



收尾

首先我們要有一個可以不斷產生出的怪物，否則當場景上的怪物都打完了你也沒事做了，我們讓場景每三秒就創作出一支新的怪物出來。

條件: System -> Every X seconds -> 3

動作: System -> Create object -> Monster, layer 1, 1400 (for X),

random(1024) (for Y)

X 的 1400 是代表場景的寬度，Y 的 1024 則是代表場景的高度。

最後，讓 Monster 可以殺掉 Player。

Condition: Monster -> On collision with another object -> Player

Action: Player -> Destroy

Conclusion

恭喜你使用 Construct2 完成了你的第一個 HTML5 的遊戲！

(官方原文：

<http://www.scirra.com/tutorials/37/beginners-guide-to-construct-2/page-1>)