Unity
2D太空射擊遊戲

吳智鴻

國立臺中教育大學數位內容科技學系 教授

CHWU@MAIL.NTCU.EDU.TW

教學網址: HTTP://120.108.221.55/PROFCHWU/MULTIMEDIA/INDEX.PHP

學習目標

攝影機控制

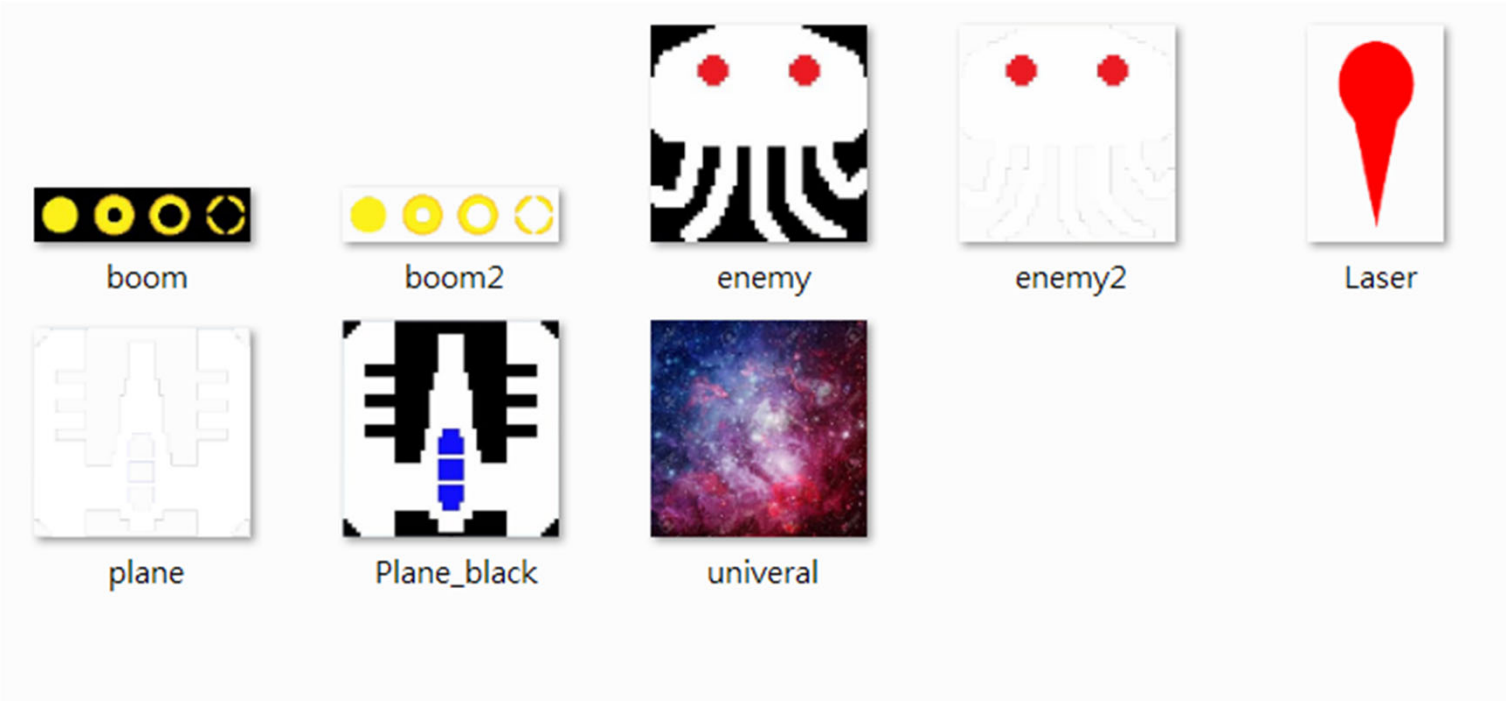
圖片控制

程式撰寫

選單製作

素材準備

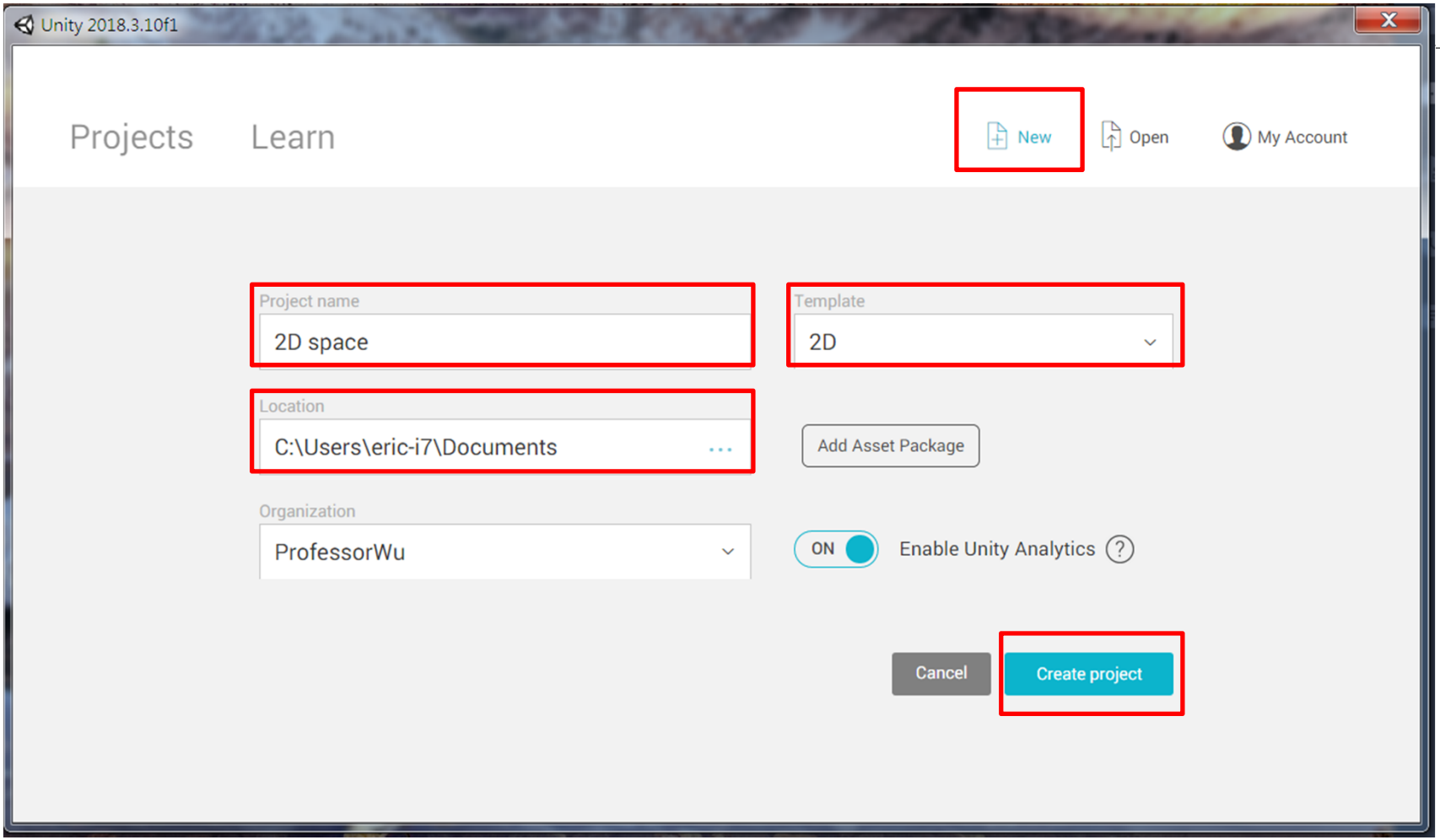
2D太空射擊遊戲



Unity

Unity 是一款由 Unity Technologies 研發的跨平台2D / 3D 遊戲引擎，可用於開發 Windows、MacOS 及 Linux 平台的單機遊戲，PlayStation、XBox、Wii、3DS 和 任天堂Switch 等遊戲主機平台的電動遊戲，或是 iOS、Android 等行動裝置的遊戲。Unity 所支援的遊戲平台還延伸到了基於 WebGL 技術的 HTML5 網頁平台，以及 tvOS、Oculus Rift、ARKit 等新一代多媒體平台。除可以用於研發電子遊戲之外，Unity 還是被廣泛用於建築視覺化、實時三維動畫等類型互動內容的綜合型創作工具。

New project


Unity 2018.3.10f1
Projects
Learn
New
Open
My Account
Project name
2D space
Template
2D
Location
C:\Users\eric-i7\Documents
Add Asset Package
Organization
ProfessorWu
ON
Enable Unity Analytics
Cancel
Create project

製作背景

600*800

PNG格式

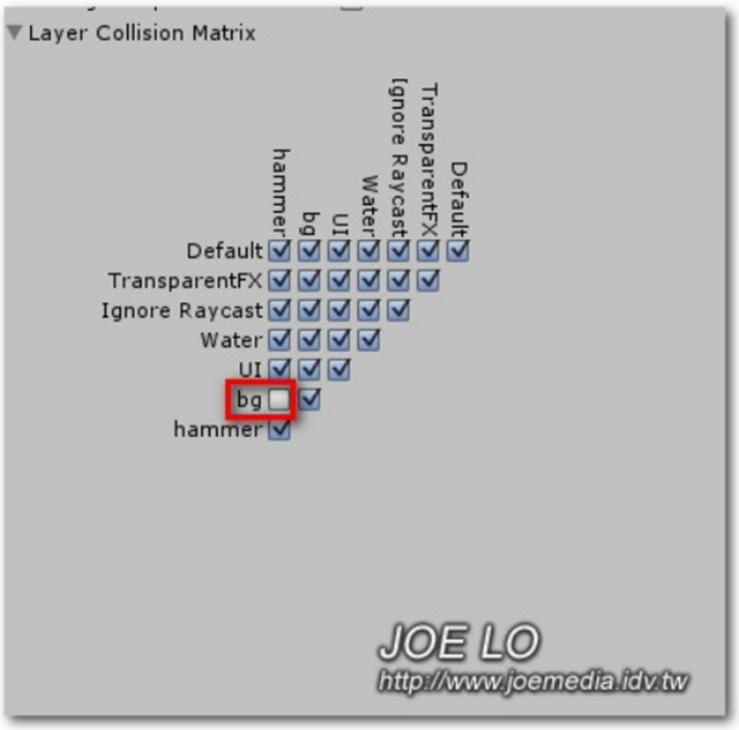
Sorting layer vs. Layer的差異

資料來源:https://blog.xuite.net/ljs.ctk/multimedia/344261876-layer+%E8%88%87+sorting+layer+%E7%9A%84%E5%B7%AE%E7%95%B0

在 unity2d 中如果要做到跟 flash 的圖層一樣有上下的關係，就要把相關的物件放在不同的 sorting layer ，例如前景層可以放在 foreground 的圖層，背景層可以放在 background 的圖層，利用在 sorting layer 拖曳圖層改變上下的關係。

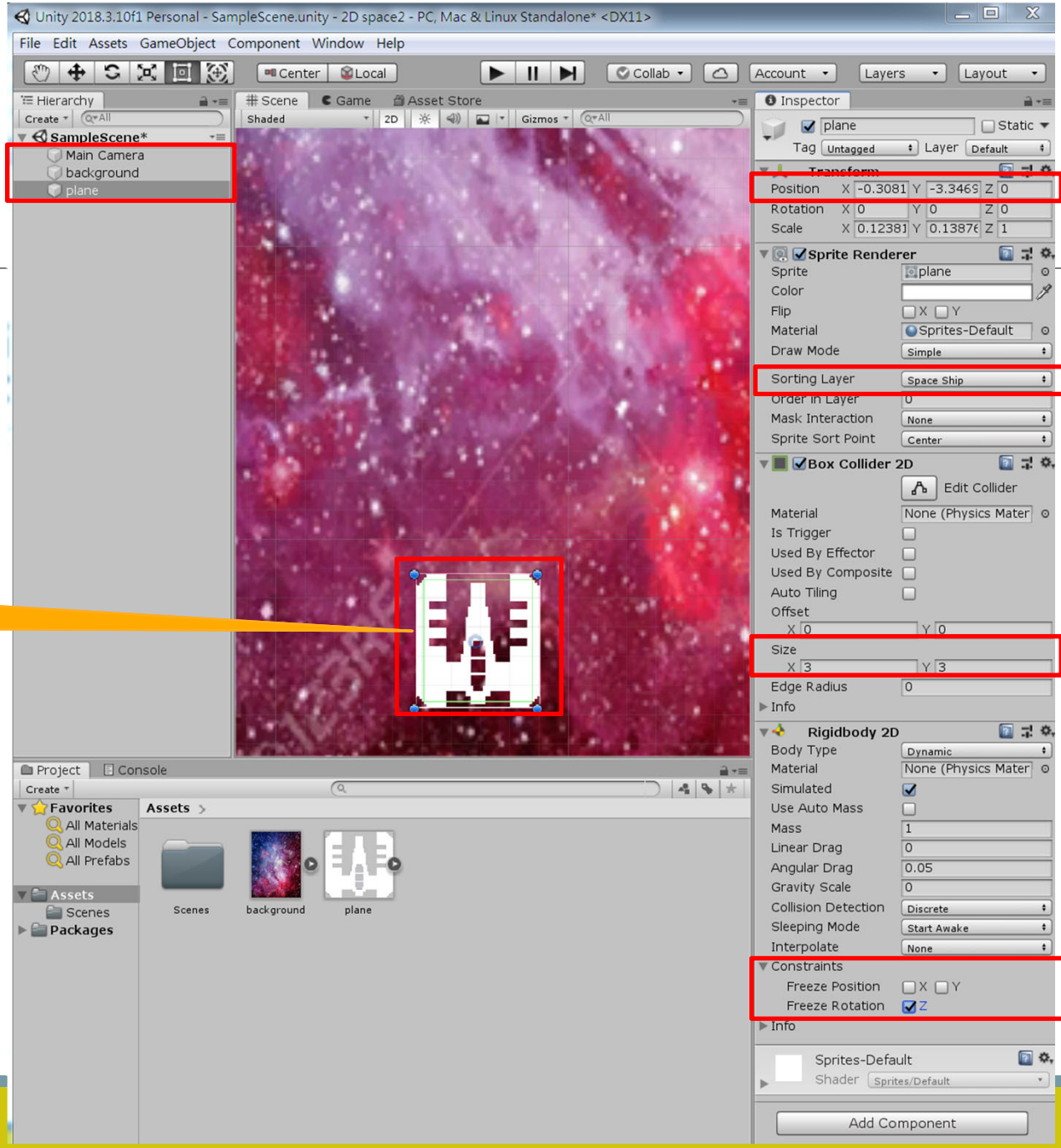
而 layer 最主要的功能是為了碰撞。假設做一個打地鼠的遊戲，槌子跟地 hammer 層，草地放在 bg 層，槌子跟地鼠都不要跟草地碰撞，則可以這定。

Edit->Project Settings->Physics



設定值

Box Collider2D
範圍(綠色)
可設定比飛機略小



ShipControl (C#) in Visual Studio

ShipControl#1

ShipControl.cs

Assembly-CSharp | ShipControl | Update()

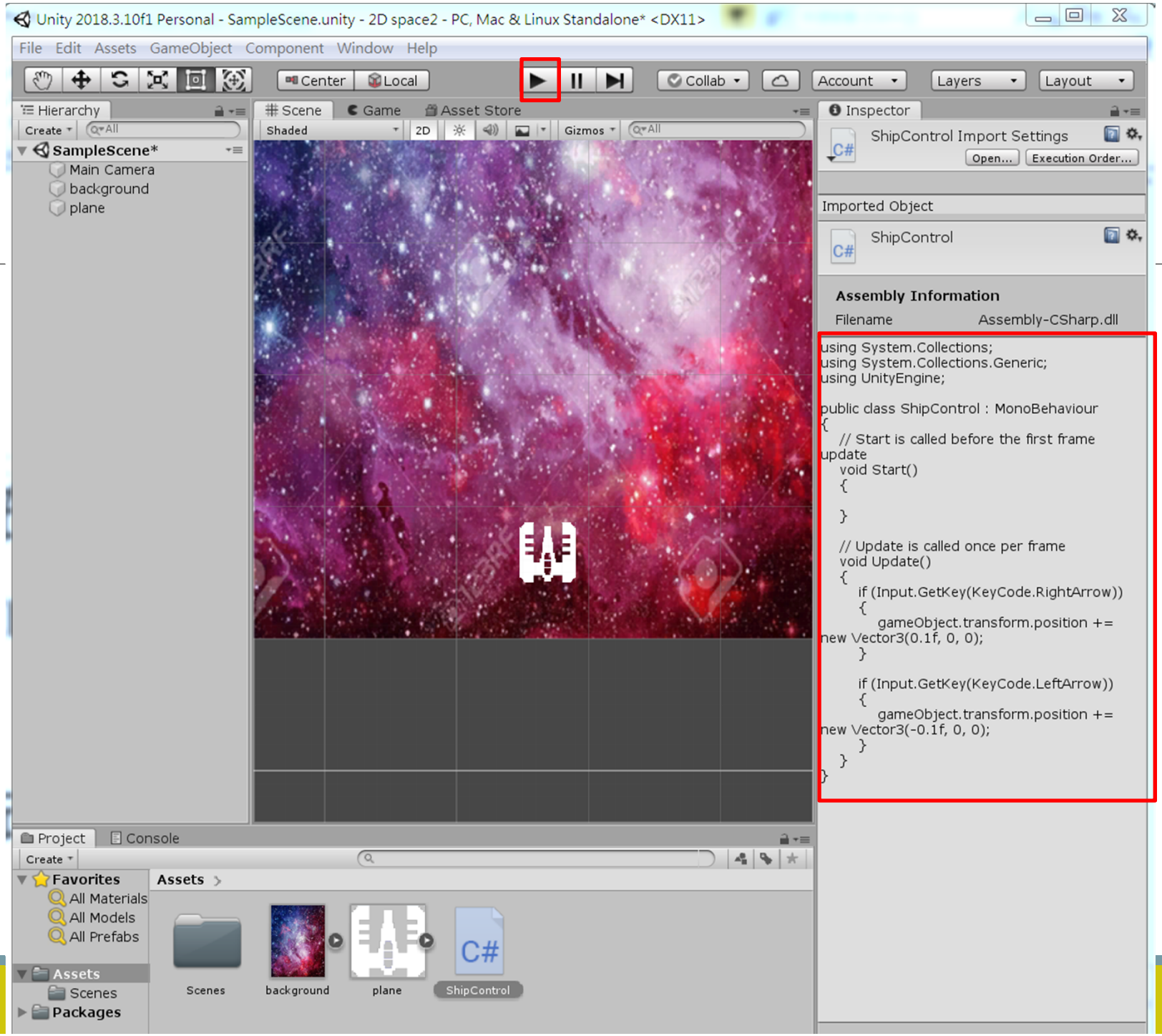
```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   public class ShipControl : MonoBehaviour
6   {
7       // Start is called before the first frame update
8       void Start()
9       {
10
11      }
12
13      // Update is called once per frame
14      void Update()
15      {
16          if (Input.GetKey(KeyCode.RightArrow))
17          {
18              gameObject.transform.position += new Vector3(0.1f, 0, 0);
19          }
20      }
21  }
22
```

加上這段程式碼

右箭頭

更改位置

X, Y, Z


Unity 2018.3.10f1 Personal - SampleScene.unity - 2D space2 - PC, Mac & Linux Standalone* <DX11>
File Edit Assets GameObject Component Window Help
Center Local
Collab Account Layers Layout
Hierarchy
Create
SampleScene*
Main Camera
background
plane
Scene Game Asset Store
Shaded 2D Gizmos
Inspector
ShipControl Import Settings
Open... Execution Order...
Imported Object
ShipControl
Assembly Information
Filename Assembly-CSharp.dll
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
public class ShipControl : MonoBehaviour
{
// Start is called before the first frame update
void Start()
{
}
// Update is called once per frame
void Update()
{
if (Input.GetKey(KeyCode.RightArrow))
{
gameObject.transform.position += new Vector3(0.1f, 0, 0);
}
if (Input.GetKey(KeyCode.LeftArrow))
{
gameObject.transform.position += new Vector3(-0.1f, 0, 0);
}
}
}
Project Console
Create
Favorites
All Materials
All Models
All Prefabs
Assets
Scenes
Packages
Assets
Scenes
background
plane
ShipControl

Try it !

現在太空船已經可以往右移動了

記得要先將程式存檔(Ctrl + S)

(1) 試著加上往左移動的程式碼吧

(2) 試著加上往上移動的程式碼吧

(3) 試著加上往下移動的程式碼吧

ShipControl#2 加入往左移動

ShipControl.cs

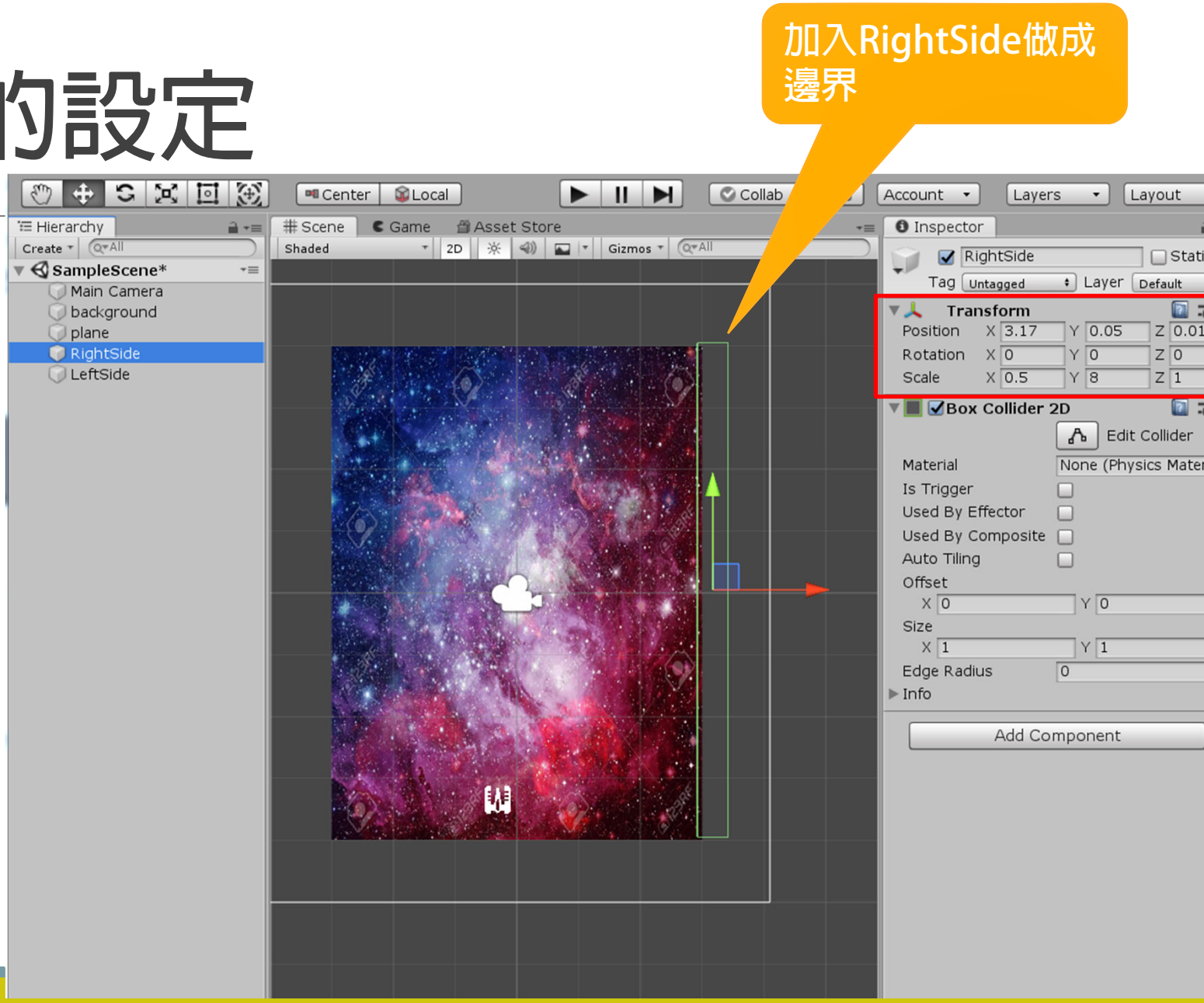
Assembly-CSharp | ShipControl | Update()

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   public class ShipControl : MonoBehaviour
6   {
7       // Start is called before the first frame update
8       void Start()
9       {
10
11      }
12
13      // Update is called once per frame
14      void Update()
15      {
16          if (Input.GetKey(KeyCode.RightArrow))
17          {
18              gameObject.transform.position += new Vector3(0.1f, 0, 0);
19          }
20
21          if (Input.GetKey(KeyCode.LeftArrow))
22          {
23              gameObject.transform.position += new Vector3(-0.1f, 0, 0);
24          }
25      }
26  }
27
```

RightSide的設定

GameObject->Empty

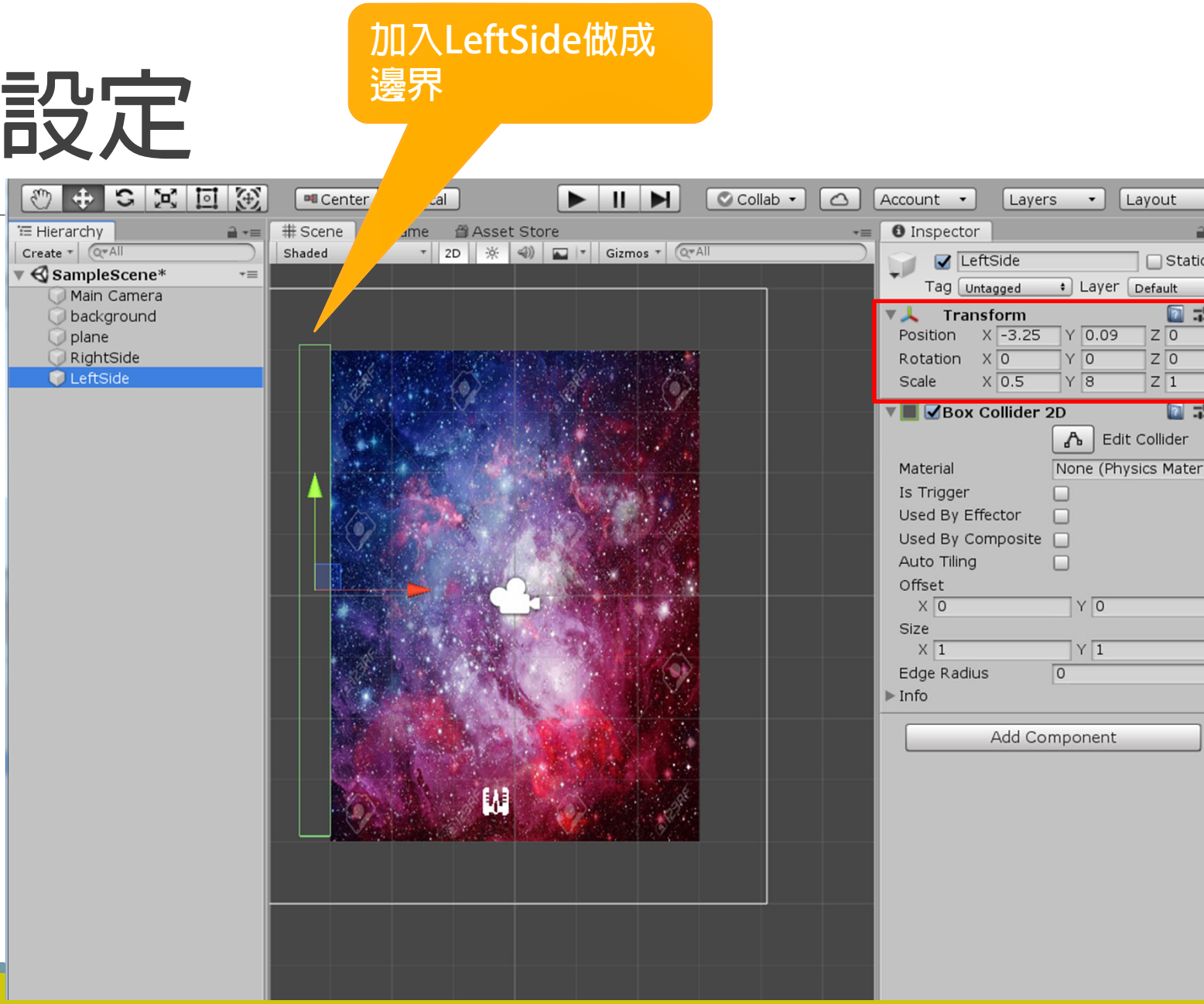
BoxCollider 2D屬性



LeftSide的設定

GameObject->Empty

BoxCollider 2D屬性



已經不會超過左右邊界了

Rigidbody & Box Collider 2D的概念

上網搜尋相關資料了解這兩者的差異。

https://dometi.com.tw/blog/unity-2d-lesson7/

Try it!

試著加入上方的邊界

試著加入往上&往下移動

試著更改背景

試著更改太空船

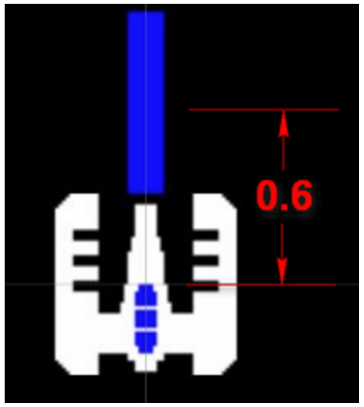
ShipControl#3 (發射子彈)

複製物件
Instantiate(物件, 初始位置,初始角度)

Laser.cs | ShipControl.cs

Assembly-CSharp | ShipControl

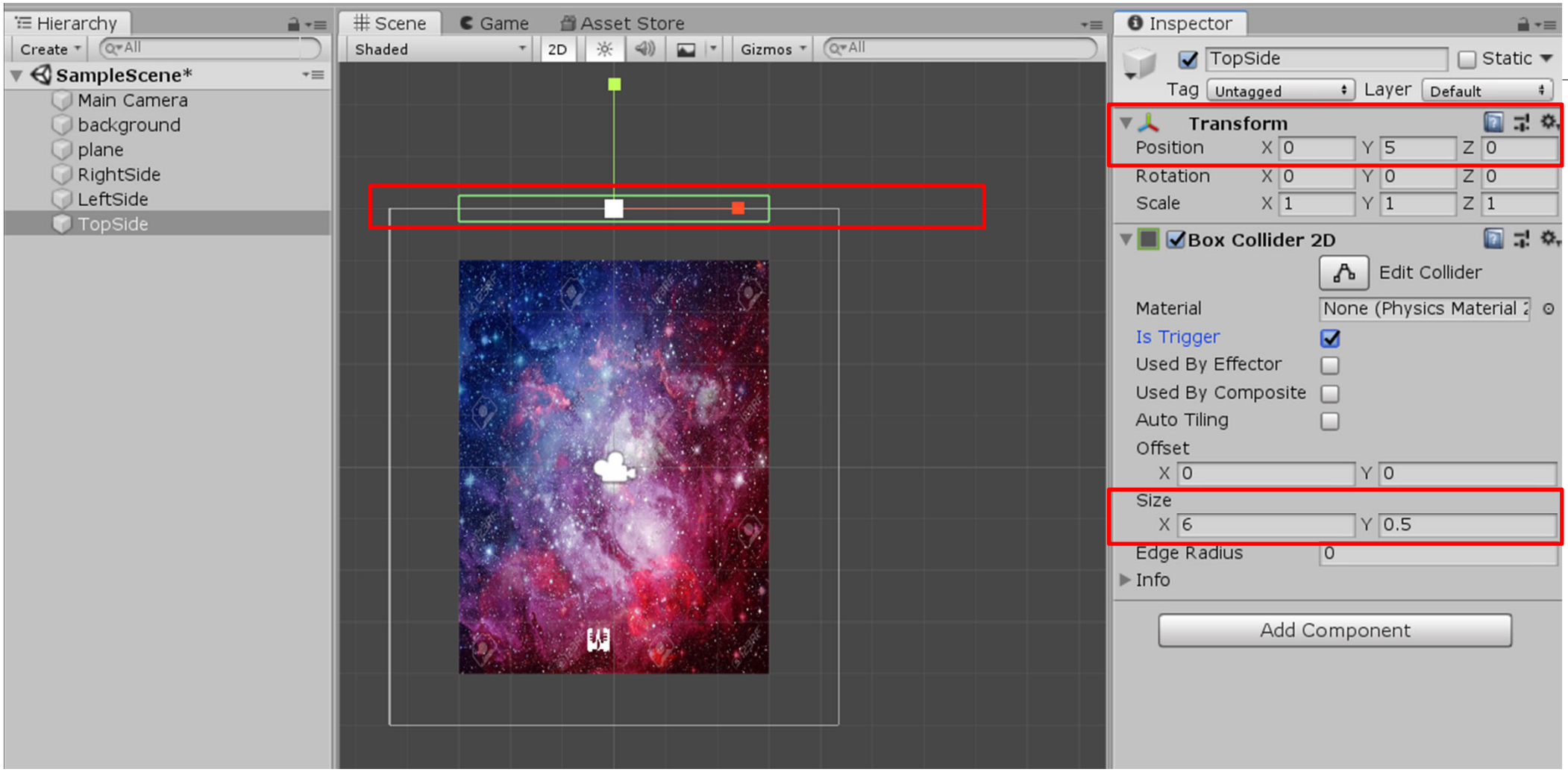
```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class ShipControl : MonoBehaviour
6  {
7      public GameObject Bullet; //宣告一個物件叫Bullet
8
9      // Start is called before the first frame update
10     void Start()
11     {
12
13     }
14
15     // Update is called once per frame
16     void Update()
17     {
18         if (Input.GetKey(KeyCode.RightArrow))
19         {
20             gameObject.transform.position += new Vector3(0.1f, 0, 0);
21         }
22
23         if (Input.GetKey(KeyCode.LeftArrow))
24         {
25             gameObject.transform.position += new Vector3(-0.1f, 0, 0);
26         }
27
28         if (Input.GetKeyDown(KeyCode.Space))
29         {
30             Vector3 pos = gameObject.transform.position + new Vector3(0, 0.6f, 0);
31
32             Instantiate(Bullet, pos, gameObject.transform.rotation);
33
34         }
35     }
36 }
37
```



目前位置Y軸+0.6 (往上移動)

複製子彈物件(Bullet)在目前位置

增加TopSide (上方牆壁)



增加TopSideFunction.cs (TopSide的碰撞偵測)

TopSideFunction.cs* | Laser.cs | ShipControl.cs

Assembly-CSharp

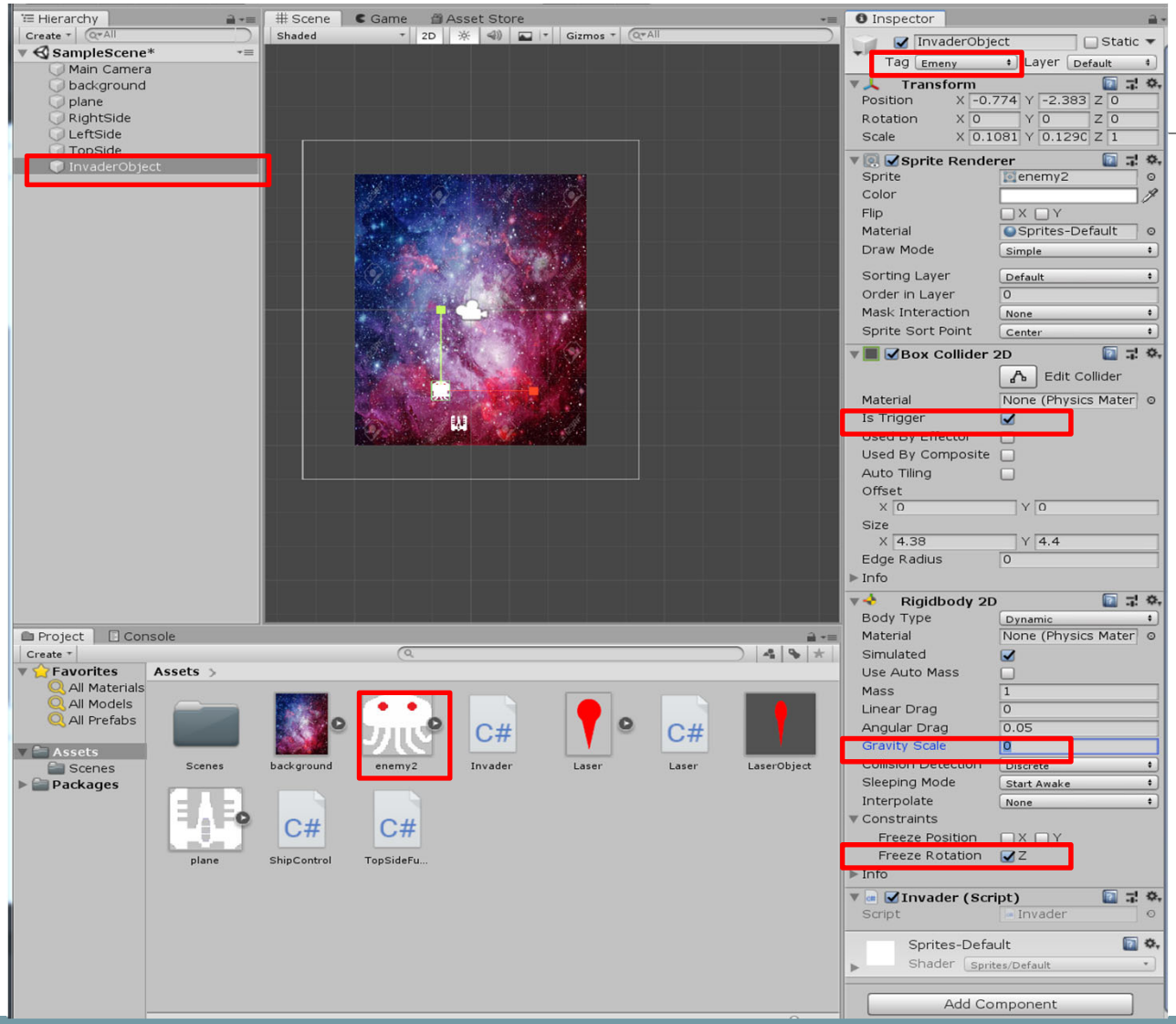
```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class TopSideFunction : MonoBehaviour
6  {
7      // Start is called before the first frame update
8      void Start()
9      {
10
11     }
12
13     // Update is called once per frame
14     void Update()
15     {
16
17     }
18
19     void OnTriggerEnter2D(Collider2D col)
20     {
21         if (col.tag == "Bullet")
22         {
23             Destroy(col.gameObject);
24         }
25     }
26 }
27
```

當偵測到碰撞到 Bullet這個tag的物件
(要先將LaserObject , add一個Bullet的tag)

刪除子彈物件(才不會一直產生子彈，浪費記憶體

建立敵人

建立敵人 (InvaderObject)



Invader.cs (控制敵人的程式)

Invader.cs* | TopSideFunction.cs | Laser.cs | ShipControl.cs

Assembly-CSharp | Invad

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   public class Invader : MonoBehaviour
6   {
7       // Start is called before the first frame update
8       void Start()
9       {
10
11      }
12
13      // Update is called once per frame
14      void Update()
15      {
16          gameObject.transform.position += new Vector3(0, -0.01f, 0);
17      }
18  }
19
```

緩慢往下移動 (Y軸)

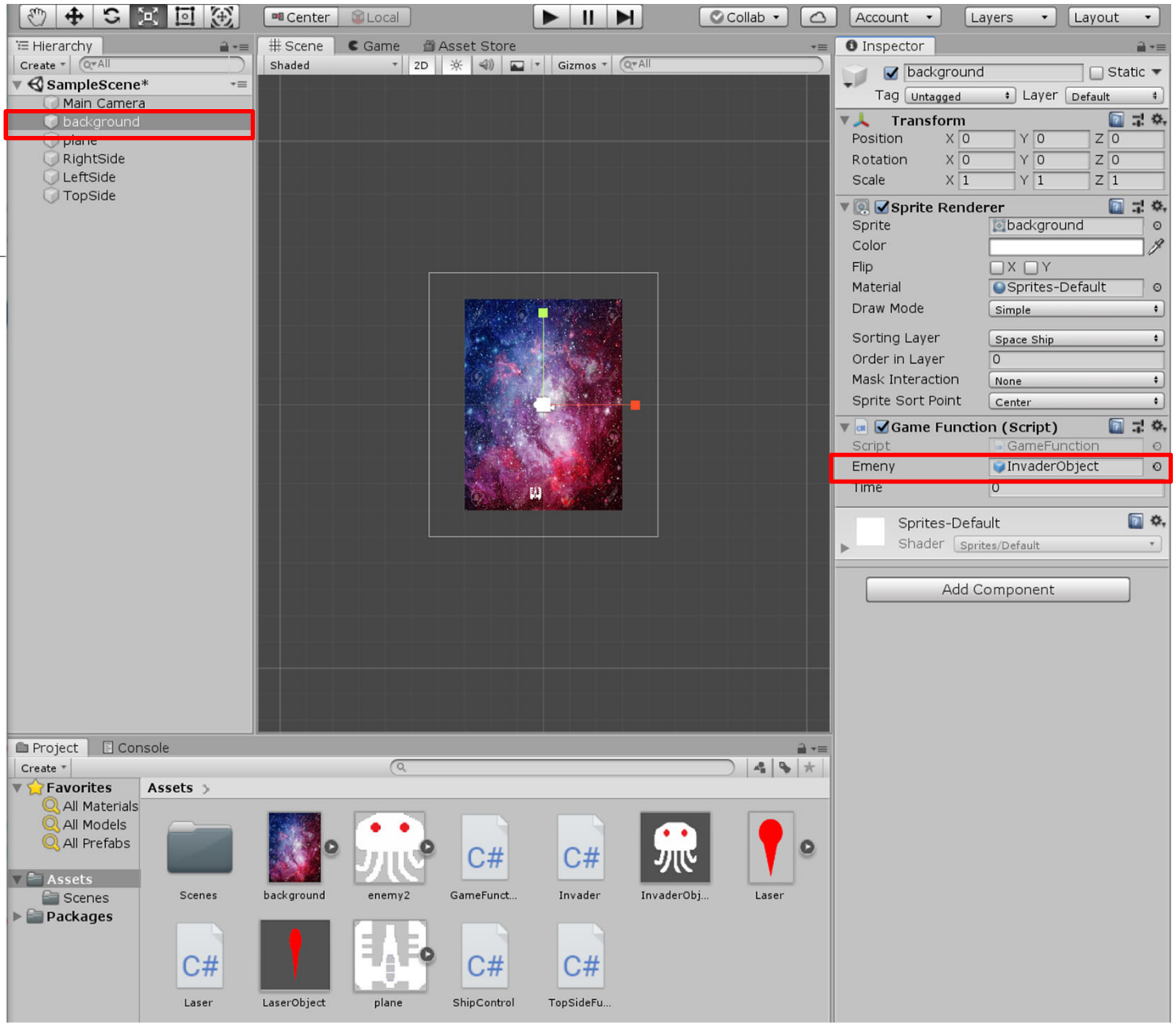
在Background物件新增GameFunction程式(產生敵人)

GameFunction.cs | Invader.cs | TopSideFunction.cs | Laser.cs | ShipControl.cs

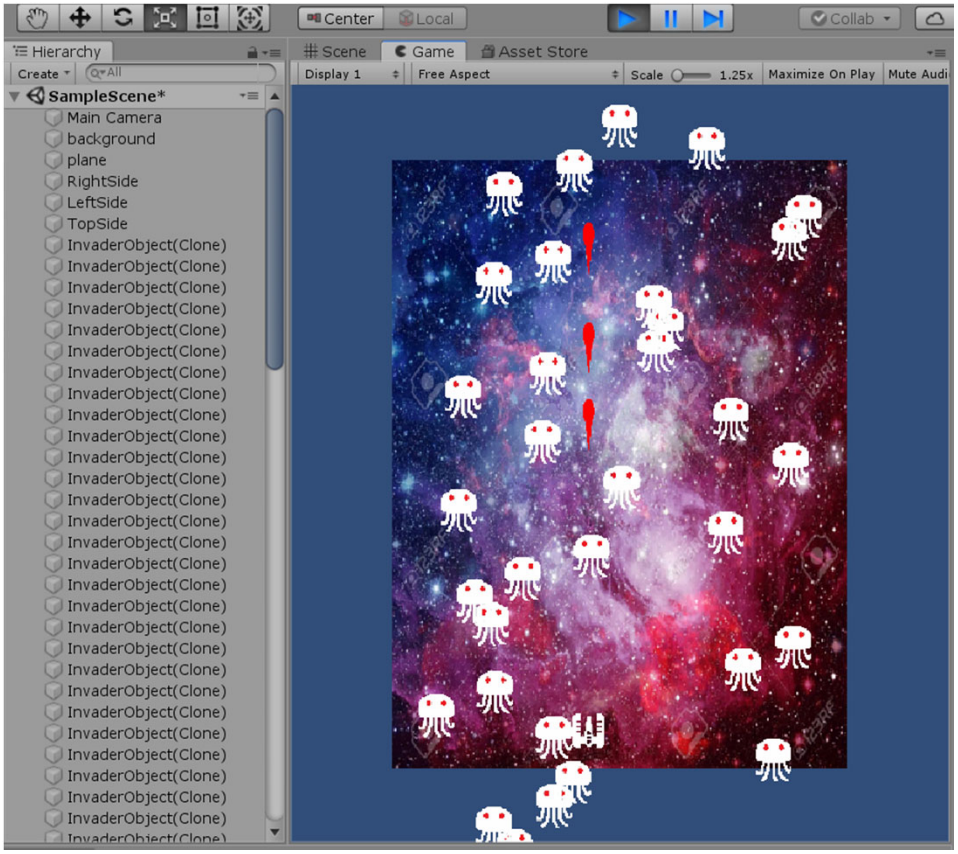
Assembly-CSharp | GameFunction

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class GameFunction : MonoBehaviour
6  {
7      public GameObject Emeny; //宣告物件，名稱Emeny
8      public float time; //宣告浮點數，名稱time
9
10     // Start is called before the first frame update
11     void Start()
12     {
13
14     }
15
16     // Update is called once per frame
17     void Update()
18     {
19         time += Time.deltaTime; //時間增加
20         if (time > 0.5f) //如果時間大於0.5(秒)
21         {
22             Vector3 pos = new Vector3(Random.Range(-2.5f, 2.5f), 4.5f, 0); //宣告位置pos，Random.Range(-2.5f,2.5f)代表X是2.5到-2.5之間隨機
23             Instantiate(Emeny, pos, transform.rotation);//產生敵人
24             time = 0f; //時間歸零
25         }
26     }
27 }
28
```

設定値


Hierarchy
Create
SampleScene*
Main Camera
background
RightSide
LeftSide
TopSide
Scene
Game
Asset Store
Shaded
2D
Gizmos
Inspector
background
Static
Tag
Untagged
Layer
Default
Transform
Position
Rotation
Scale
Sprite Renderer
Sprite
Color
Flip
Material
Sprites-Default
Draw Mode
Simple
Sorting Layer
Space Ship
Order in Layer
Mask Interaction
None
Sprite Sort Point
Center
Game Function (Script)
Script
GameFunction
Emeny
InvaderObject
Time
Shader
Sprites/Default
Add Component
Project
Console
Favorites
All Materials
All Models
All Prefabs
Assets
Scenes
Packages
Laser
LaserObject
plane
ShipControl
TopSideFu...

目前完成狀態



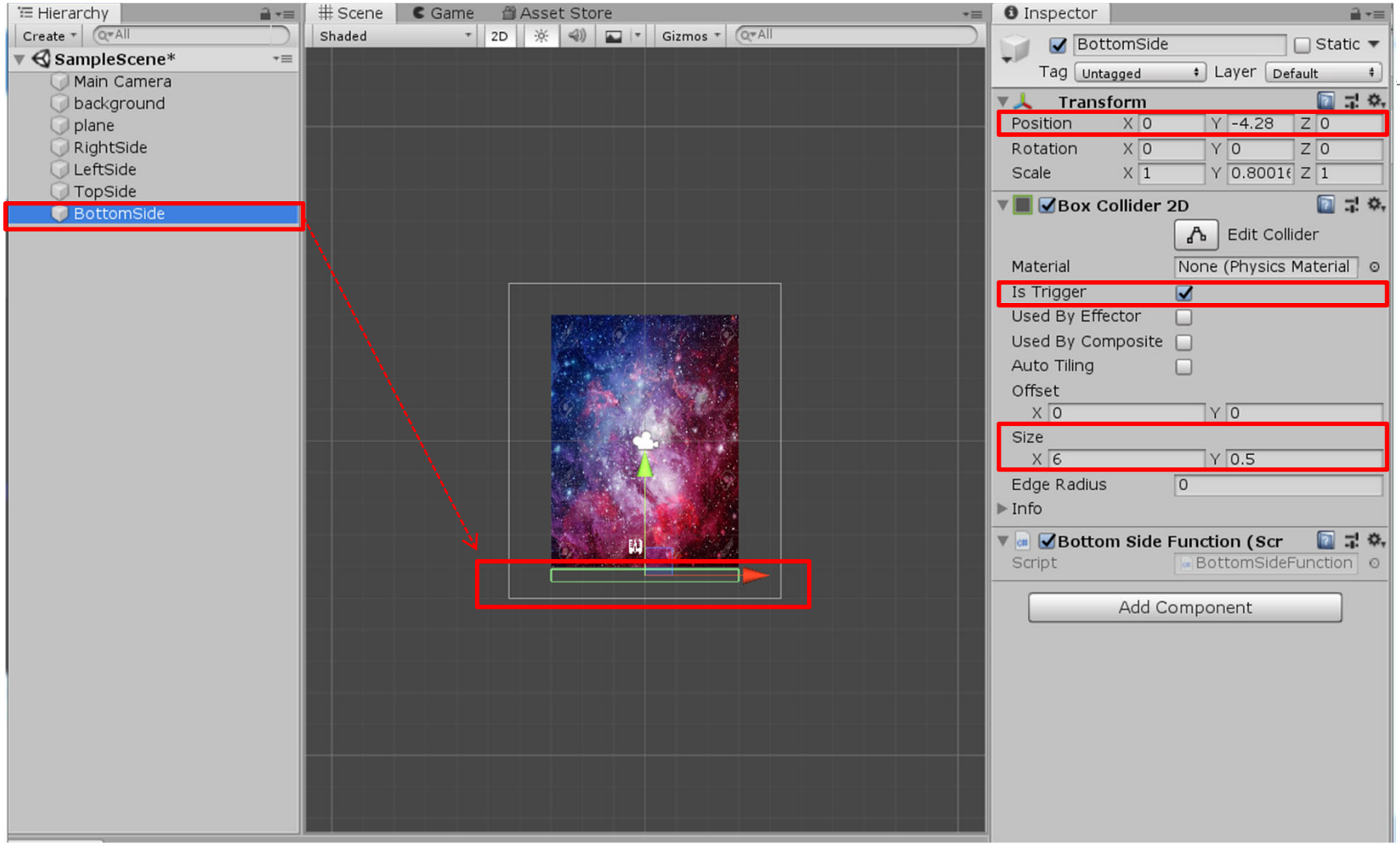
應該會有很多敵人產生

子彈射出到最上方碰到Topside應該會消滅

Try it!

1. 讓敵人可以左右移動
2. 讓敵人可以隨機移動
3. 讓敵人移動速度加快

建立BottomSide物件與設定值



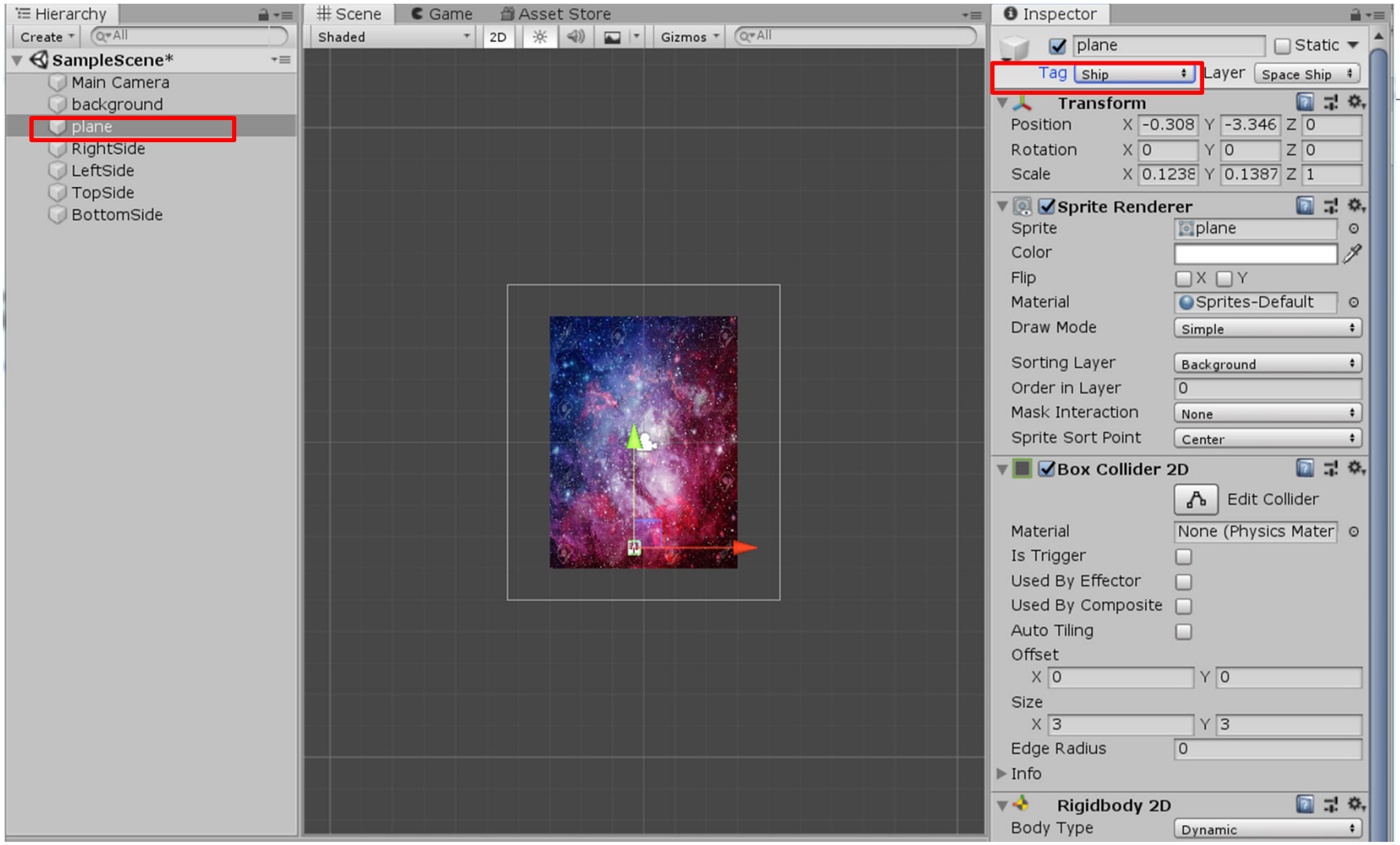
加入BottomSideFunction程式在BottomSide物件上

BottomSideFunction.cs | GameFunction.cs | Invader.cs | TopSideF

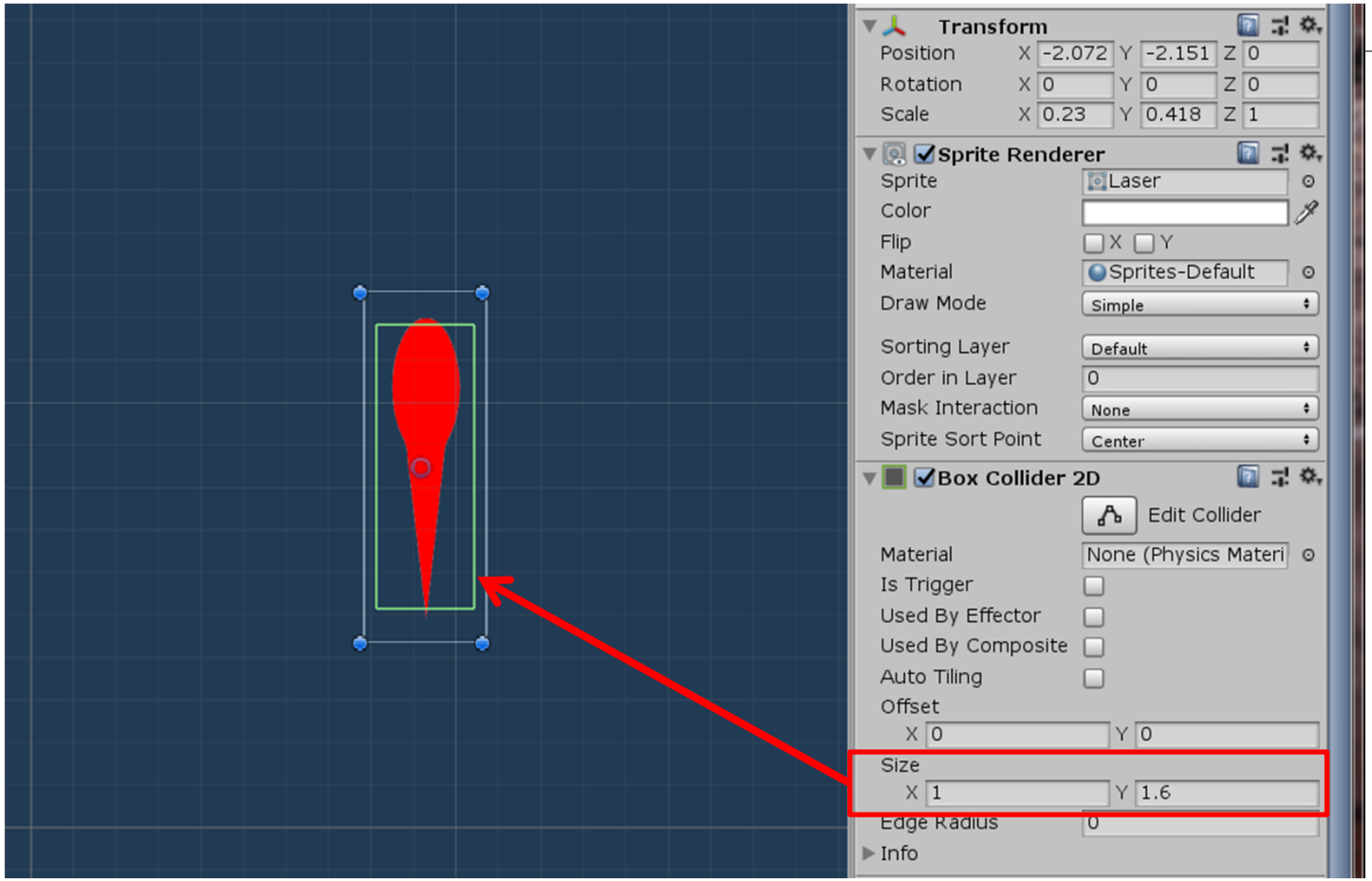
Assembly-CSharp

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   public class BottomSideFunction : MonoBehaviour
6   {
7       // Start is called before the first frame update
8       void Start()
9       {
10
11      }
12
13      // Update is called once per frame
14      void Update()
15      {
16
17      }
18
19      void OnTriggerEnter2D(Collider2D col) //碰撞事件
20      {
21          if (col.tag == "Enemy") //如果標籤是Enemy
22          {
23              Destroy(col.gameObject); //消滅碰撞的物件
24          }
25      }
26  }
27
```

Plane飛機物件記得加上Ship標籤



注意: 子彈LaserObject的Box Collider 2D範圍要設的小一點。避免太空船發射後會往後退 (因太空船與子彈發生碰撞)



Invader.cs 增加消滅敵人與飛機 (在InvaderObject裡面修改程式)

BottomSideFunction.cs | GameFunction.cs | Invader.cs* | TopSideFunction.cs | Laser.cs

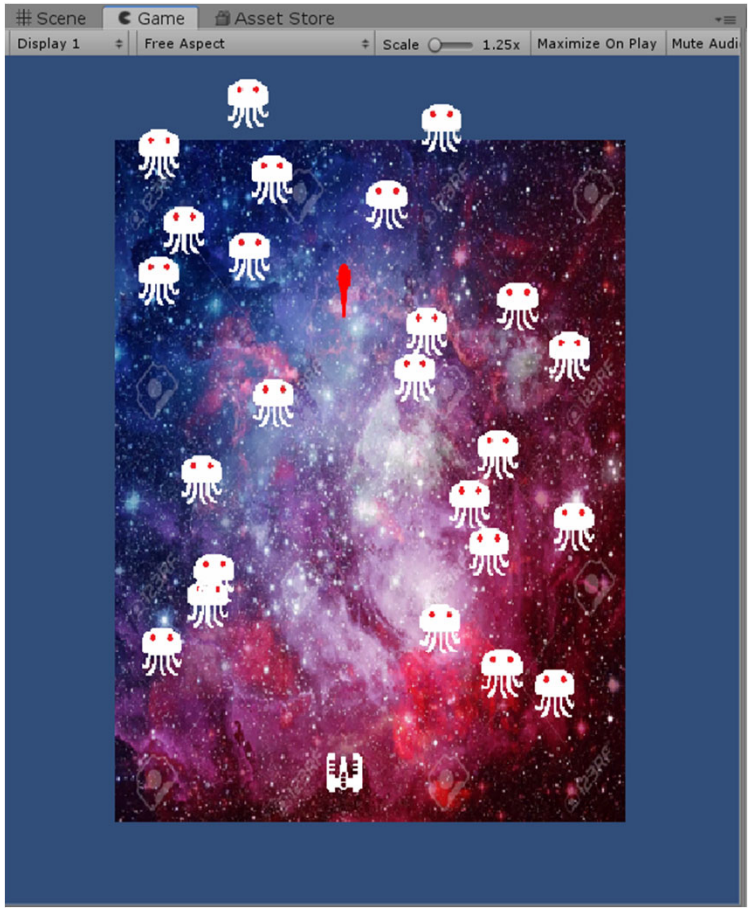
Assembly-CSharp | Invader

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   public class Invader : MonoBehaviour
6   {
7       // Start is called before the first frame update
8       void Start()
9       {
10
11      }
12
13      // Update is called once per frame
14      void Update()
15      {
16          gameObject.transform.position += new Vector3(0, -0.01f, 0);
17      }
18
19      void OnTriggerEnter2D(Collider2D col) //名為col的觸發事件
20      {
21          if (col.tag == "Ship" || col.tag == "Bullet") //如果碰撞的標籤是Ship或Bullet
22          {
23              Destroy(col.gameObject); //消滅被碰撞的物件
24              Destroy(gameObject); //消滅物件本身
25          }
26      }
27  }
28
```

現在應該已經可以玩了

敵人會被子彈射死

飛機碰到敵人會消失

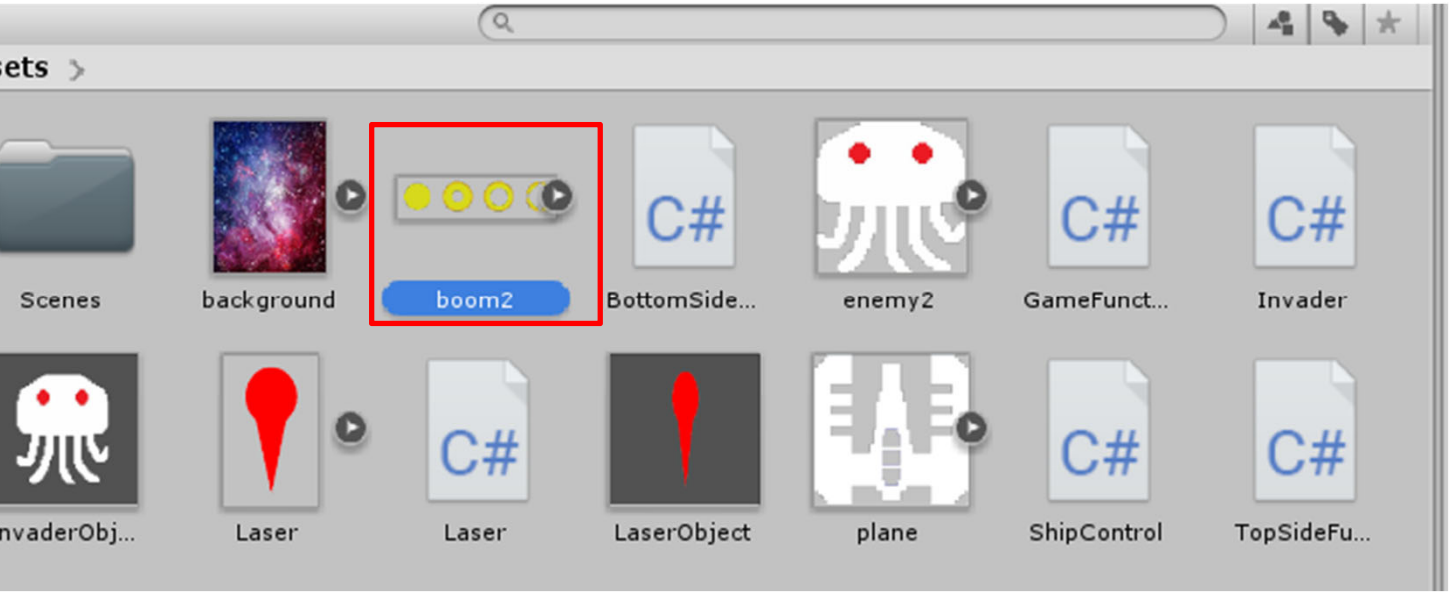


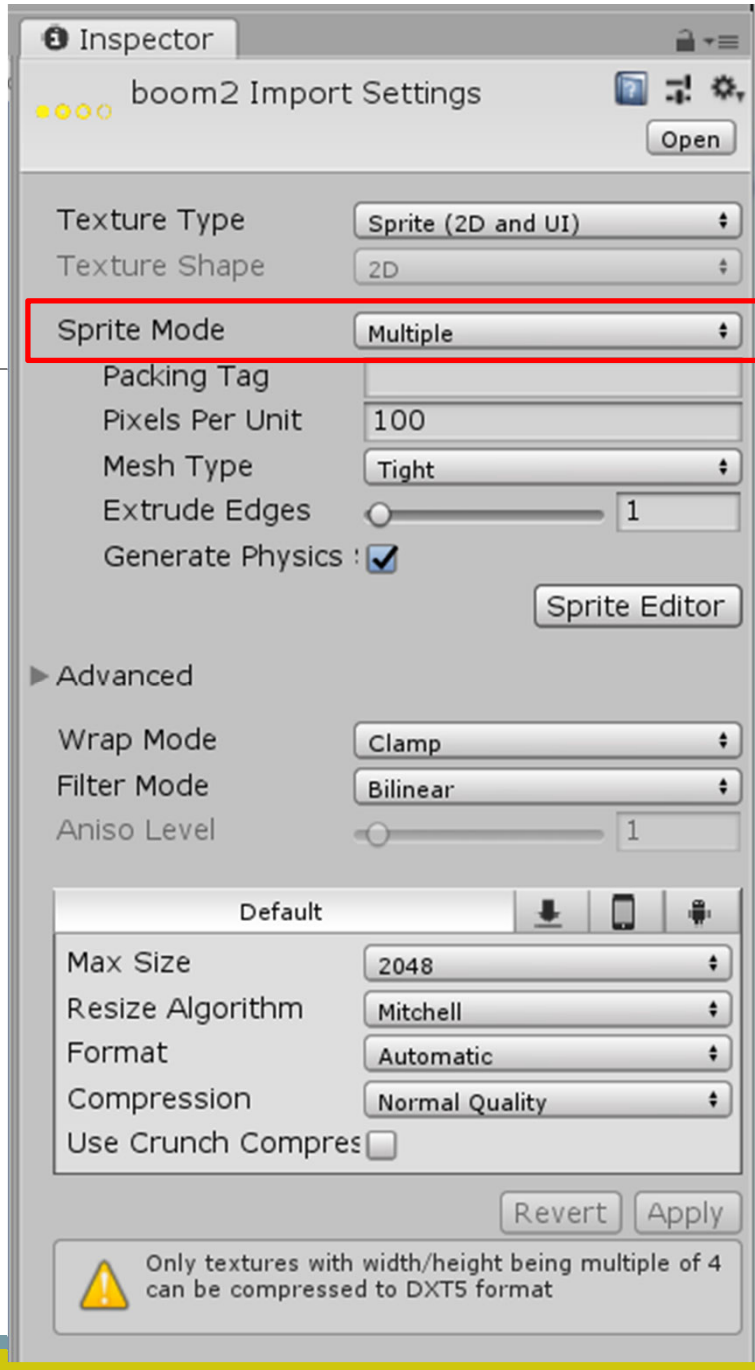
Part III 動畫、音效

動畫匯入與處理

(Step 1) 匯入boom爆炸動畫-> Rename Explode

(Step 2) Sprite Mode選Multiple (因為要多張動畫)





匯入動畫

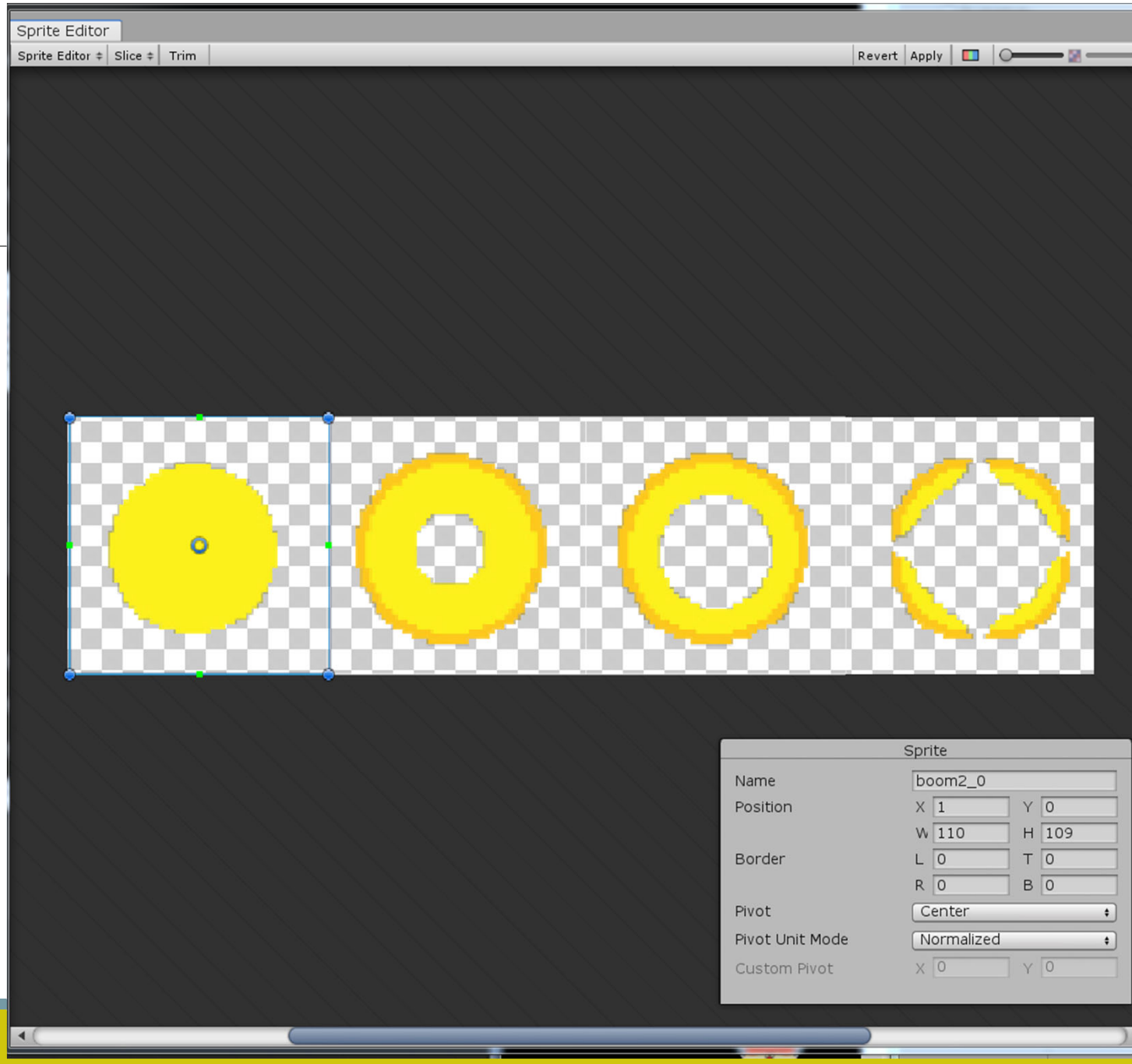
按下Spirte Editor

Slice->Type->Grid by Size

Size: X:108, Y: 109

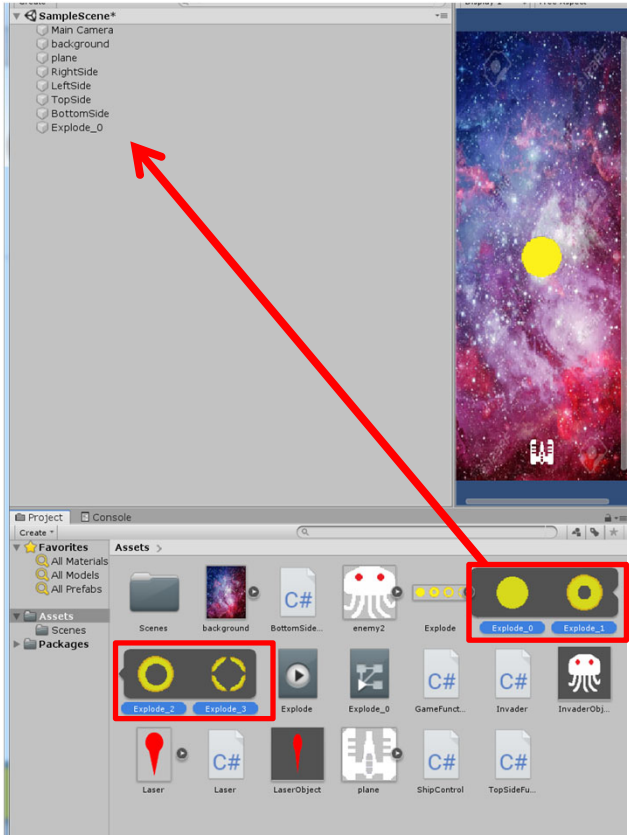
調整X，Y讓動畫剛好切四個

像下面這樣



把動畫Explode拉到場景

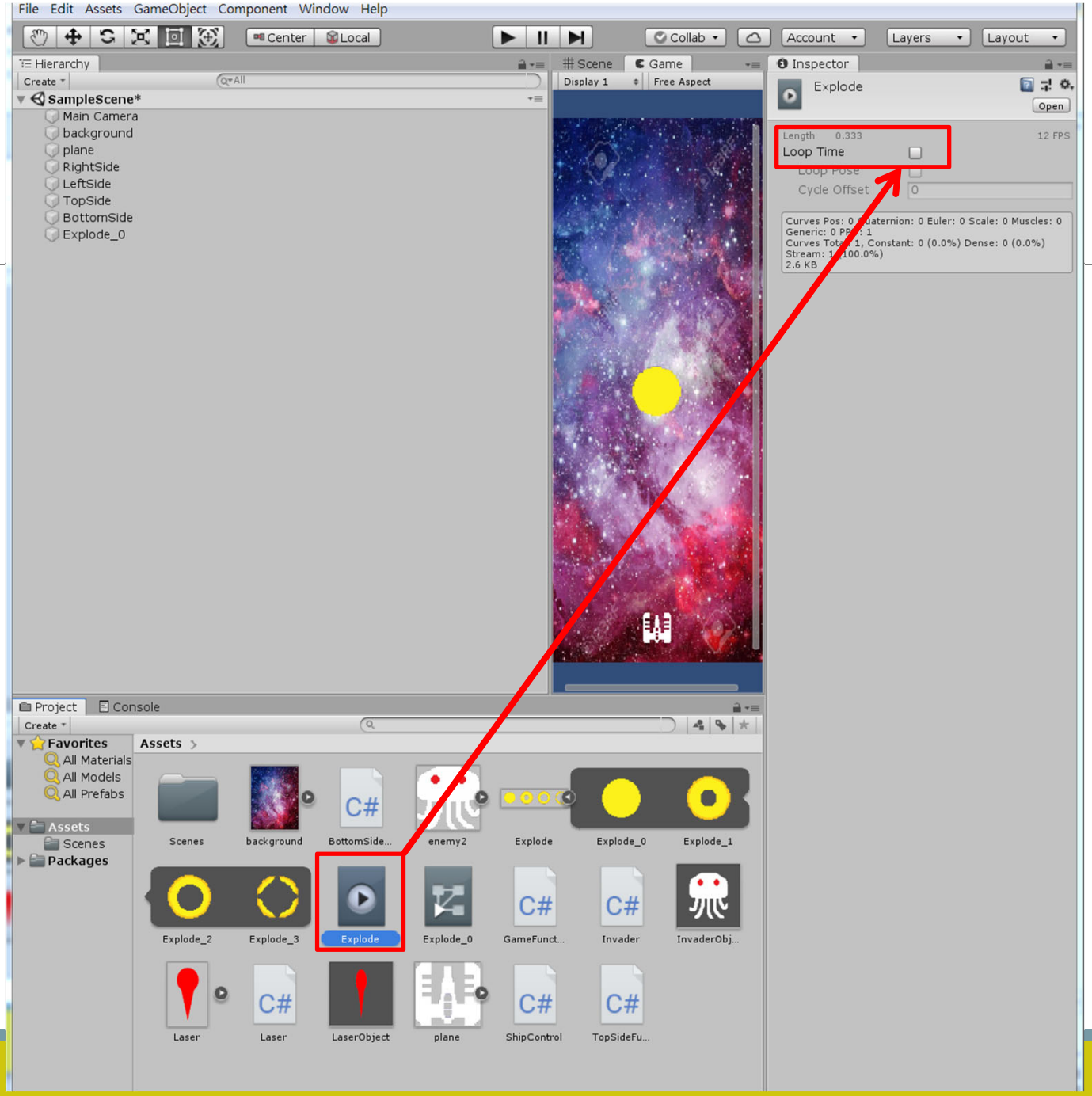
會產生三個物件。(要拖曳動畫才對)



設定動畫

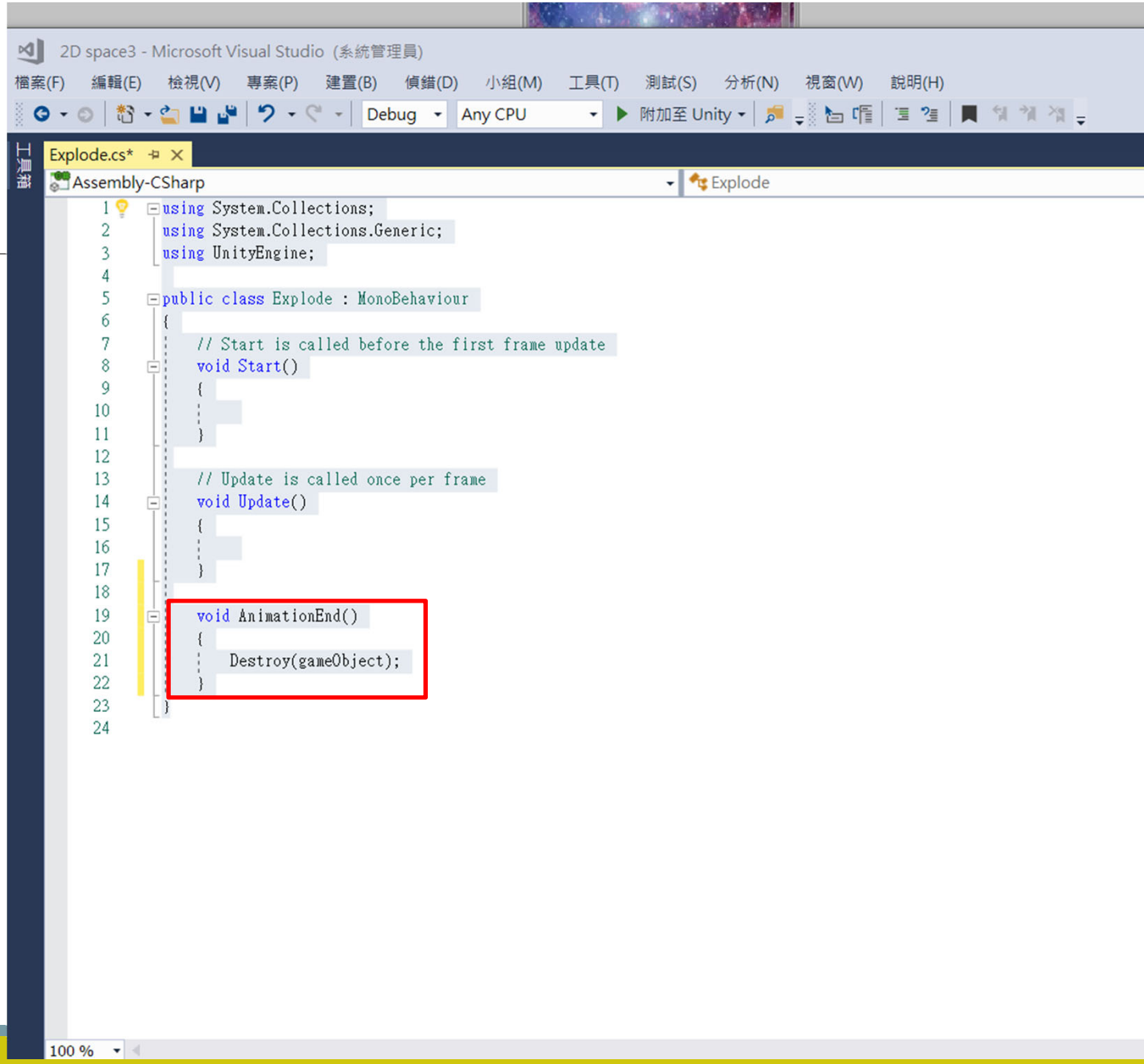
點選Explode，

取消Loop Time

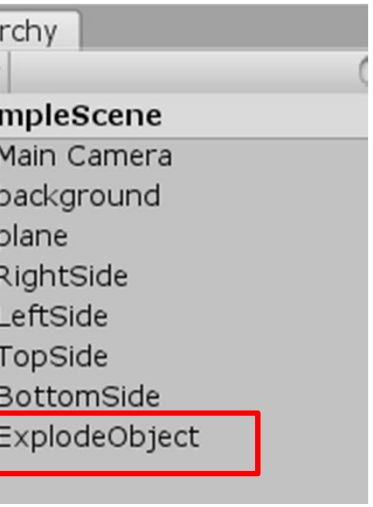


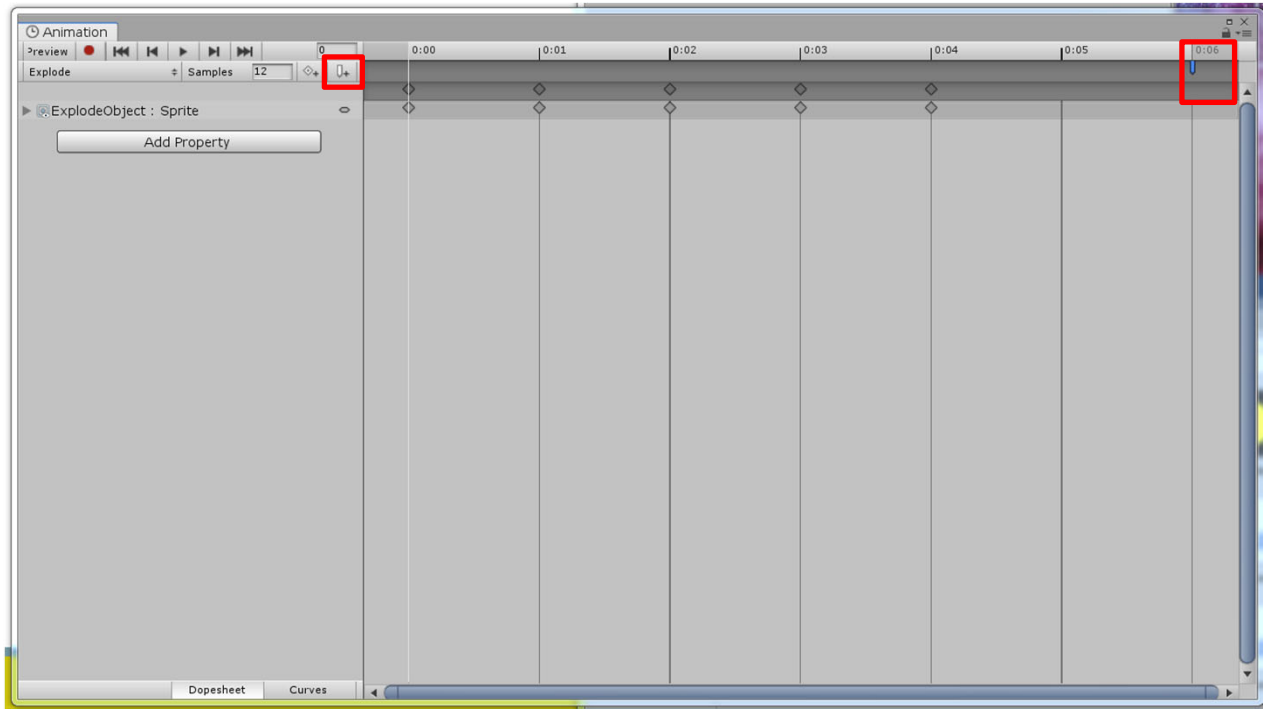
在Explode新增程式

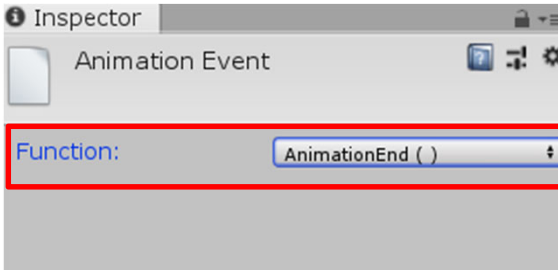
- Explode.cs
- 新增程式



選Explode Object -> Windows Animation
Add Event->拖到第五格
Inspector，設定Function-> AnimationEnd()







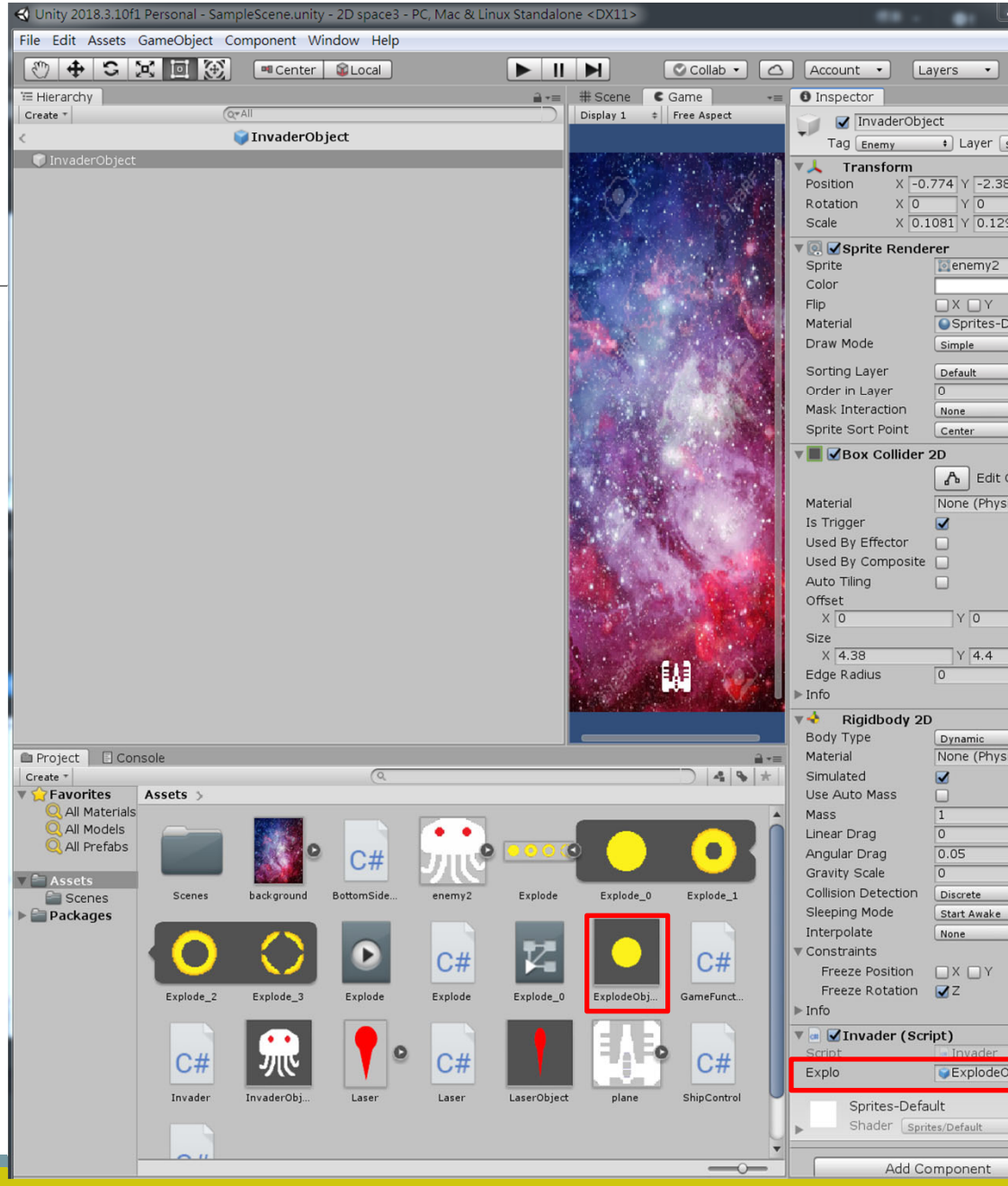
修改Invader.cs

Invader.cs | Explode.cs

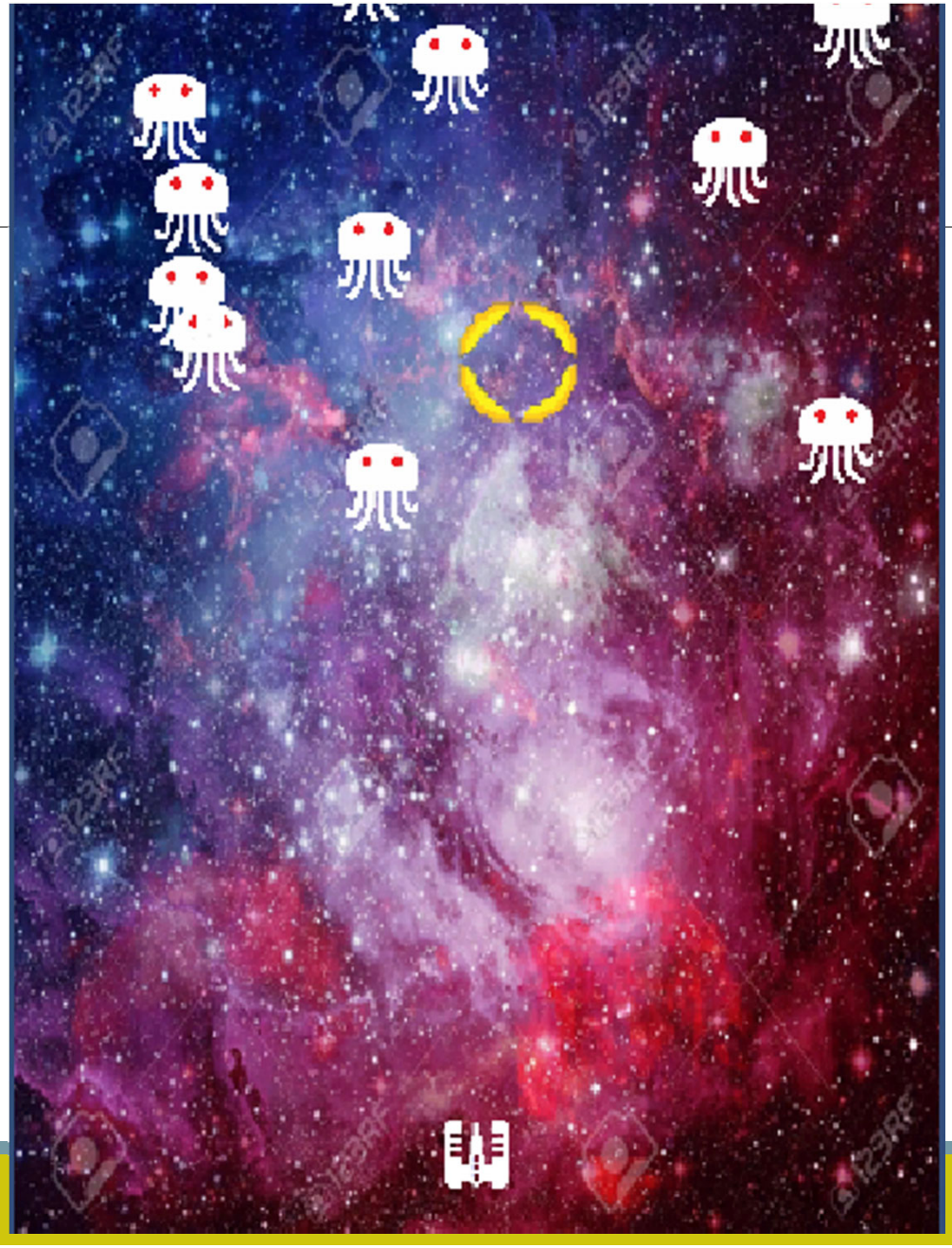
Assembly-CSharp | Invader

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   public class Invader : MonoBehaviour
6   {
7       public GameObject explo; // 宣告一個名為explo的物件
8
9       // Start is called before the first frame update
10      void Start()
11      {
12
13      }
14
15      // Update is called once per frame
16      void Update()
17      {
18          gameObject.transform.position += new Vector3(0, -0.01f, 0);
19      }
20
21      void OnTriggerEnter2D(Collider2D col) //名為col的觸發事件
22      {
23          if (col.tag == "Ship" || col.tag == "Bullet") //如果碰撞的標籤是Ship或Bullet
24          {
25              Destroy(col.gameObject); //消滅被碰撞的物件
26              Destroy(gameObject); //消滅物件本身
27
28              Instantiate(explo, transform.position, transform.rotation); //在外星人的位置產生爆炸
29              if (col.tag == "Ship")
30              {
31                  Instantiate(explo, col.gameObject.transform.position, col.gameObject.transform.rotation);
32                  //在碰撞物件的位置產生爆炸，也就是在太空船的位置產生爆炸
33              }
34          }
35      }
36  }
37
```

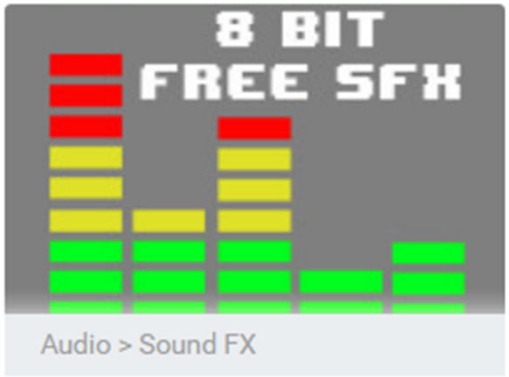
把ExplodeObj拉到InvderObj的Explo欄位



可以呈現爆炸效果了



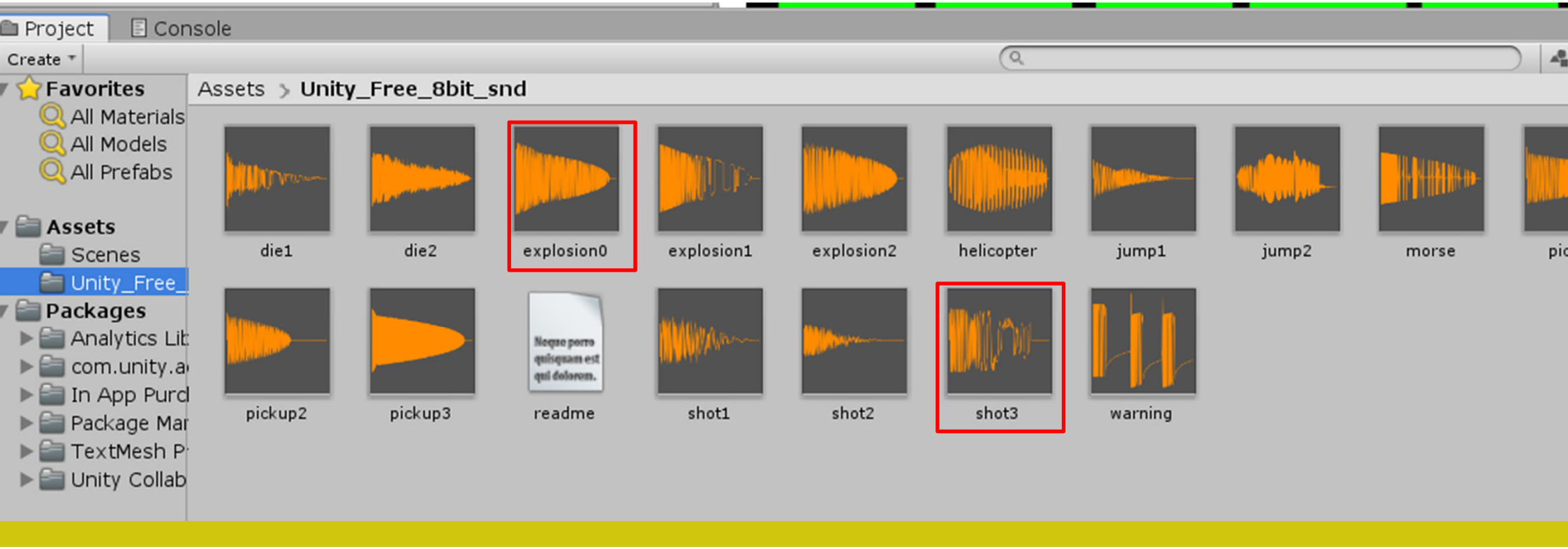
音效
至Asset Store下載



Audio -> Sound SFX

匯入後，在Assets底下會有一個Unity_Free資料夾

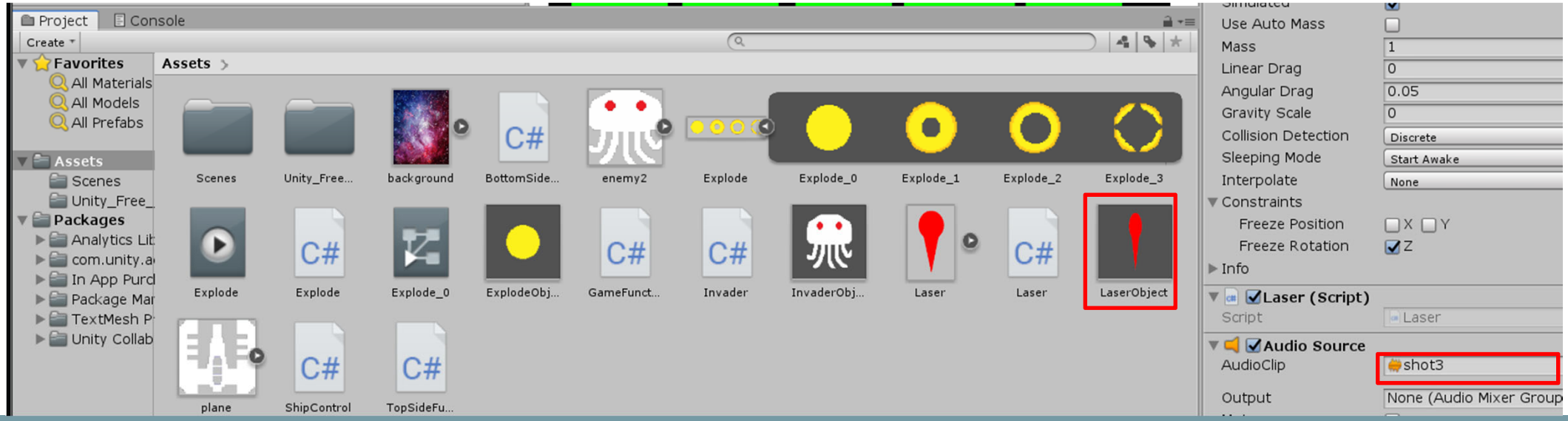
保留以下音效，其他刪除



替子彈加入音效

LaserObject -> Add Componet -> Audio -> Audio Source

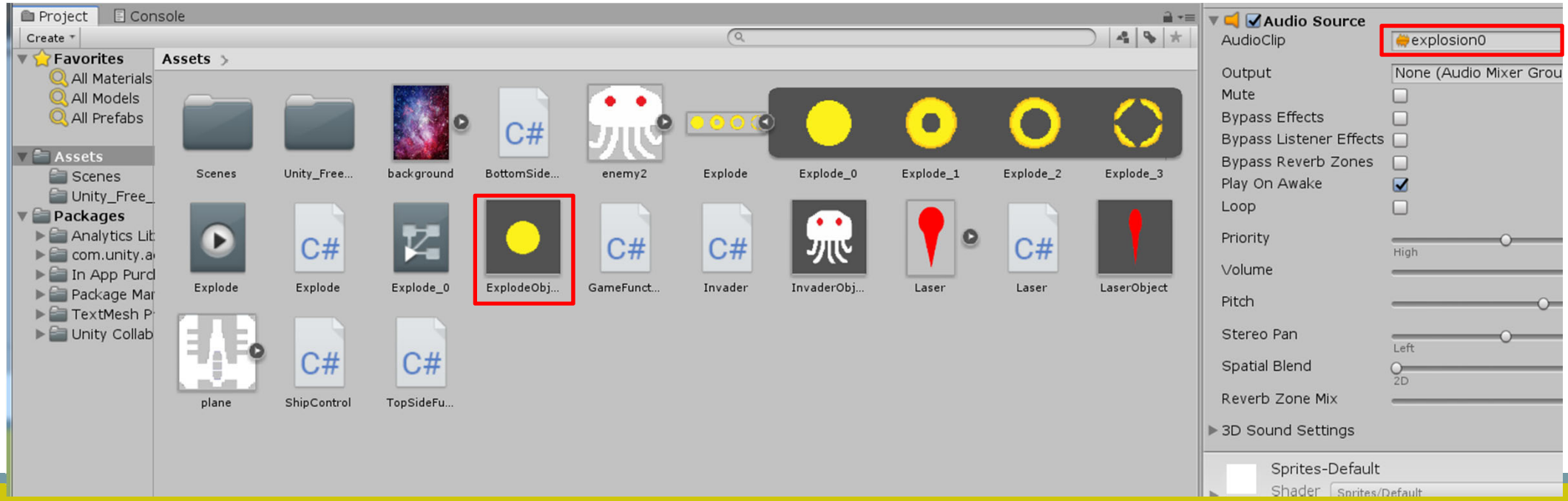
把shot3 拉到Audio Source ->Audio Clip



替爆炸加入音效

選ExplodeObj -> Add Component -> Audio Source

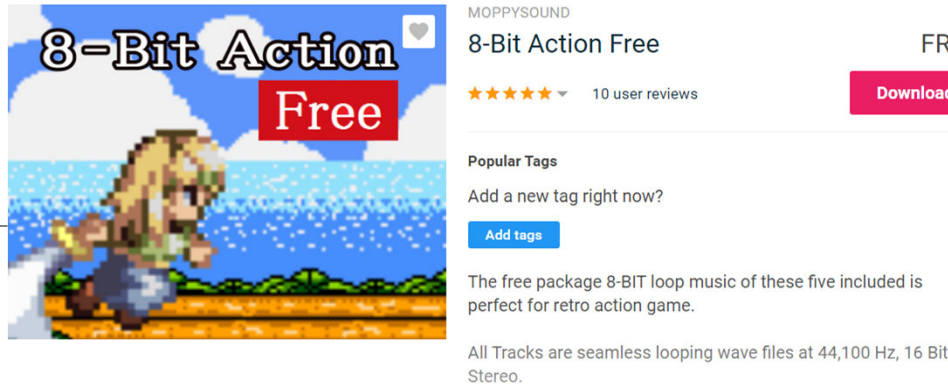
把Explosion0音效拉到 AudioClip



背景音效

Asset Store

Music-> 8-bit Action Free



添加背景音樂

Background

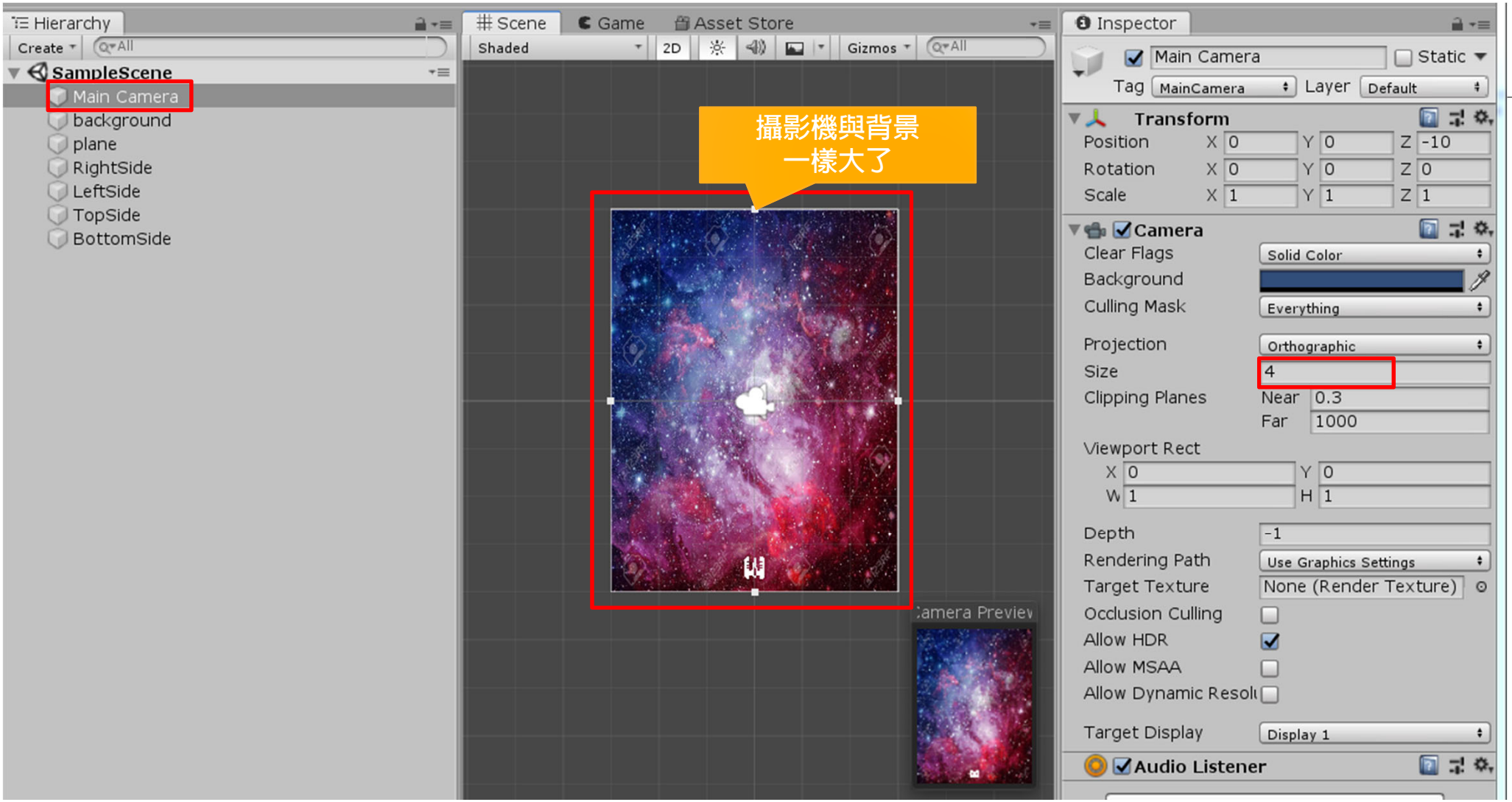
->Add Component

->Audio Source

把音樂拖曳到Audio Source

UI 設定

設定攝影機Main Camera

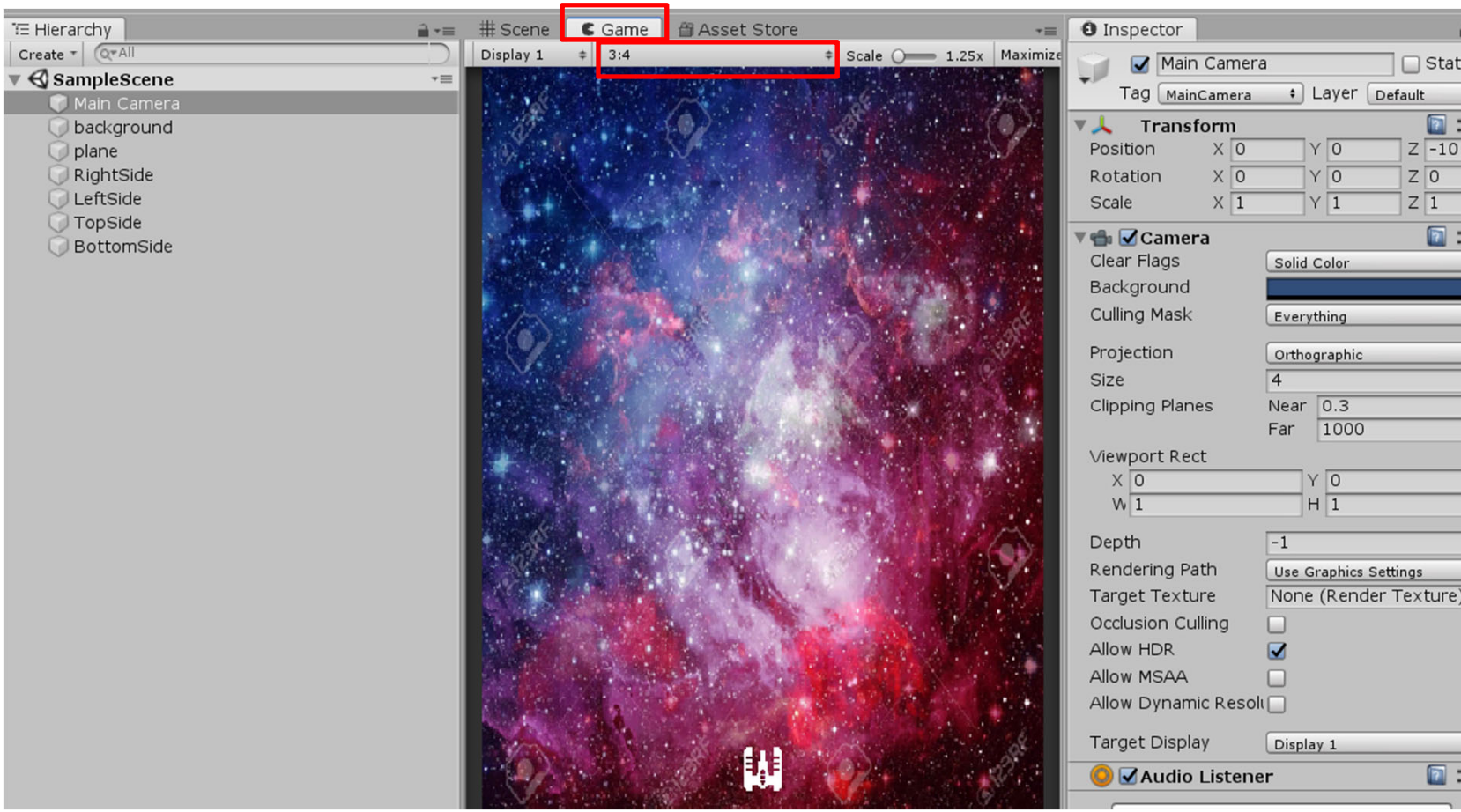


設定攝影機

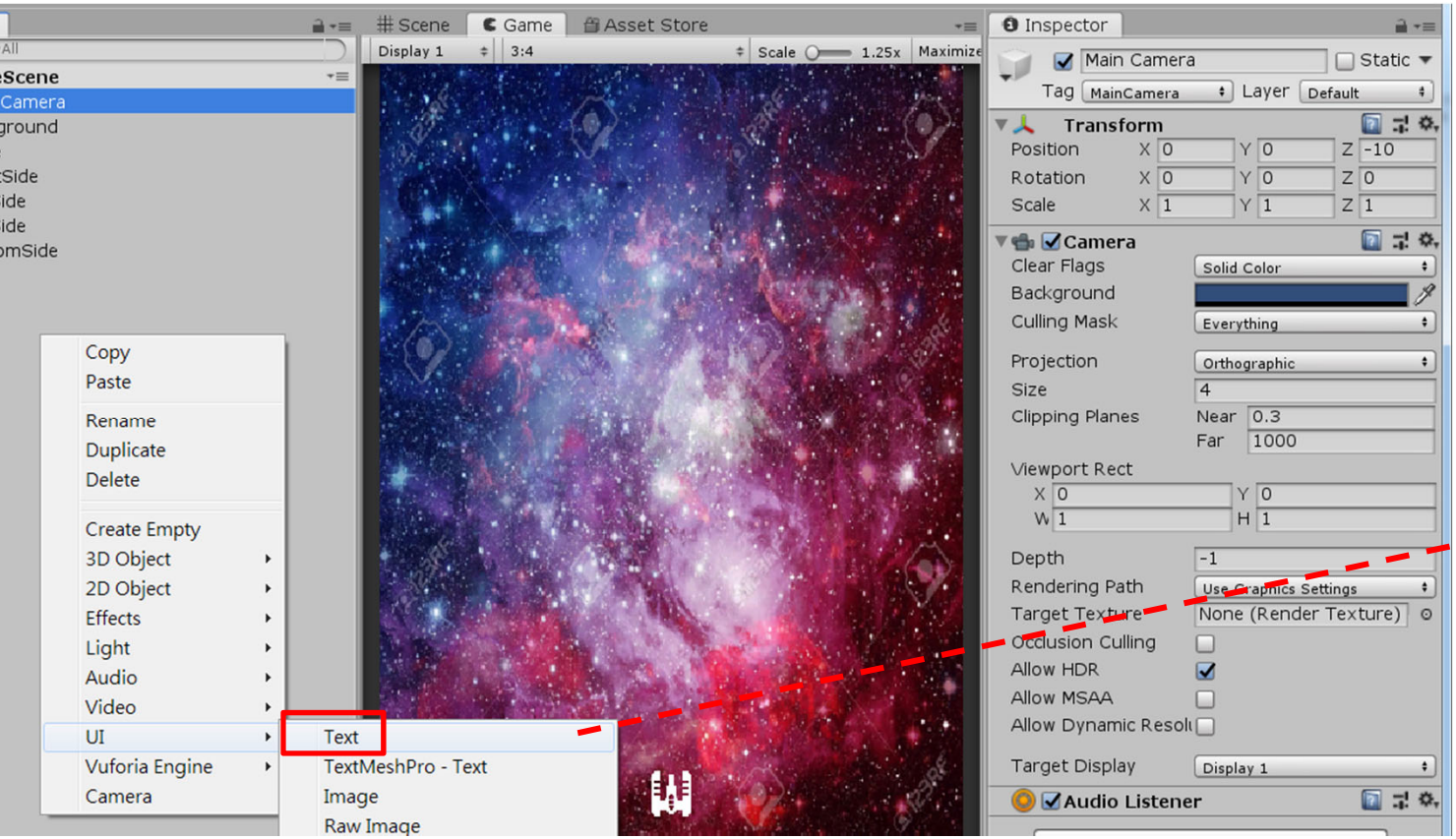
選Game Windows

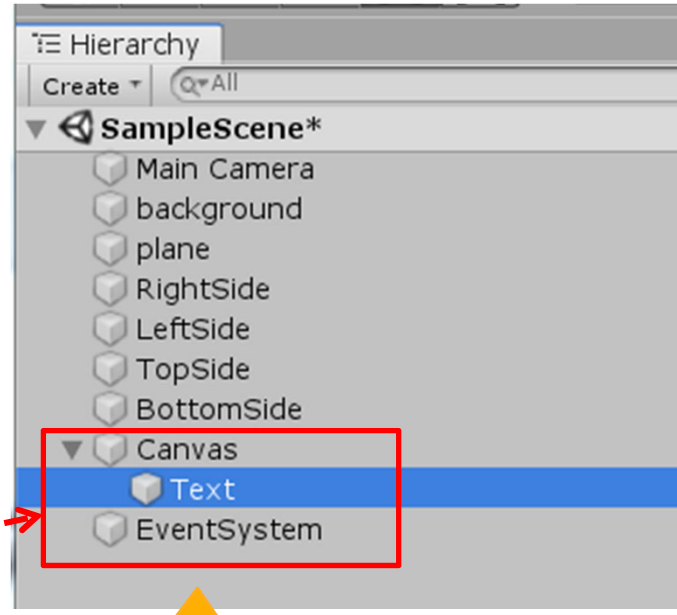
Free Aspect

自訂 3:4的比例



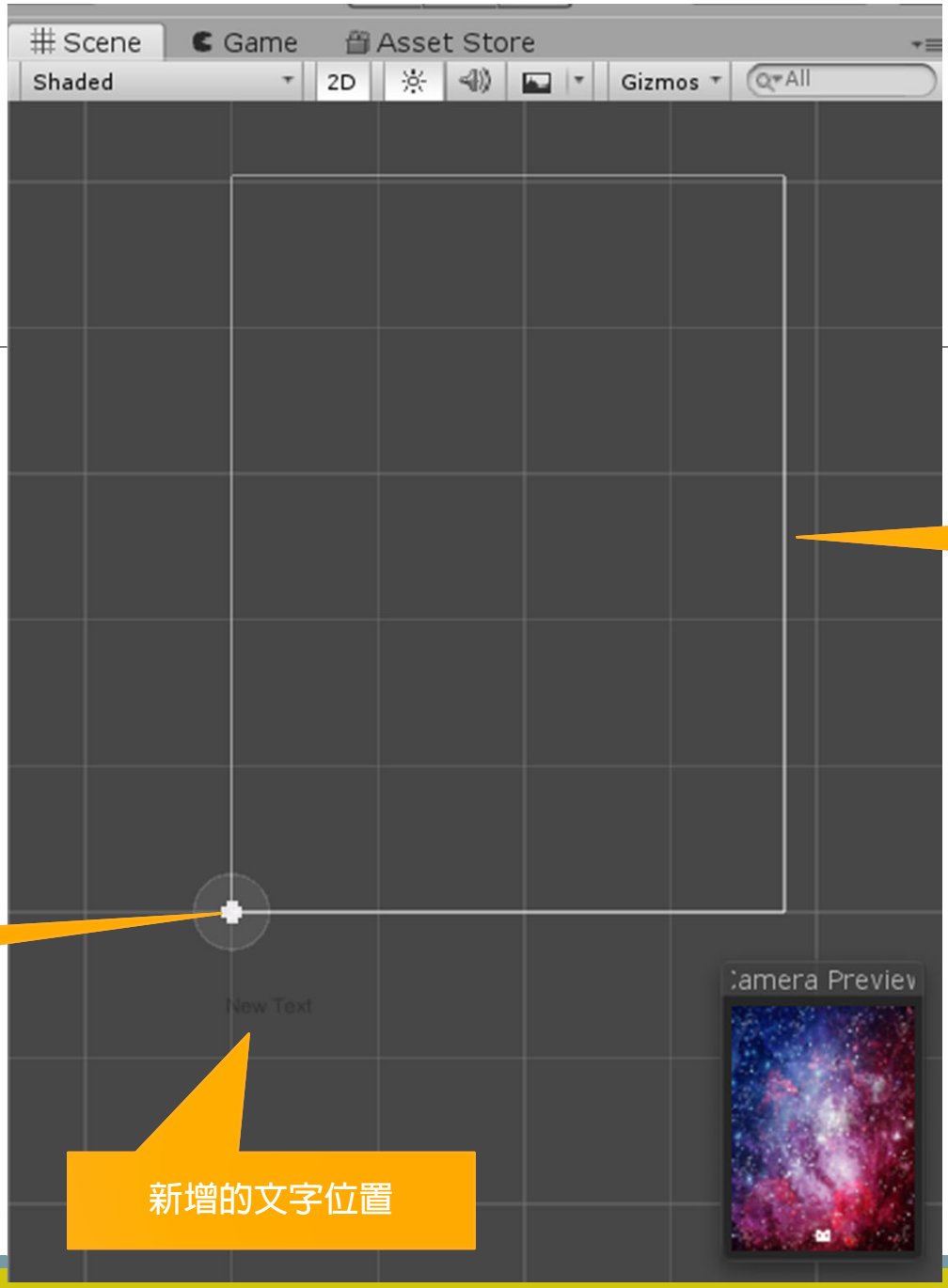
新增UI
Hierarchy裡面按滑鼠右鍵
UI->Text





有產生Canvas, Text, EventSystem才可以

Canvas大小


Scene
Game
Asset Store
Shaded
2D
Gizmos
All
Canvas大小
遠大於遊戲畫面
場景畫面大小
New Text
新增的文字位置
Camera Preview

文字設定

文字改為ScoreText

字內容:Score: 0

定文字顏色

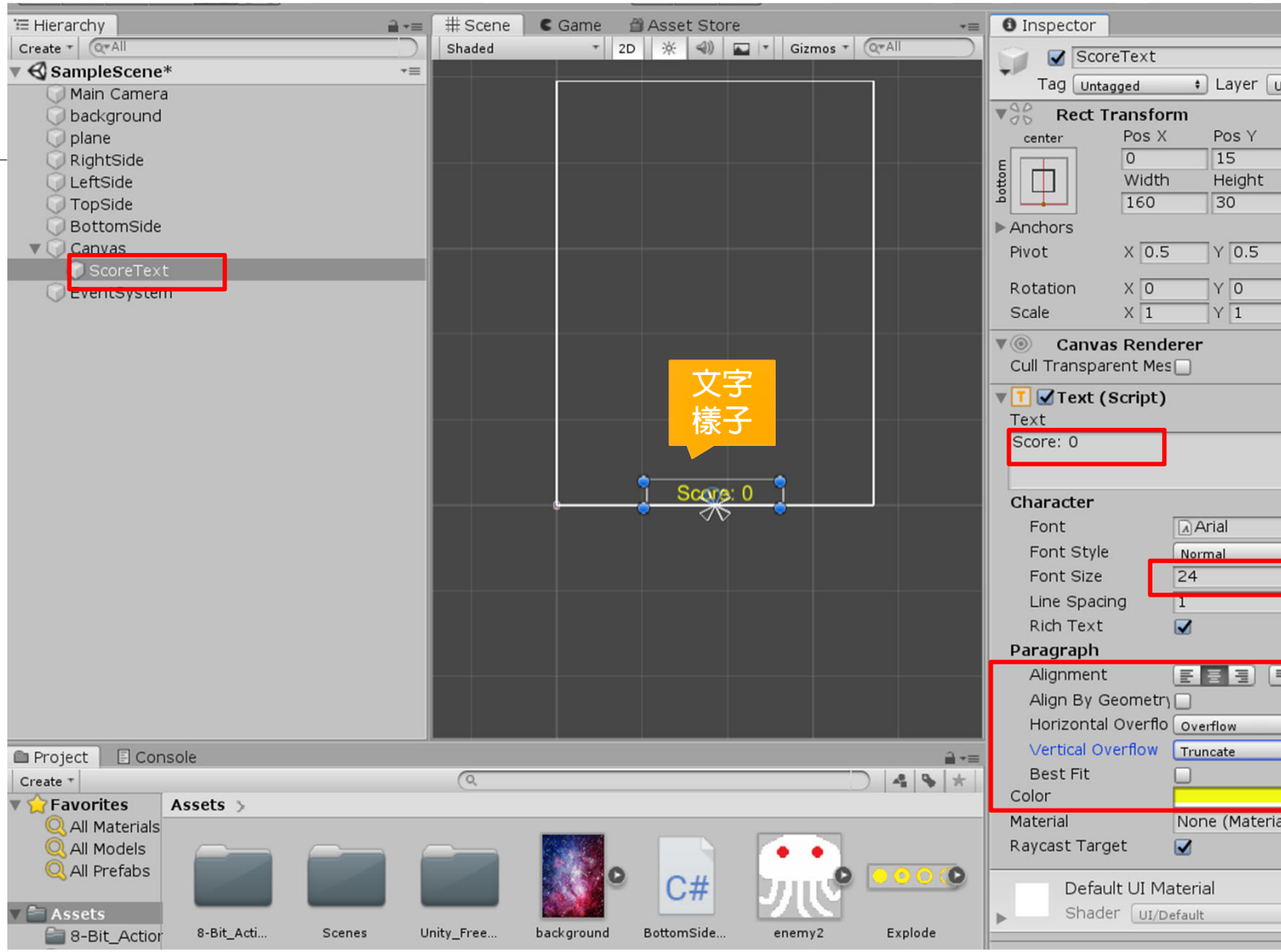
小:24

chors: 設定中間下方

gnment: 置中

rizontal Overflow: Overflow

rtical Overflow: Truncate



修改GameFunction.cs 讓他可以更改分數

GameFunction.cs* | Invader.cs | Explode.cs

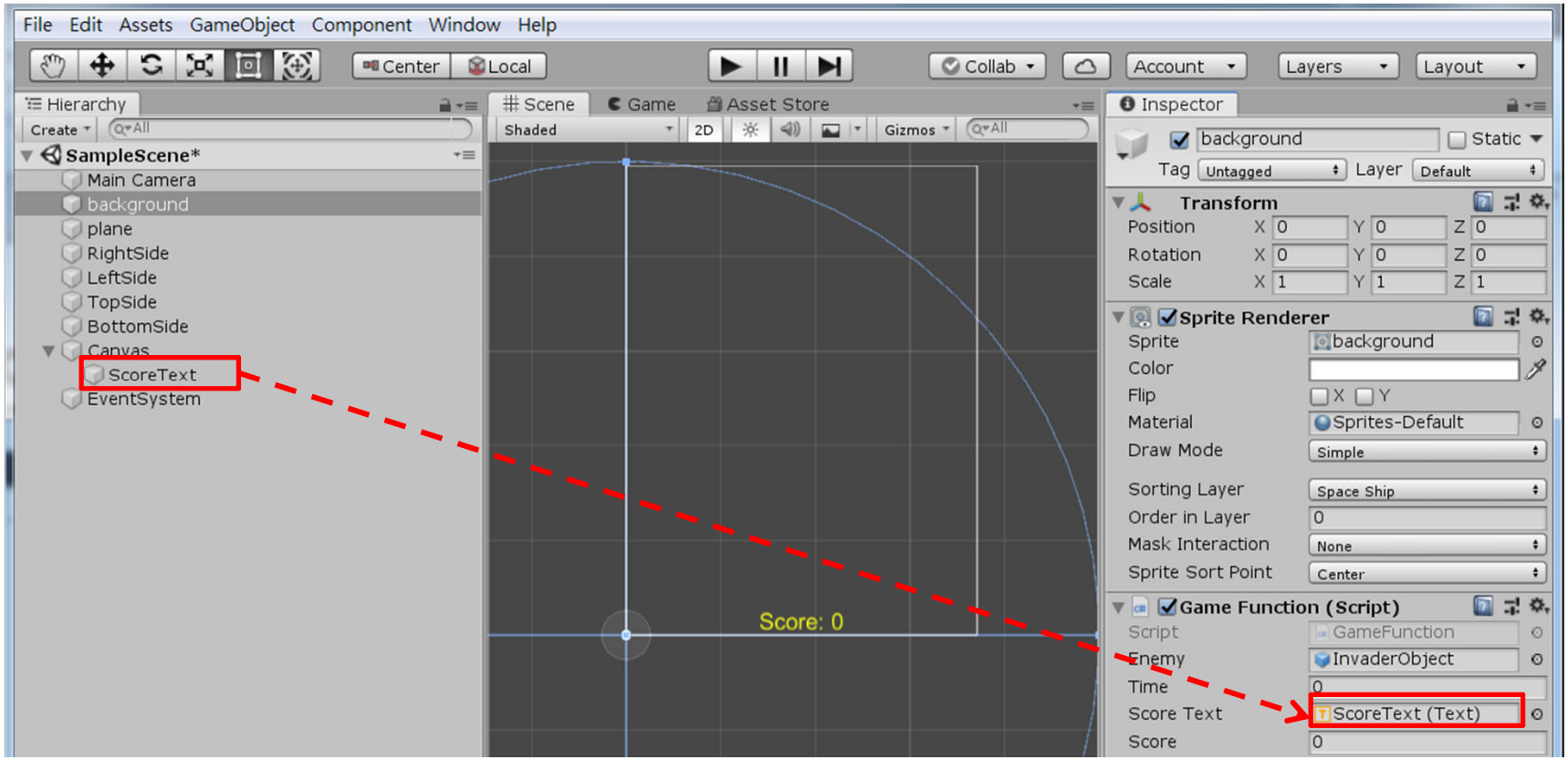
Assembly-CSharp | GameFunction | AddSc

```
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   using UnityEngine.UI;  //使用UI
6
7   public class GameFunction : MonoBehaviour
8   {
9       public GameObject Enemy; //宣告物件，名稱Enemy
10      public float time; //宣告浮點數，名稱time
11
12      public Text ScoreText;                  // 宣告一個文字型態的變數 ScoreText
13      public int Score = 0;                   // 宣告一個整數Score
14      public static GameFunction Instance;    // 設定Instance，讓其他程式能讀取GameFunction裡的東西
15
16      // Start is called before the first frame update
17      void Start()
18      {
19          Instance = this;    // 指定Instance參考這個程式
20      }
21
22      // Update is called once per frame
23      void Update()
24      {
25          time += Time.deltaTime; //時間增加
26          if (time > 0.5f) //如果時間大於0.5(秒)
27          {
28              Vector3 pos = new Vector3(Random.Range(-2.5f, 2.5f), 4.5f, 0); //宣告位置pos，Random.Range(-2.5f,2.5f)代表X是2.5到-2.5之間隨機
29              Instantiate(Enemy, pos, transform.rotation);//產生敵人
30              time = 0f; //時間歸零
31          }
32      }
33
34      public void AddScore()
35      {
36          Score += 10;        //分數+10分
37          ScoreText.text = "Score: " + Score;     //更新ScoreText的內容
38      }
```

100 %

設定ScoreText的物件

把ScoreText拖曳到Background Inspector視窗裡面的Score Text



修改Invader.cs 偵測到碰撞後就加分

GameFunction.cs* | Invader.cs* | Explode.cs

Assembly-CSharp | Invader

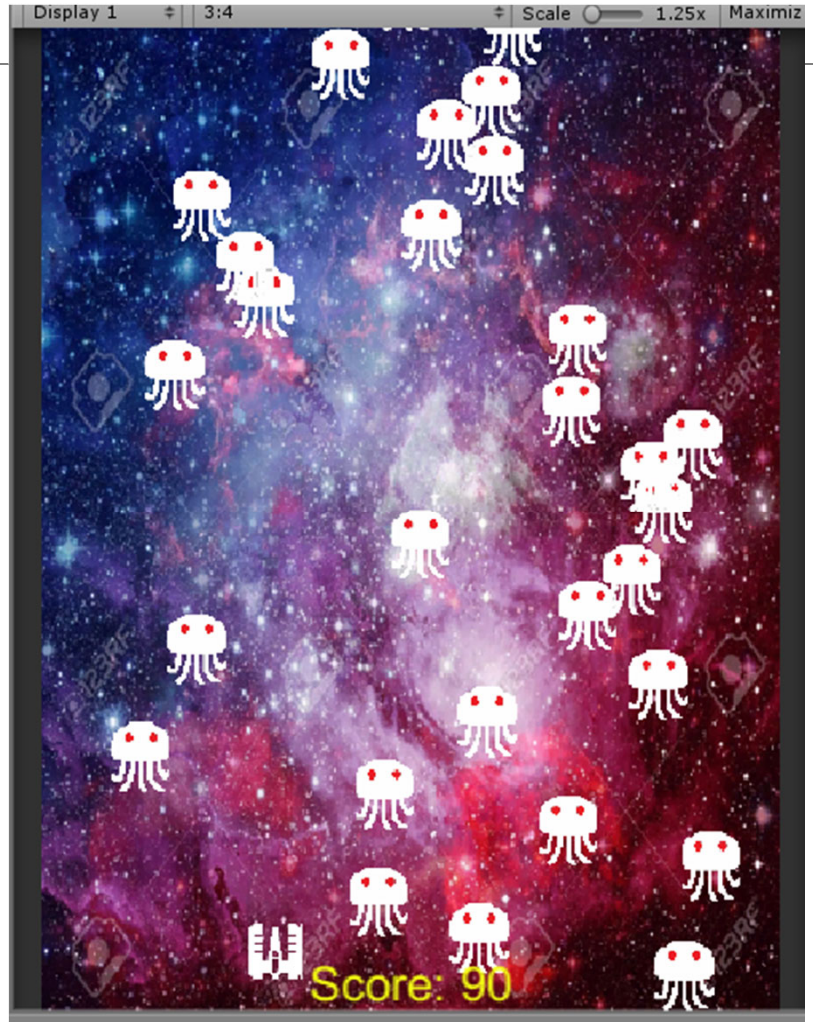
```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   public class Invader : MonoBehaviour
6   {
7       public GameObject explo; // 宣告一個名為explo的物件
8
9       // Start is called before the first frame update
10      void Start()
11      {
12
13      }
14
15      // Update is called once per frame
16      void Update()
17      {
18          gameObject.transform.position += new Vector3(0, -0.01f, 0);
19      }
20
21      void OnTriggerEnter2D(Collider2D col) //名為col的觸發事件
22      {
23          if (col.tag == "Ship" || col.tag == "Bullet") //如果碰撞的標籤是Ship或Bullet
24          {
25              Destroy(col.gameObject); //消滅被碰撞的物件
26              Destroy(gameObject); //消滅物件本身
27
28              Instantiate(explo, transform.position, transform.rotation); //在外星人的位置產生爆炸
29              if (col.tag == "Ship")
30              {
31                  Instantiate(explo, col.gameObject.transform.position, col.gameObject.transform.rotation);
32                  //在碰撞物件的位置產生爆炸，也就是在太空船的位置產生爆炸
33              }
34              GameFunction.Instance.AddScore();   //呼叫GameFunction底下的AddScore() 當碰撞觸發就加分
35          }
36      }
37  }
```

現在已經可以有加分了

Try it:

可以設定給予不同分數

有加分與扣分的敵人

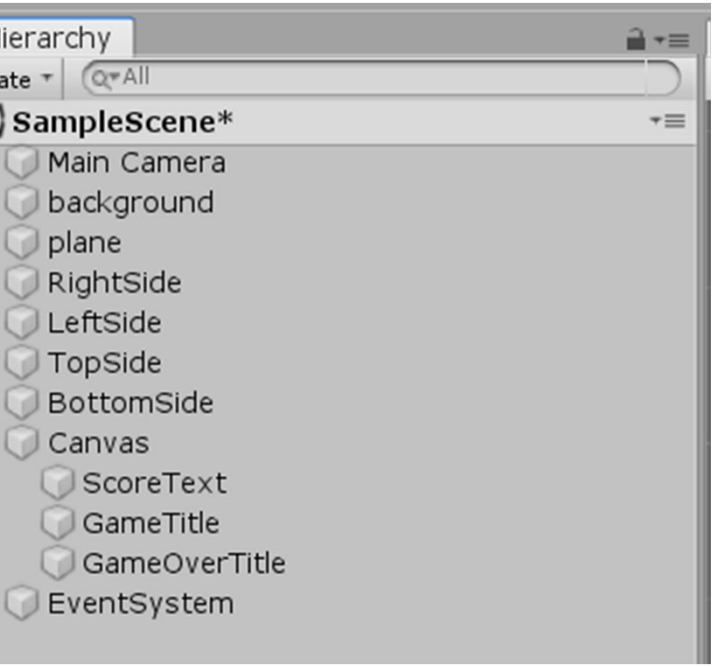


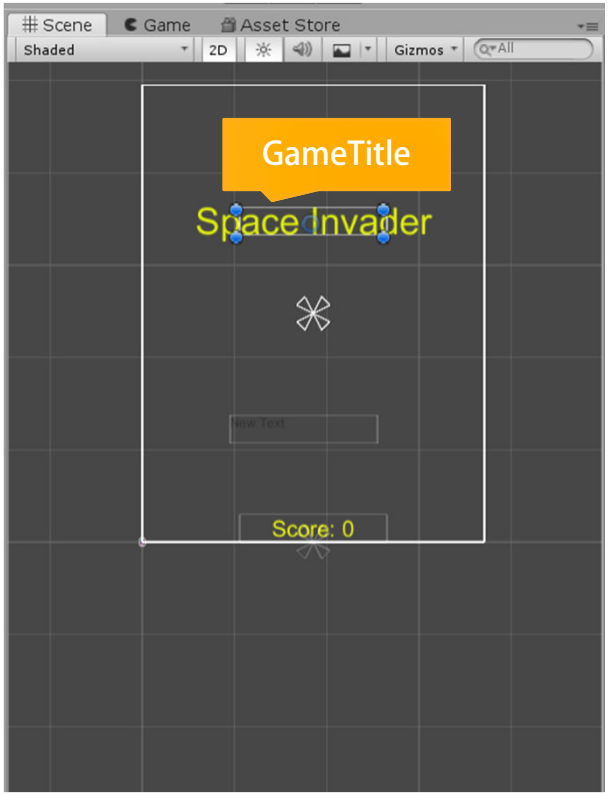
加入Game Start & End的畫面

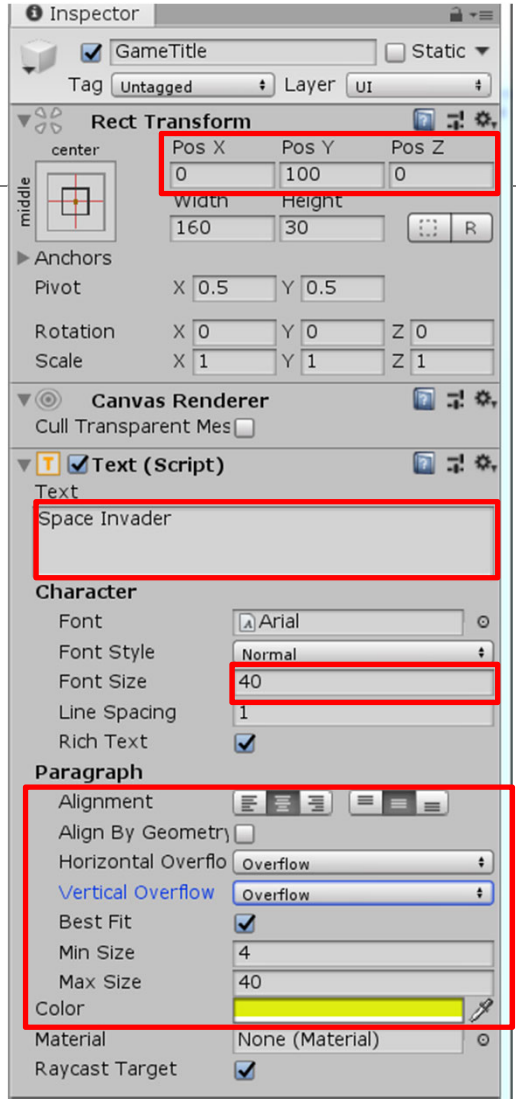
設定GameTitle

加入兩個文字

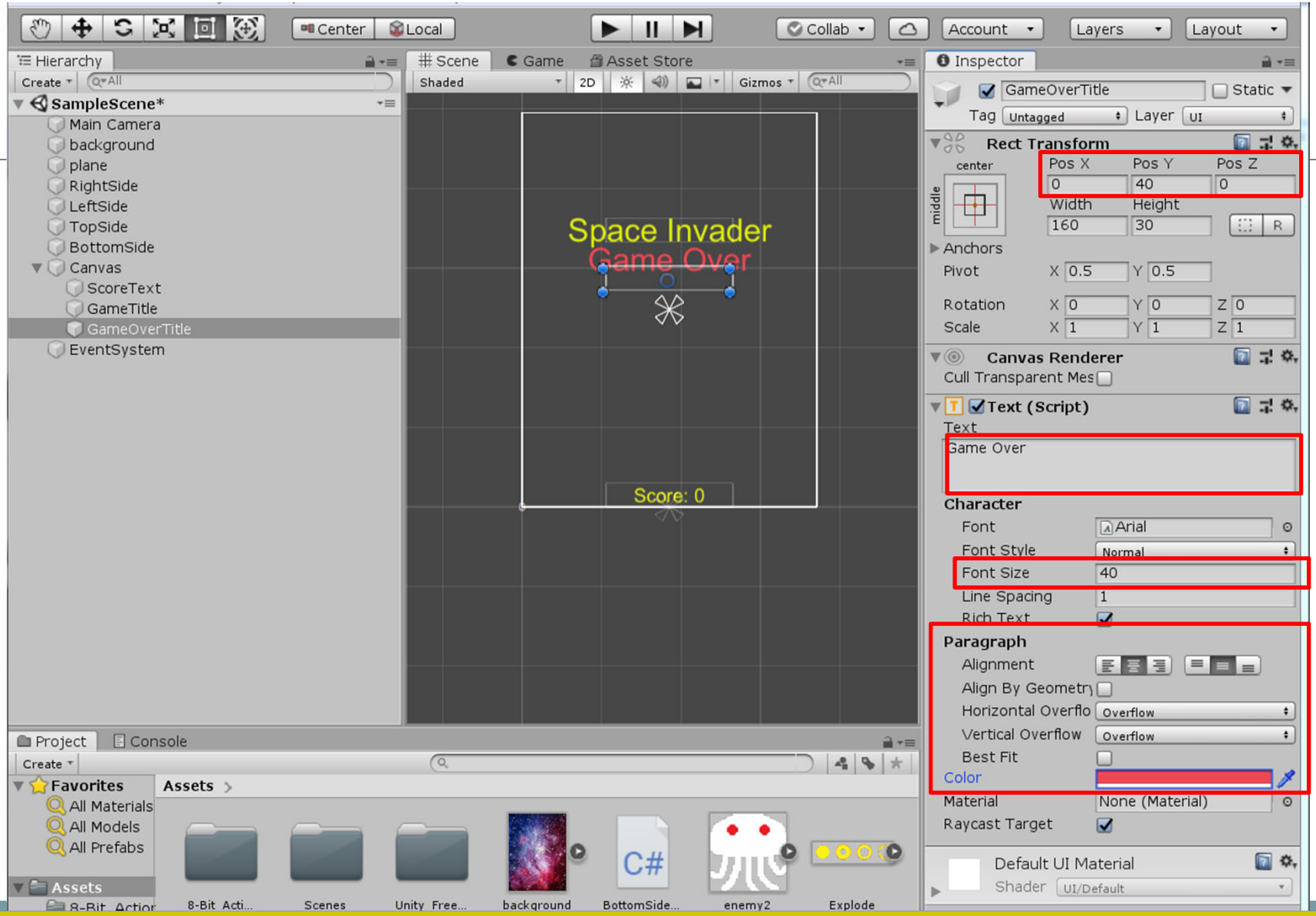
- GameTitle
- GameOverTitle



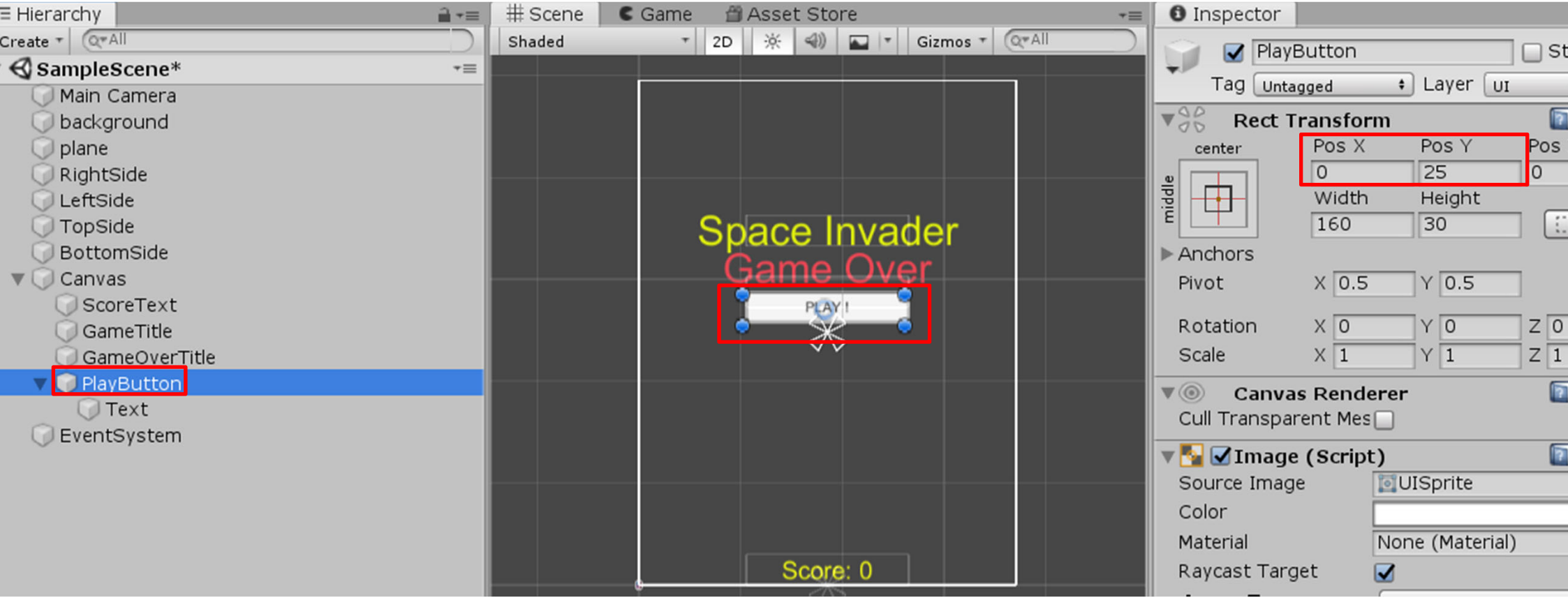




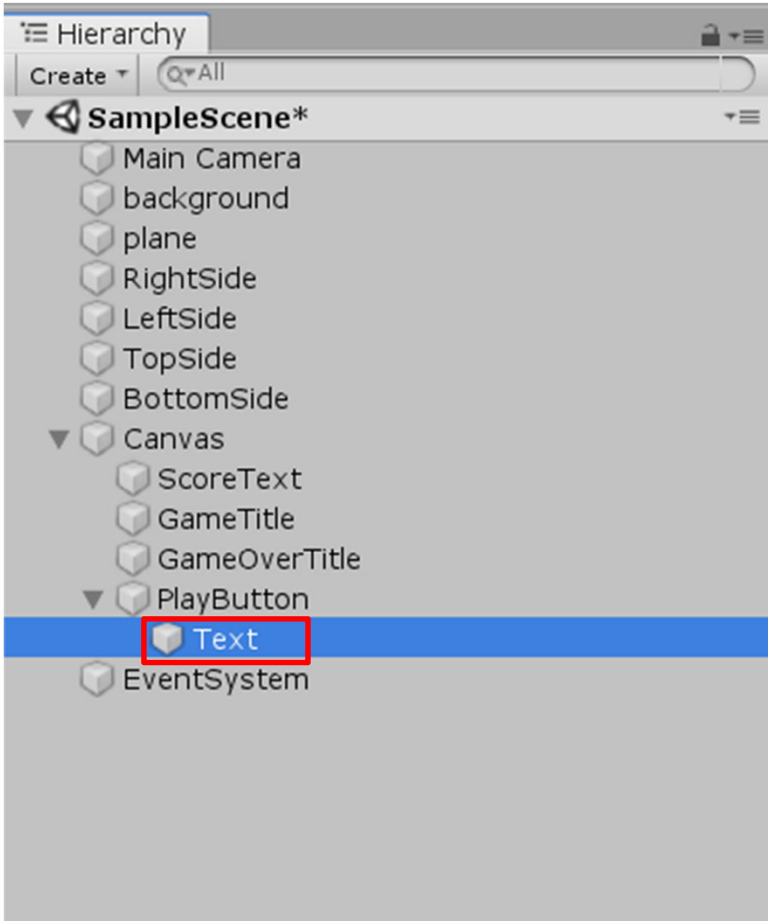
設定GameOverTitle

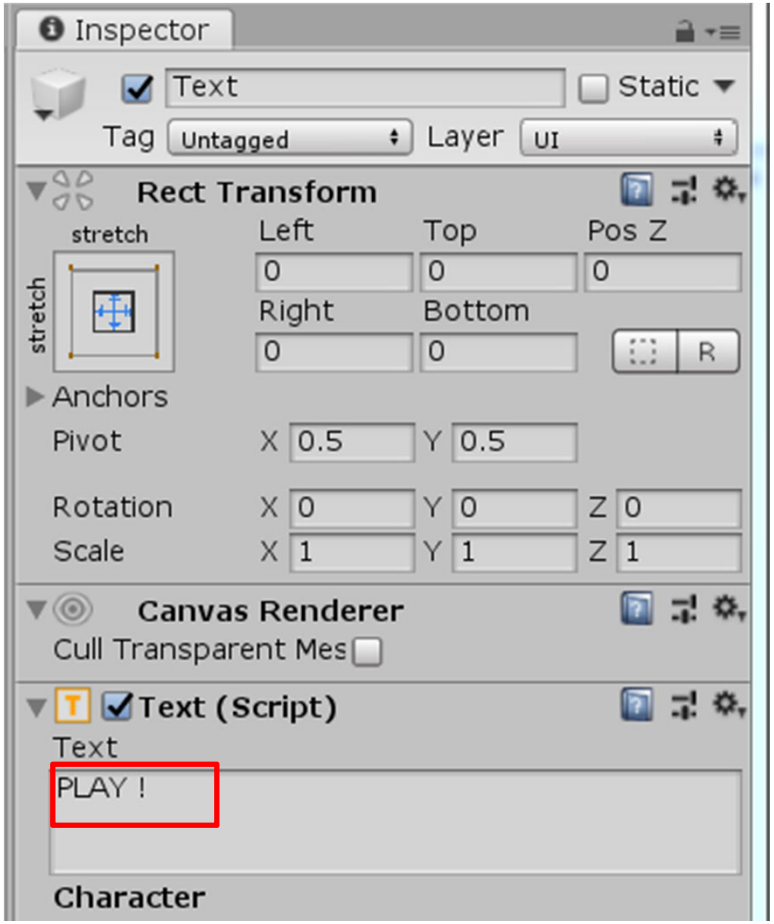


新增Play Button
Hierarchy(右鍵) UI->Button



更改PlayButton 按鈕文字



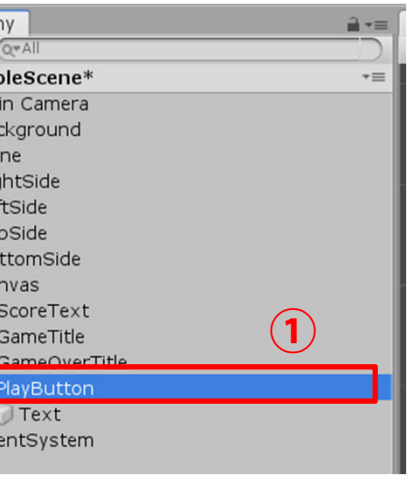


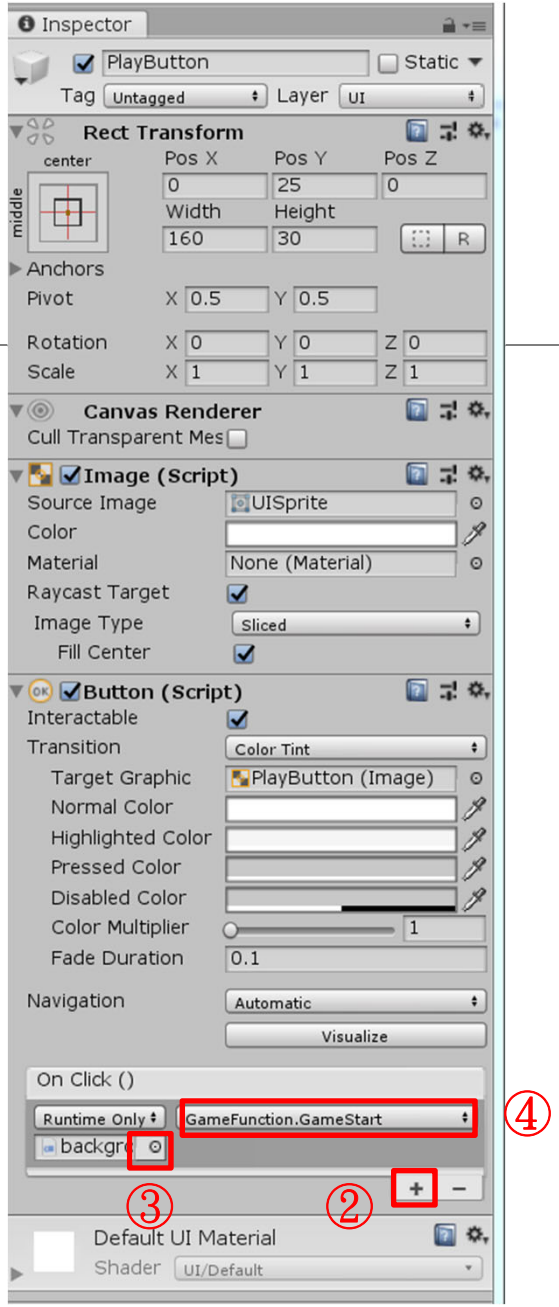
修改GameFunction.cs

控制GameTitle,GameOvertitle是否要顯示

控制PlayButton作用

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   using UnityEngine.UI;  //使用UI
6
7   public class GameFunction : MonoBehaviour
8   {
9       public GameObject Enemy; //宣告物件，名稱Enemy
10      public float time; //宣告浮點數，名稱time
11
12      public Text ScoreText;                  // 宣告一個文字型態的變數 ScoreText
13      public int Score = 0;                   // 宣告一個整數Score
14      public static GameFunction Instance;    // 設定Instance，讓其他程式能讀取GameFunction裡的東西
15
16      public GameObject GameTitle;            // 宣告 GameTitle物件
17      public GameObject GameOverTitle;        // 宣告 GameOverTitle物件
18      public GameObject PlayButton;           // 宣告 PlayButton物件
19      public bool isPlaying = false;          // 宣告 isPlaying的boolean變數，初始值false
20
21      // Start is called before the first frame update
22      void Start()
23      {
24          Instance = this;    // 指定Instance參考這個程式
25
26          GameOverTitle.SetActive(false);     // 設定GameOverTitle一開始不顯示
27      }
28
29      // Update is called once per frame
30      void Update()
31      {
32          time += Time.deltaTime; //時間增加
33          if (time > 0.5f && isPlaying == true) //如果時間大於0.5(秒) && isPlaying = True
34          {
35              Vector3 pos = new Vector3(Random.Range(-2.5f, 2.5f), 4.5f, 0); //宣告位置pos，Random.Range(-2.5f,2.5f)代表X是2.5到-2.5之間隨機
36              Instantiate(Enemy, pos, transform.rotation);//產生敵人
37              time = 0f; //時間歸零
38          }
39      }
40
41      public void AddScore()
42      {
43          Score += 10;        //分數+10分
44          ScoreText.text = "Score: " + Score;     //更新ScoreText的內容
45      }
46
47      public void GameStart()
48      {
49          isPlaying = true;                   // 設定isPlaying = true,代表遊戲進行中
50          GameTitle.SetActive(false);         // 不顯示GameTitle
51          PlayButton.SetActive(false);        // 不顯示PlayButton
52
53      }
54  }
55
```





設定PlayButton屬性

Step #1. 選PlayButton

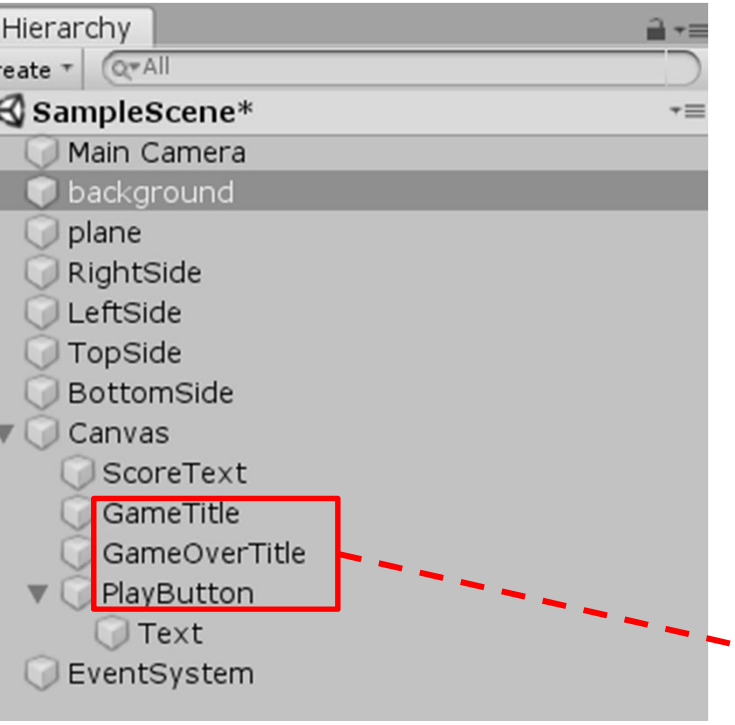
Step #2. 按 +

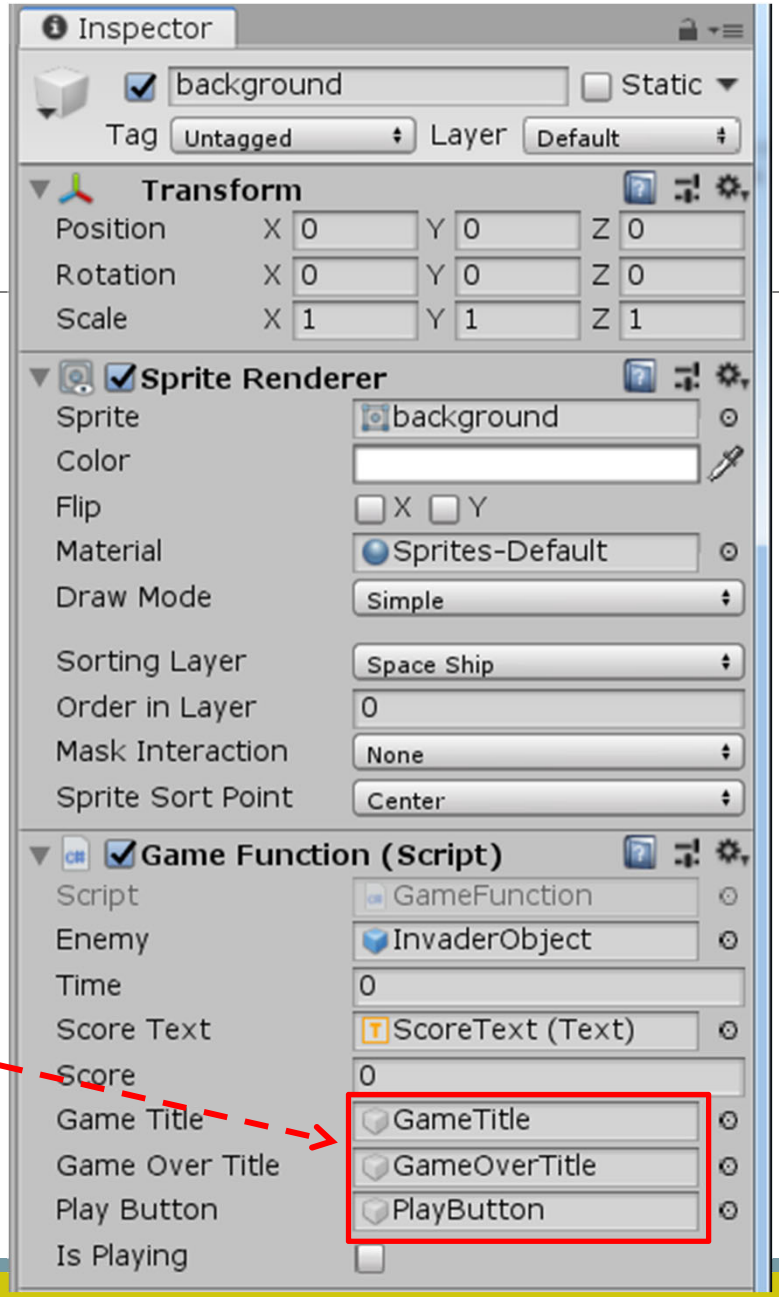
Step #3. 按小圓圈

Step #4. 選Scene分頁，選background

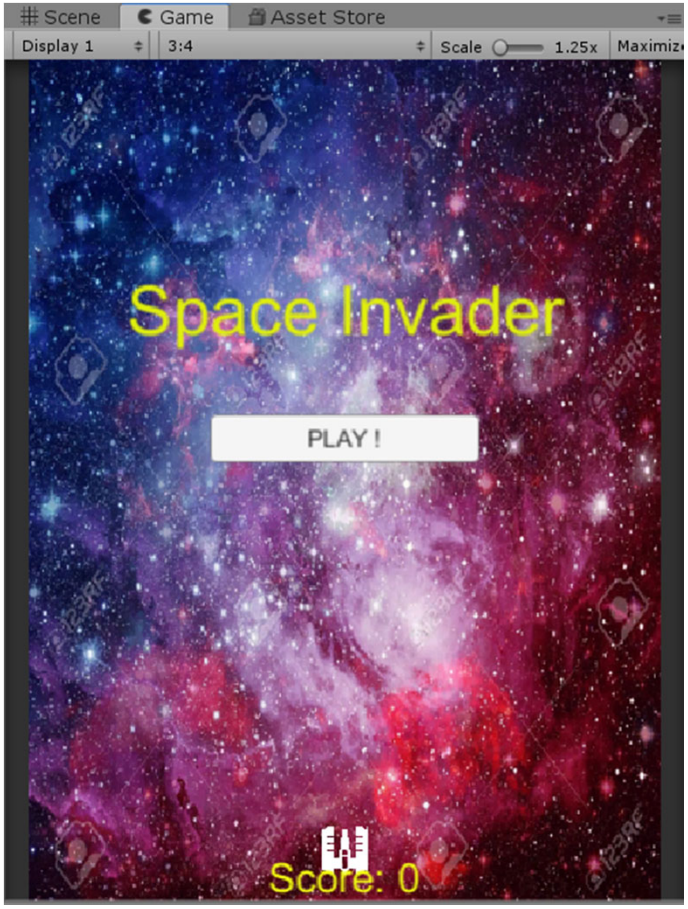
Step #5. 選GameFunction.GameStart()

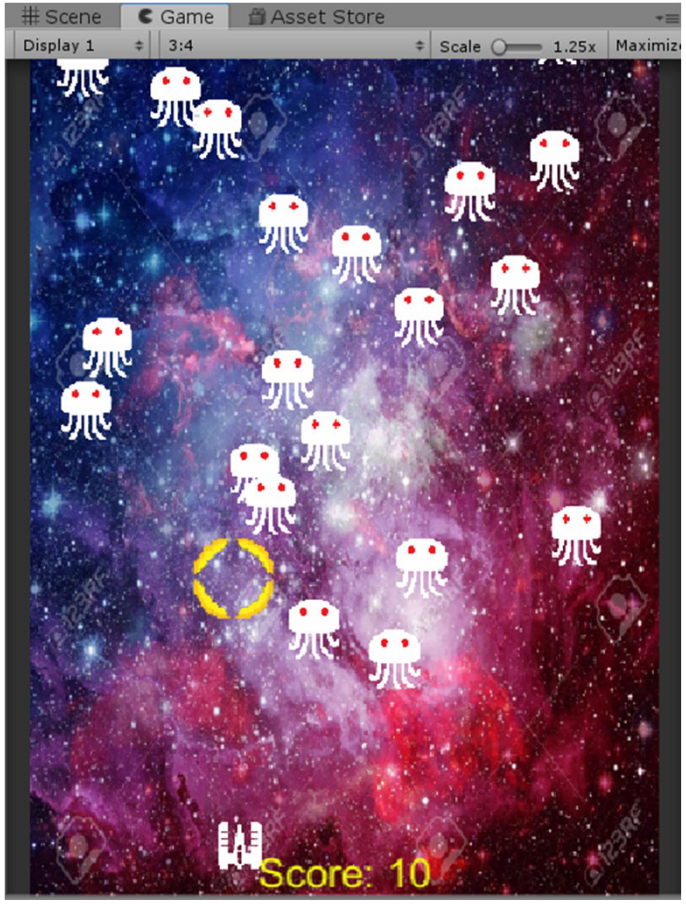
指定GameFunction的物件





目前完成畫面：
有Play按鈕可以開始玩





修改GameFunction
遊戲GameOverTitle

Assembly-CSharp | GameFunction | Start()

```
7   public class GameFunction : MonoBehaviour
8   {
9       public GameObject Enemy; //宣告物件，名稱Enemy
10      public float time; //宣告浮點數，名稱time
11
12      public Text ScoreText;                  // 宣告一個文字型態的變數 ScoreText
13      public int Score = 0;                   // 宣告一個整數Score
14      public static GameFunction Instance;    // 設定Instance，讓其他程式能讀取GameFunction裡的東西
15
16      public GameObject GameTitle;            // 宣告 GameTitle物件
17      public GameObject GameOverTitle;        // 宣告 GameOverTitle物件
18      public GameObject PlayButton;           // 宣告 PlayButton物件
19      public bool isPlaying = false;          // 宣告 isPlaying的boolean變數，初始值false
20
21      // Start is called before the first frame update
22      void Start()
23      {
24          Instance = this;    // 指定Instance參考這個程式
25
26          GameOverTitle.SetActive(false);     // 設定GameOverTitle一開始不顯示
27      }
28
29      // Update is called once per frame
30      void Update()
31      {
32          time += Time.deltaTime; //時間增加
33          if (time > 0.5f && isPlaying == true) //如果時間大於0.5(秒) && isPlaying = True
34          {
35              Vector3 pos = new Vector3(Random.Range(-2.5f, 2.5f), 4.5f, 0); //宣告位置pos，Random.Range(-2.5f,2.5f)代表X是2.5到-2.5之間隨機
36              Instantiate(Enemy, pos, transform.rotation);//產生敵人
37              time = 0f; //時間歸零
38          }
39      }
40
41      public void AddScore()
42      {
43          Score += 10;        //分數+10分
44          ScoreText.text = "Score: " + Score;     //更新ScoreText的內容
45      }
46
47      public void GameStart()
48      {
49          isPlaying = true;                   // 設定isPlaying = true,代表遊戲進行中
50          GameTitle.SetActive(false);         // 不顯示GameTitle
51          PlayButton.SetActive(false);        // 不顯示PlayButton
52
53      }
54
55      public void GameOver()
56      {
57          isPlaying = false;                  // 設定isPlaying = false, 代表遊戲結束
58          GameOverTitle.SetActive(true);      // 設定顯示GameOverTitle
59      }
60  }
61
```

修改GameFunction
遊戲GameOverTitle

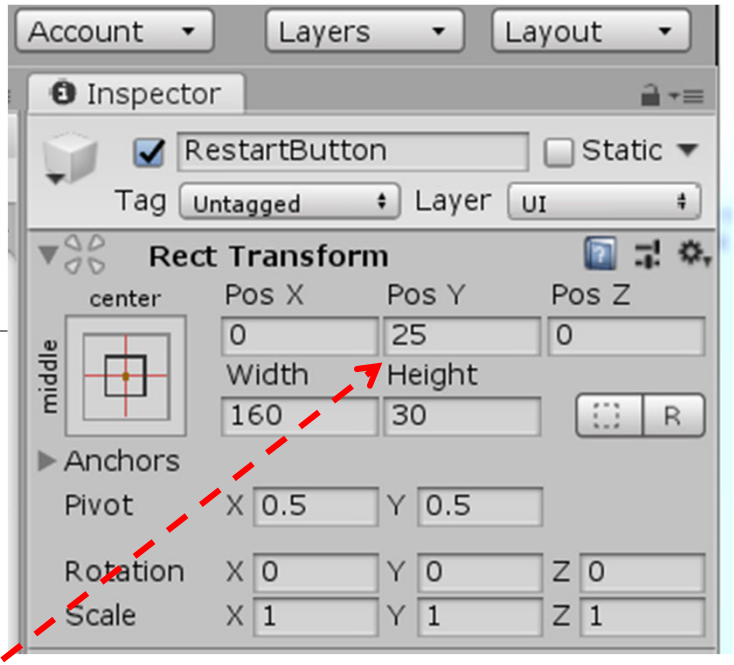
Assembly-CSharp | Invader | OnTriggerEnter2D(Collider2D col)

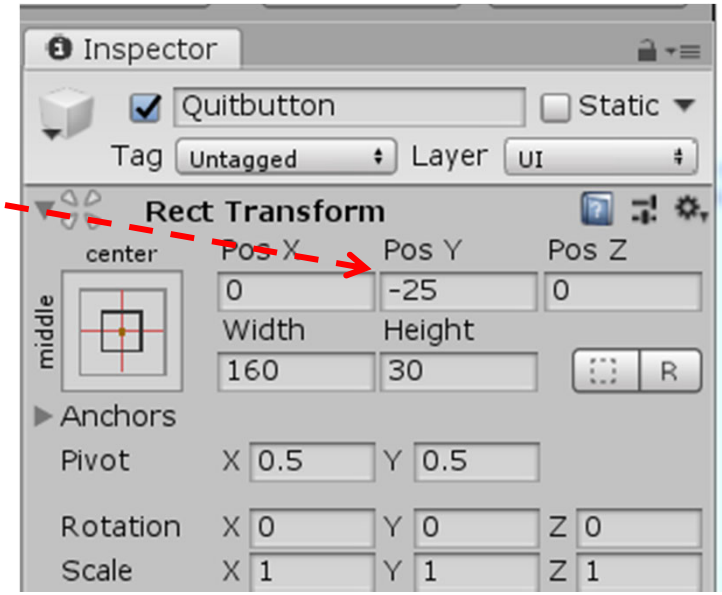
```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class Invader : MonoBehaviour
6  {
7      public GameObject explo; // 宣告一個名為explo的物件
8
9      // Start is called before the first frame update
10     void Start()
11     {
12
13     }
14
15     // Update is called once per frame
16     void Update()
17     {
18         gameObject.transform.position += new Vector3(0, -0.01f, 0);
19     }
20
21     void OnTriggerEnter2D(Collider2D col) //名為col的觸發事件
22     {
23         if (col.tag == "Ship" || col.tag == "Bullet") //如果碰撞的標籤是Ship或Bullet
24         {
25             Destroy(col.gameObject); //消滅被碰撞的物件
26             Destroy(gameObject); //消滅物件本身
27
28             Instantiate(explo, transform.position, transform.rotation); //在外星人的位置產生爆炸
29             if (col.tag == "Ship")
30             {
31                 Instantiate(explo, col.gameObject.transform.position, col.gameObject.transform.rotation);
32                 //在碰撞物件的位置產生爆炸，也就是在太空船的位置產生爆炸
33
34                 GameFunction.Instance.GameOver();
35             }
36             GameFunction.Instance.AddScore();    //呼叫GameFunction底下的AddScore() 當碰撞觸發就加分
37         }
38     }
39 }
40
```

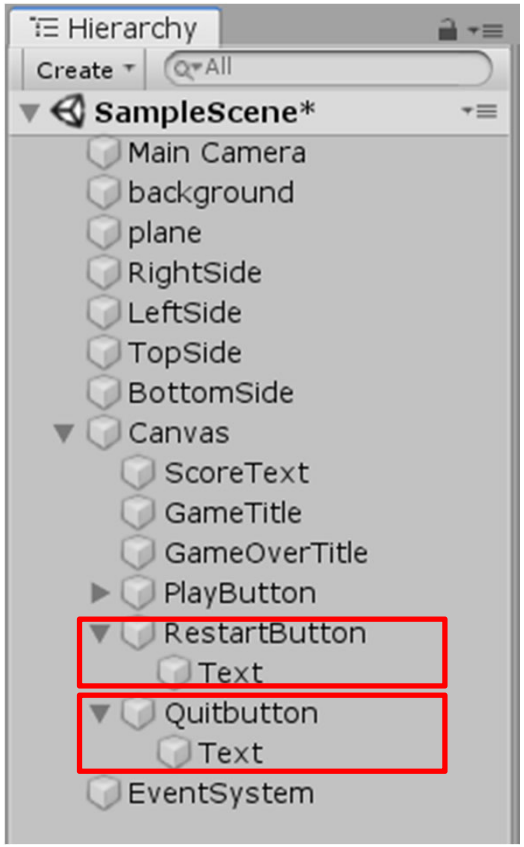
已經可以顯示GameOver
遊戲結束

加入RESTART & QUIT按鈕

加入
Restart & Quit按鈕







GameFunction

加入對RestartButton按鈕的控制

GameFunction.cs

Assembly-CSharp

GameFunction

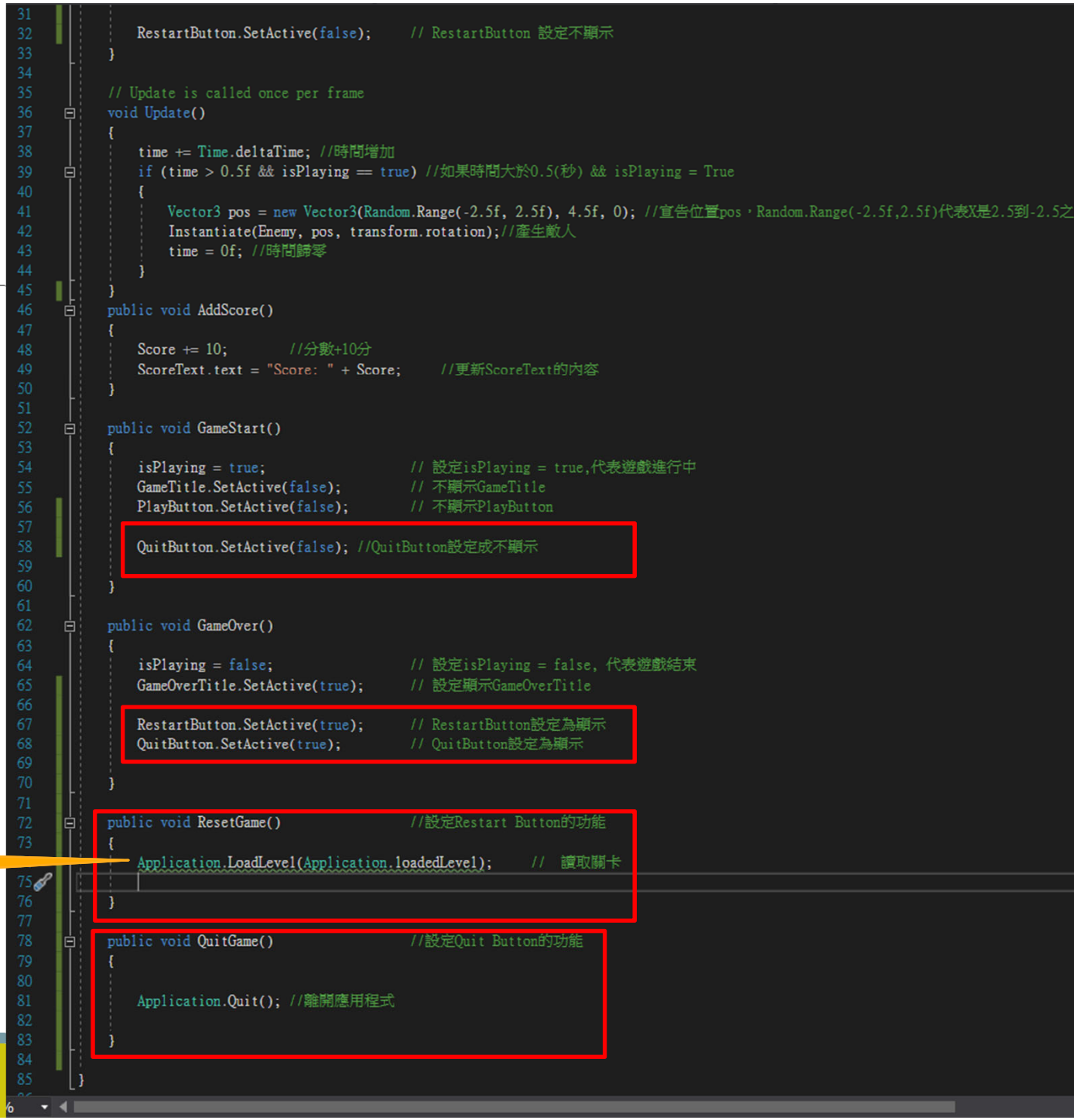
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

using UnityEngine.UI;  //使用UI

public class GameFunction : MonoBehaviour
{
    public GameObject Enemy; //宣告物件，名稱Enemy
    public float time; //宣告浮點數，名稱time

    public Text ScoreText;                  // 宣告一個文字型態的變數 ScoreText
    public int Score = 0;                   // 宣告一個整數Score
    public static GameFunction Instance;    // 設定Instance，讓其他程式能讀取GameFunction裡的東西

    public GameObject GameTitle;            // 宣告 GameTitle物件
    public GameObject GameOverTitle;        // 宣告 GameOverTitle物件
    public GameObject PlayButton;           // 宣告 PlayButton物件
    public bool isPlaying = false;          // 宣告 isPlaying的boolean變數，初始值false


    public GameObject RestartButton;        //宣告RestartButto的物件
    public GameObject QuitButton;           //宣告QuitButton的物件

    // Start is called before the first frame update
    void Start()
    {
        Instance = this;    // 指定Instance參考這個程式

        GameOverTitle.SetActive(false);     // 設定GameOverTitle一開始不顯示

        RestartButton.SetActive(false);     // RestartButton 設定不顯示
    }

    // Update is called once per frame
    void Update()
    {
        time += Time.deltaTime; //時間增加
        if (time > 0.5f && isPlaying == true) //如果時間大於0.5(秒) && isPlaying = True
        {
            Vector3 pos = new Vector3(Random.Range(-2.5f, 2.5f), 4.5f, 0); //宣告位置pos，Random.Range(-2.5f,2.5f)代表X是2.5到-2
            Instantiate(Enemy, pos, transform.rotation);//產生敵人
            time = 0f; //時間歸零
        }
    }

    public void AddScore()
    {
```

GameFunction

入對RestartButton按鈕的控制

入對QuitButton按鈕的控制

程式碼過期
需要修正



```
31
32          RestartButton.SetActive(false);     // RestartButton 設定不顯示
33      }
34
35      // Update is called once per frame
36      void Update()
37      {
38          time += Time.deltaTime; //時間增加
39          if (time > 0.5f && isPlaying == true) //如果時間大於0.5(秒) && isPlaying = True
40          {
41              Vector3 pos = new Vector3(Random.Range(-2.5f, 2.5f), 4.5f, 0); //宣告位置pos，Random.Range(-2.5f,2.5f)代表X是2.5到-2.5之
42              Instantiate(Enemy, pos, transform.rotation);//產生敵人
43              time = 0f; //時間歸零
44          }
45      }
46      public void AddScore()
47      {
48          Score += 10;        //分數+10分
49          ScoreText.text = "Score: " + Score;     //更新ScoreText的內容
50      }
51
52      public void GameStart()
53      {
54          isPlaying = true;                   // 設定isPlaying = true,代表遊戲進行中
55          GameTitle.SetActive(false);         // 不顯示GameTitle
56          PlayButton.SetActive(false);        // 不顯示PlayButton
57
58          QuitButton.SetActive(false); //QuitButton設定成不顯示
59
60      }
61
62      public void GameOver()
63      {
64          isPlaying = false;                  // 設定isPlaying = false, 代表遊戲結束
65          GameOverTitle.SetActive(true);      // 設定顯示GameOverTitle
66
67          RestartButton.SetActive(true);      // RestartButton設定為顯示
68          QuitButton.SetActive(true);         // QuitButton設定為顯示
69
70      }
71
72      public void ResetGame()                 //設定Restart Button的功能
73      {
74          Application.LoadLevel(Application.loadedLevel);     //  讀取關卡
75
76      }
77
78      public void QuitGame()                  //設定Quit Button的功能
79      {
80
81          Application.Quit(); //離開應用程式
82
83      }
84
85  }
```

GameFunction.cs修正

先ResetGame()
取關卡的程式碼在新版已經修改。

plication.LoadLevel(Application.loadLevel)
經不能用

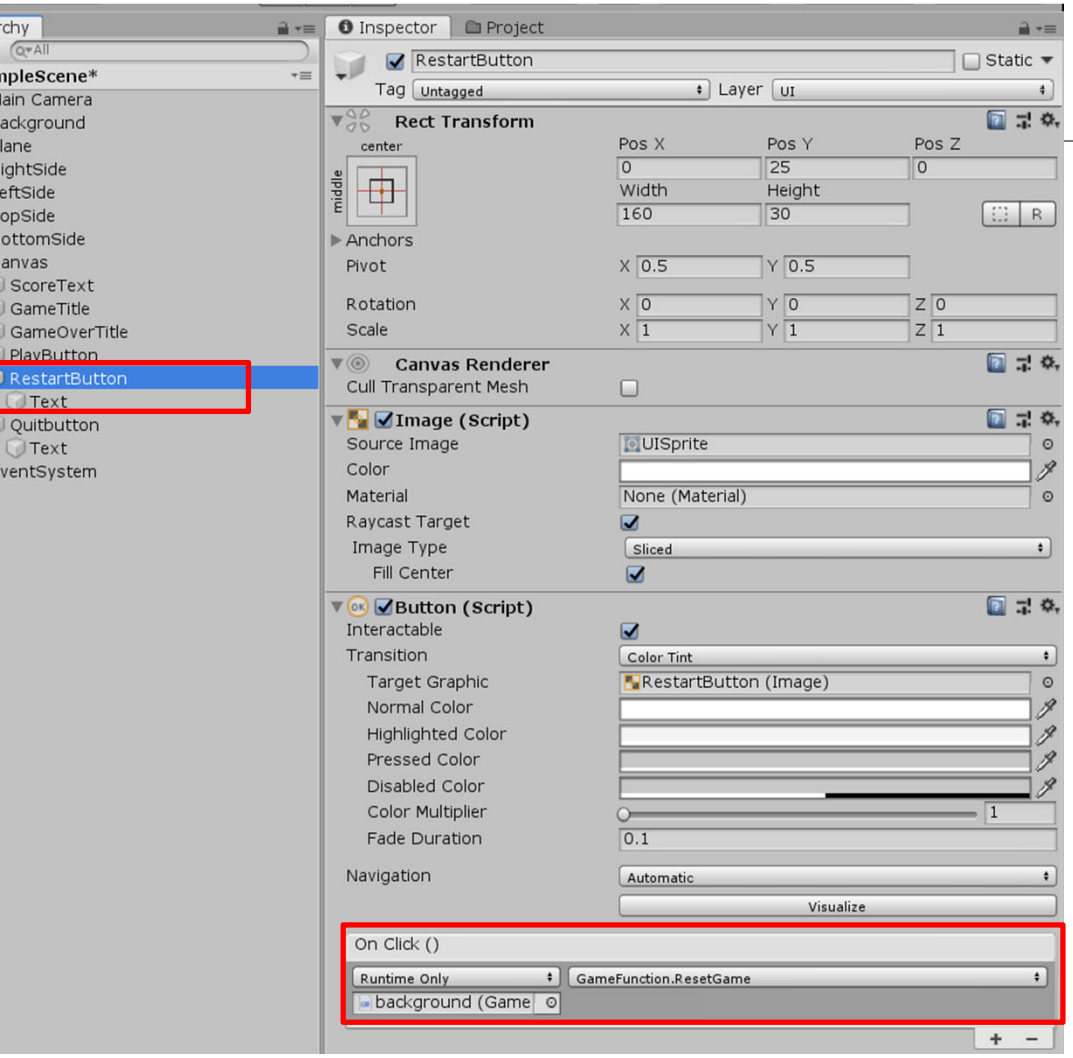
改為 SceneManagement.LoadScene()

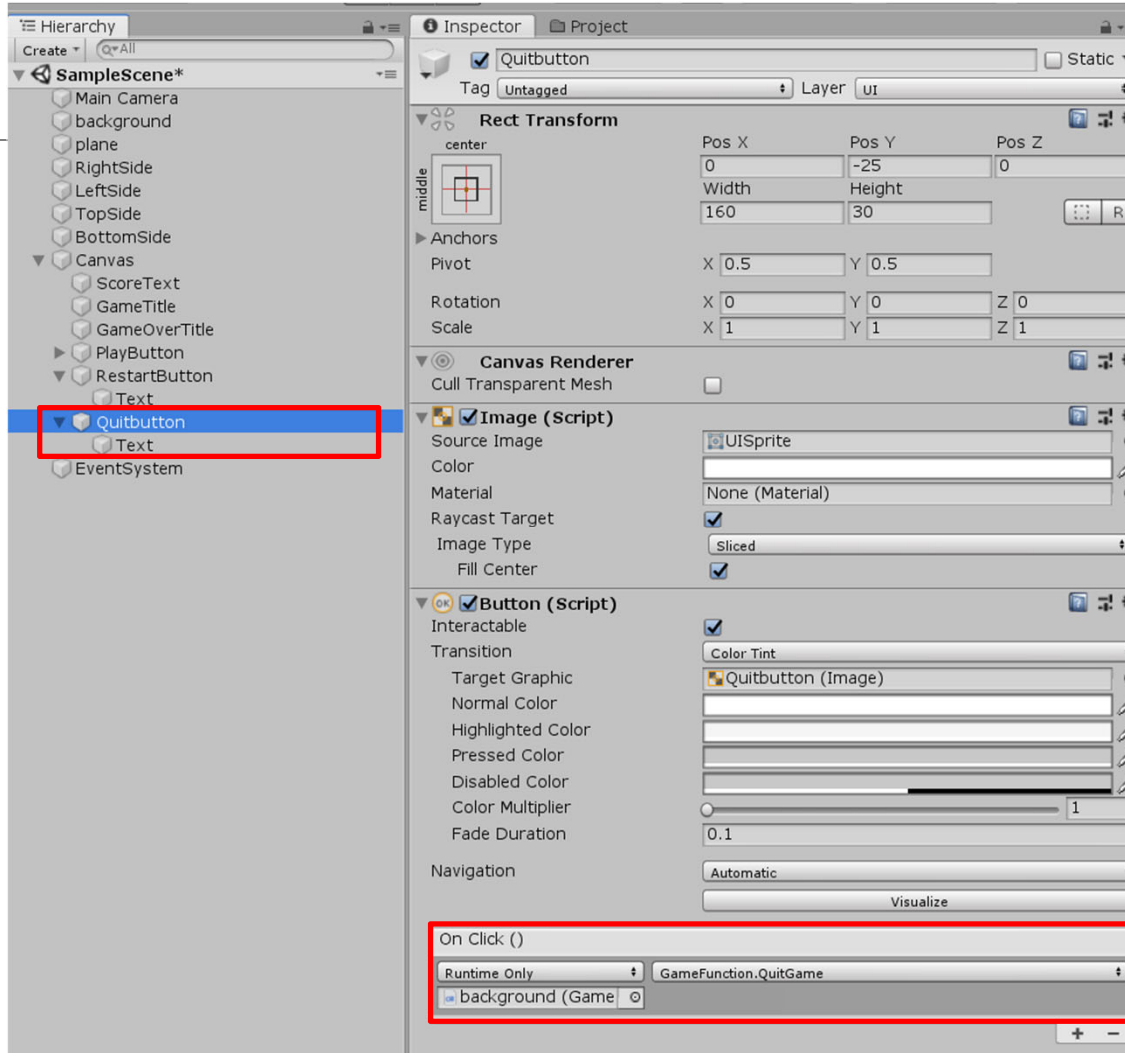
Assembly-CSharp — GameFunction

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   using UnityEngine.UI;  //使用UI
6   using UnityEngine.SceneManagement;
7
```

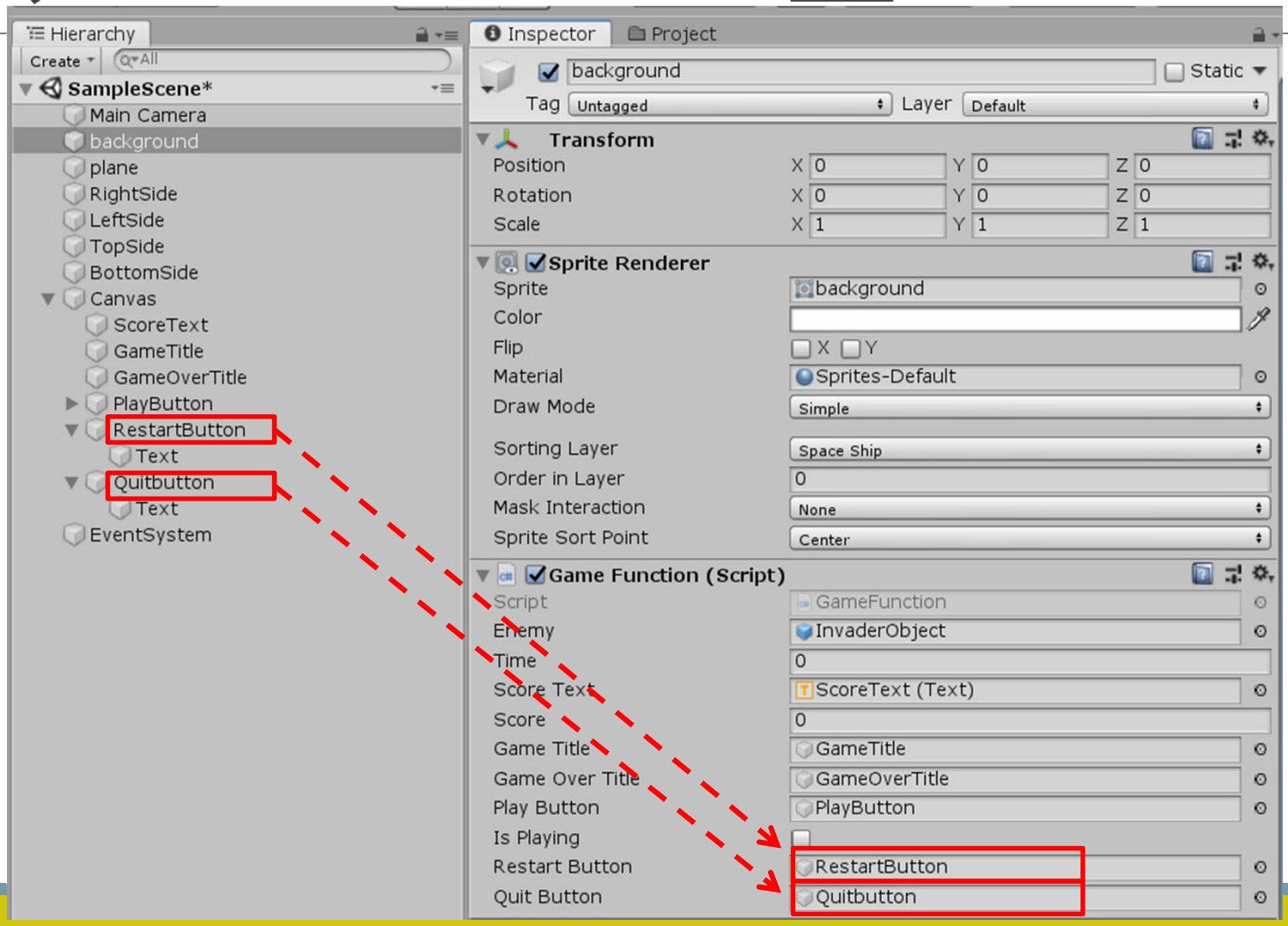
```
74      public void ResetGame()                 //設定Restart Button的功能
75      {
76          //Application.LoadLevel(Application.loadedLevel);     //  讀取關卡
77
78          // get the current scene name
79          string sceneName = SceneManager.GetActiveScene().name;
80
81          // load the same scene
82          SceneManager.LoadScene(sceneName, LoadSceneMode.Single); // 讀取現在的場景
83
84      }
85
86      public void QuitGame()                  //設定Quit Button的功能
87      {
88
89          Application.Quit(); //離開應用程式
90
91      }
92
93  }
```

設定RestartButton & Quit Button按下去啟動的副程式



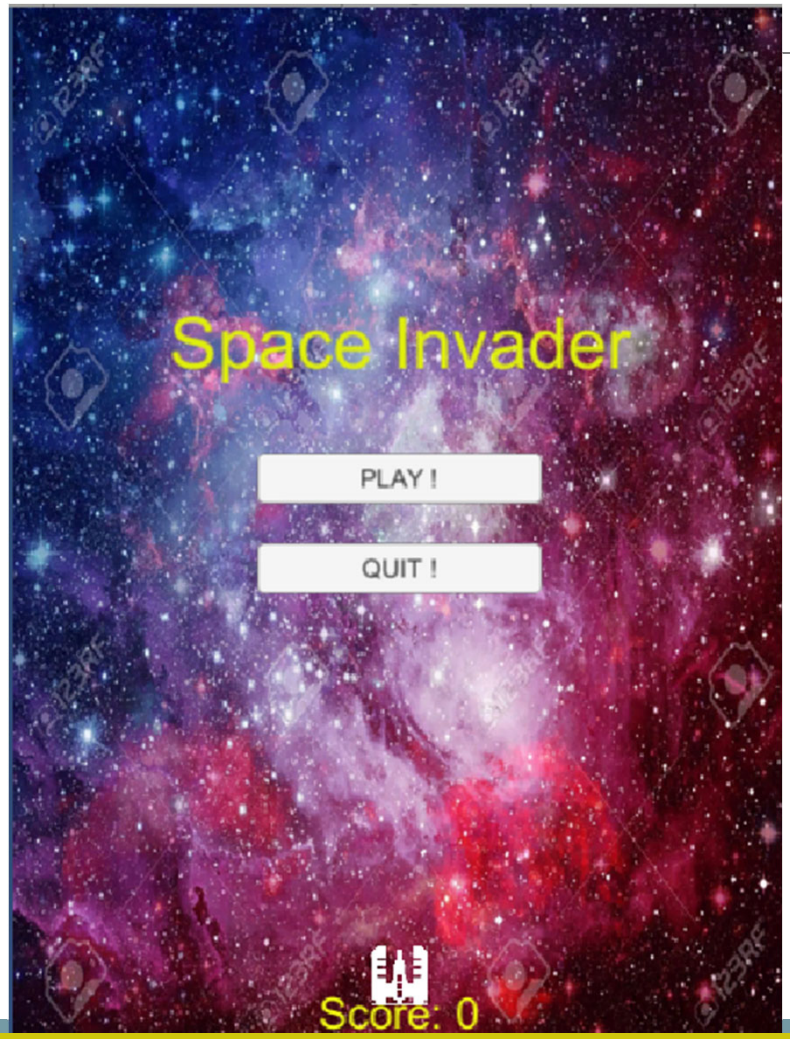


把RestartButton & QuitButton拖曳加入GameFunction上



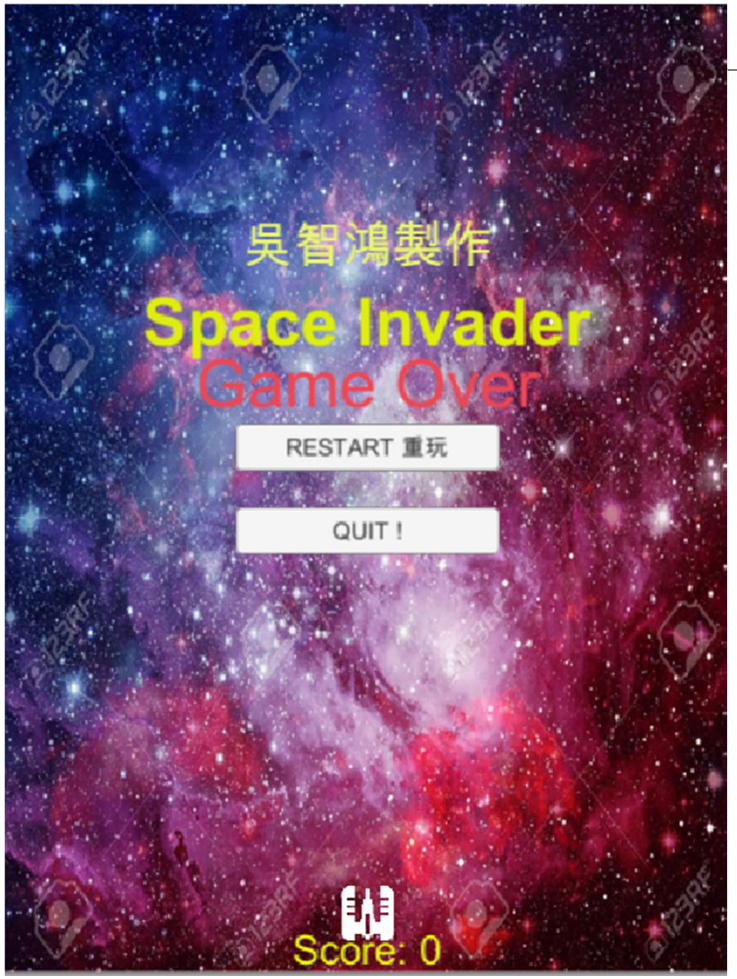
完成樣子

開始畫面



結束畫面 (可以重玩&離開)

加入作者資訊



GameFunction.cs

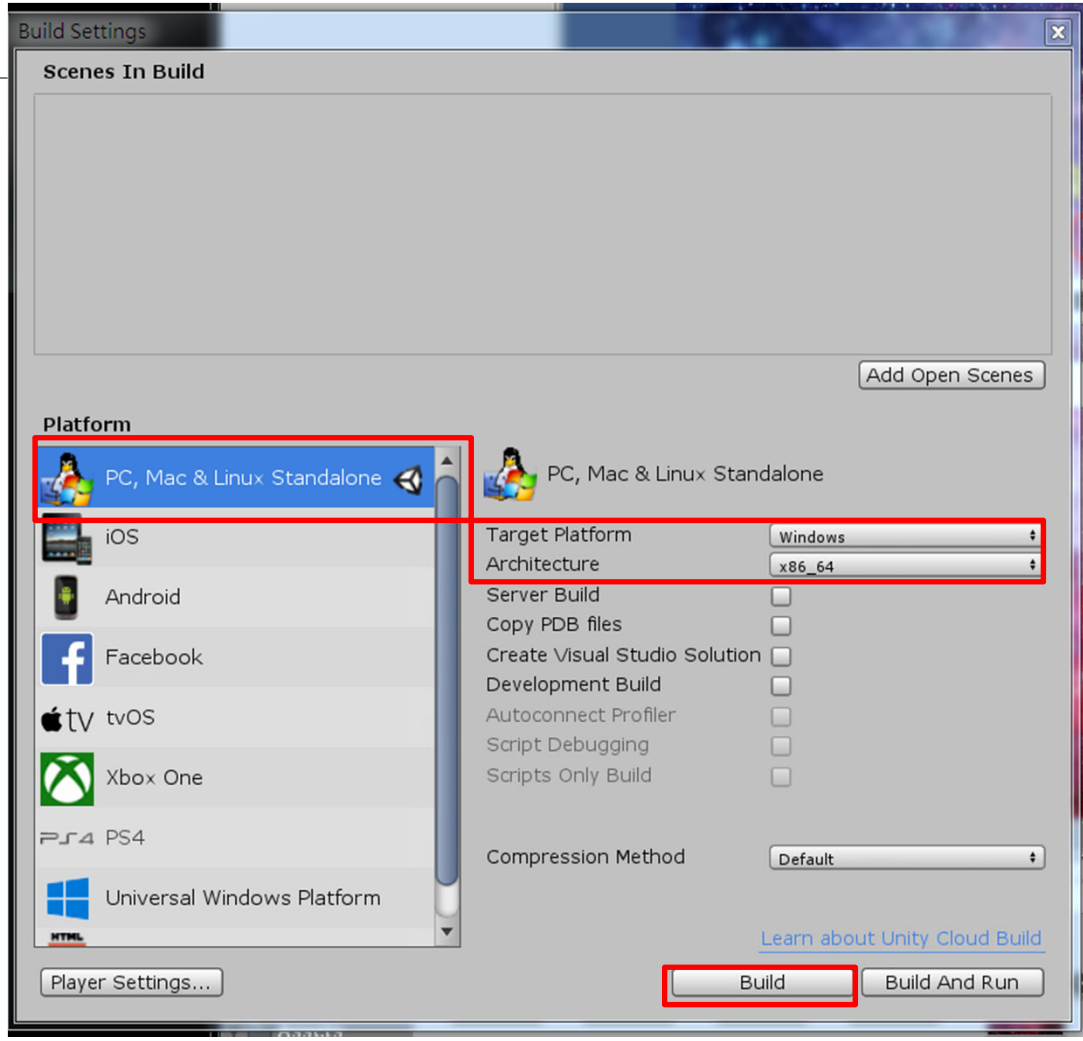
Assembly-CSharp

GameFunction

```
19    public GameObject GameOverTitle;        // 宣告 GameOverTitle物件
20    public GameObject PlayButton;           // 宣告 PlayButton物件
21    public bool isPlaying = false;          // 宣告 isPlaying的boolean變數，初始值false
22
23
24    public GameObject RestartButton;        //宣告RestartButto的物件
25    public GameObject QuitButton;           //宣告QuitButton的物件
26
27    public GameObject AuthorTitle;          // 宣告 GameTitle物件
28
29    // Start is called before the first frame update
30    void Start()
31    {
32        Instance = this;    // 指定Instance參考這個程式
33
34        GameOverTitle.SetActive(false);     // 設定GameOverTitle一開始不顯示
35
36        RestartButton.SetActive(false);     // RestartButton 設定不顯示
37    }
38
39    // Update is called once per frame
40    void Update()
41    {
42        time += Time.deltaTime; //時間增加
43        if (time > 0.5f && isPlaying == true) //如果時間大於0.5(秒) && isPlaying = True
44        {
45            Vector3 pos = new Vector3(Random.Range(-2.5f, 2.5f), 4.5f, 0); //宣告位置pos，Random.Range(-2.5f,2.5f)代表X是2.5到
46            Instantiate(Enemy, pos, transform.rotation);//產生敵人
47            time = 0f; //時間歸零
48        }
49    }
50    public void AddScore()
51    {
52        Score += 10;        //分數+10分
53        ScoreText.text = "Score: " + Score;     //更新ScoreText的內容
54    }
55
56    public void GameStart()
57    {
58        isPlaying = true;                   // 設定isPlaying = true,代表遊戲進行中
59        GameTitle.SetActive(false);         // 不顯示GameTitle
60        PlayButton.SetActive(false);        // 不顯示PlayButton
61        AuthorTitle.SetActive(false);
62
63        QuitButton.SetActive(false);        //QuitButton設定成不顯示
64
65    }
66
```

File->Build Settings

發布成Windows可以執行的exe檔



參考來源:

http://readandplay.pixnet.net/blog/post/197395500?fbclid=IwAR3QT0g810RicDzPKPr014abmgKxJYloGbGoYjOXFb7-rDy54pPQzHXQVWo