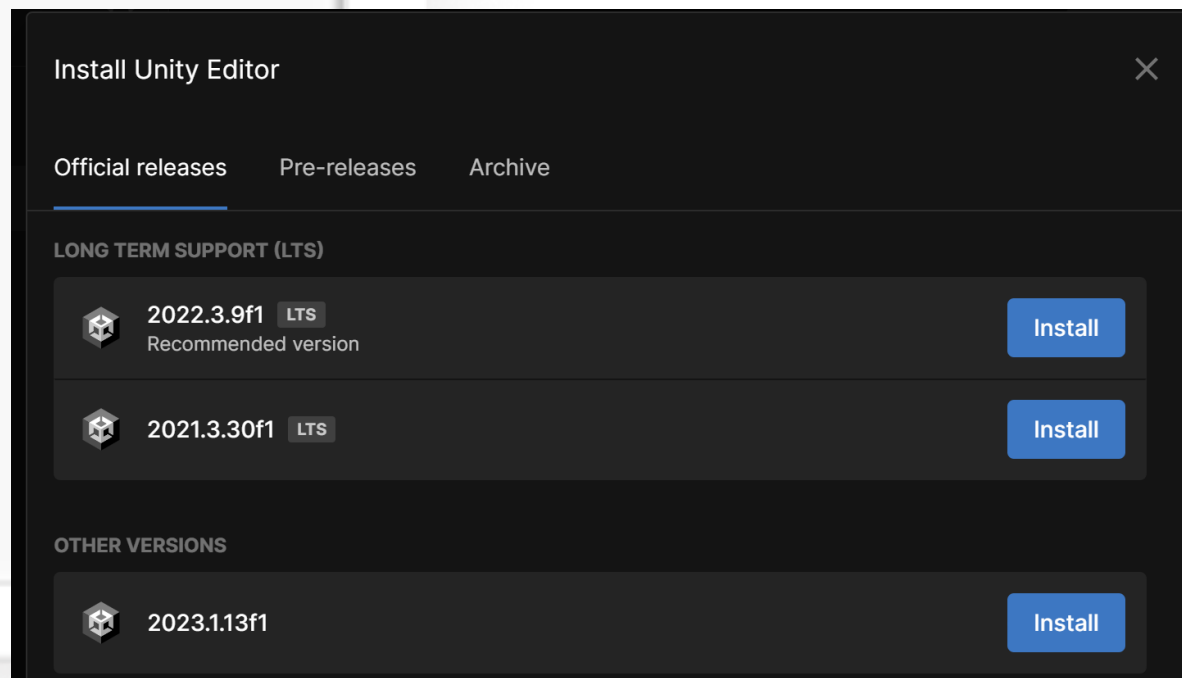


安裝 Unity Hub unity 2022.3.X (LTS)

<https://unity3d.com/get-unity/download>

Install Unity Hub

申請帳號 > 填問卷 > 完成!



Unity 基本介紹

INTERACTIVE
MEDIA



Outline

- What is Unity 3D?
- Unity interface
- Unity中重要的名詞介紹
- Unity遊戲設計架構
- Reference

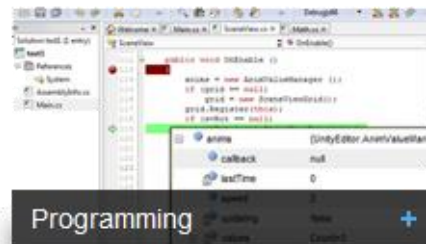
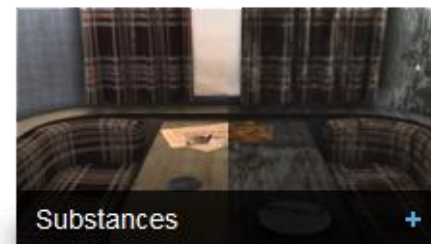
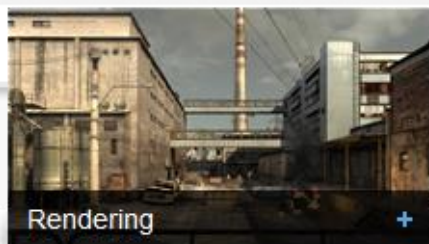


What's Unity 3D?

iNTERACTIVE
MEDiA

What's Unity 3D

 Powerful functions



INTERACTIVE
MEDIA

Game



console

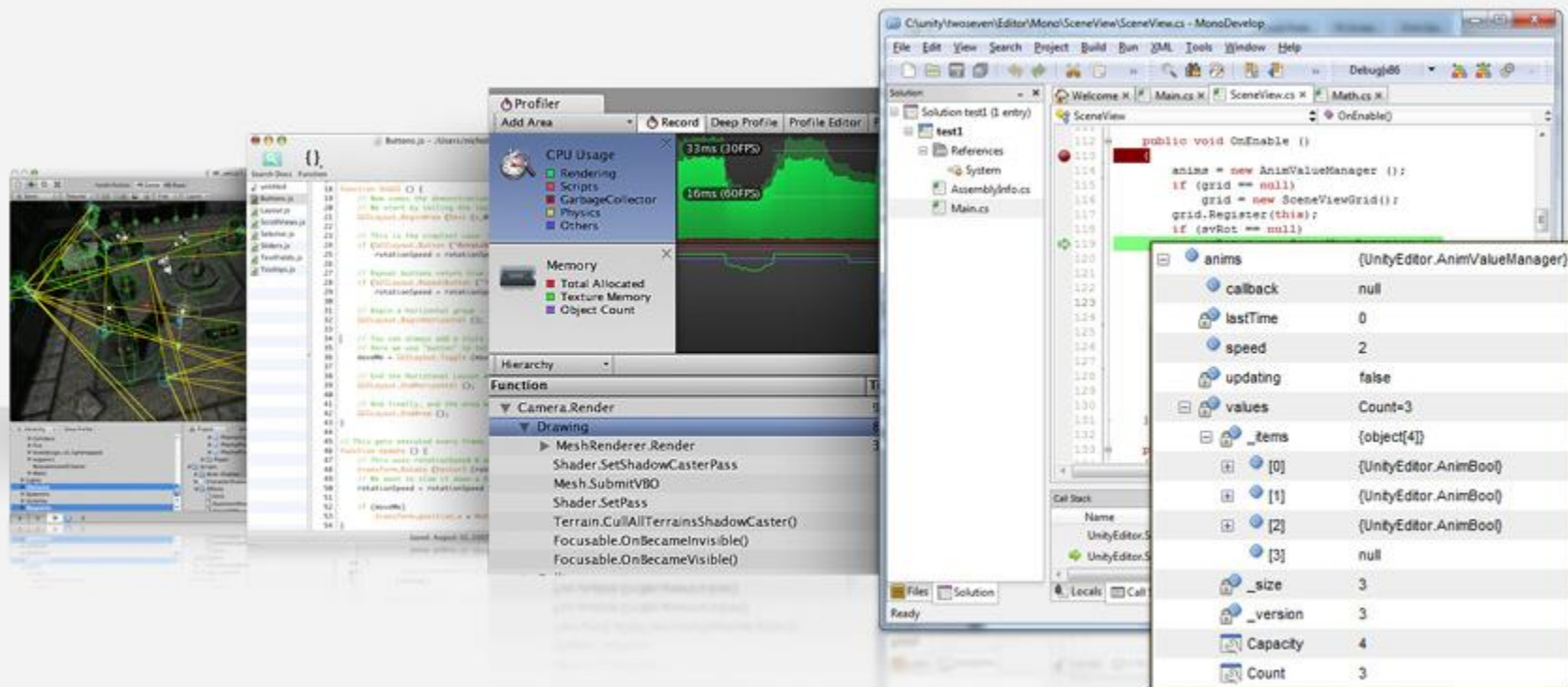


mobile




VR AR






Programming C#




INTERACTIVE
MEDIA


Asset Store


 **Unity Asset Store**




3D2DAdd-OnsAudioAIDecentralization**Essentials**TemplatesToolsVFXSaleSell Assets

 Over **11,000 five-star** assets

 Rated by **85,000+** customers

 Supported by **100,000+** forum members

 Every asset **moderated** by Unity

Home > Essentials




Essentials

Sub-categories

Asset PacksBeta ProjectsCertificationTutorial Projects

1-24 of 61 resultsSort byPopularityView Results24

Essentials



Refine by

[clear filters](#)

☐ Hide Purchased Assets

All Categories

☐ 2D (11602)☐ 3D (49318)☐ Add-Ons (127)☐ Audio (10672)☐ Decentralization (15)☒ Essentials (61)☐ Templates (3050)☐ Tools (10914)☐ VFX (3597)

Pricing

Unparalleled platform support – Create once, ship anywhere

iOS

android

Windows



WebGL

PS4

PS5

XBOX ONE

SERIES X|S

androidtv

tvOS

NINTENDO
SWITCH

ARCore

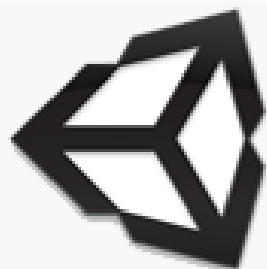
Microsoft
HoloLens

magic
leap

Meta



INTERACTIVE
MEDIA



Unity interface

iNTERACTIVE
MEDiA

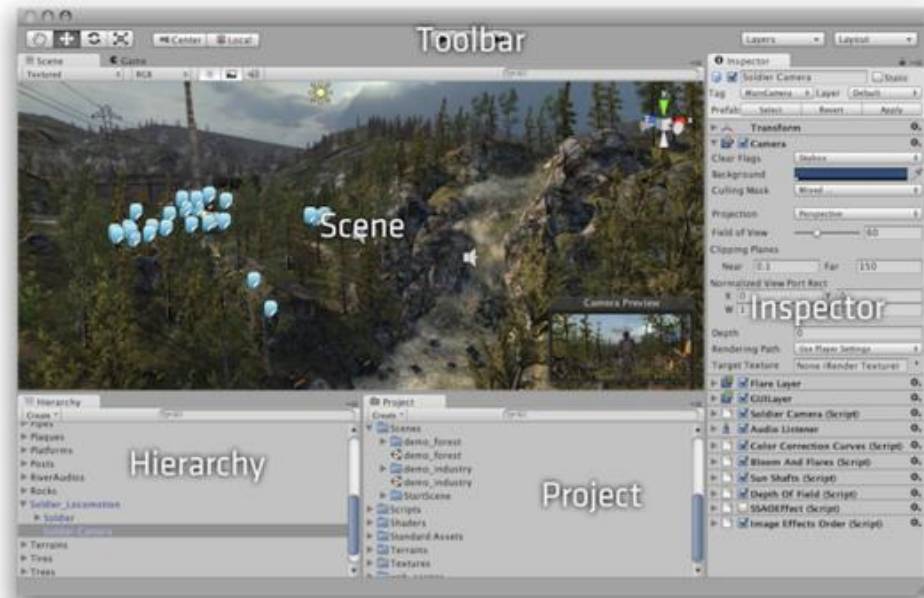
Unity3D IDE Interface



<http://docs.unity3d.com/Manual/LearningtheInterface.html>

Unity3D IDE Interface

- ToolBar – 最常用的幾個操作選項
- Scene View – 擺放遊戲元件(GameObject)，用以架構場景
- Game View – 執行時，實際的遊戲畫面(由Camera的視角決定)



Unity3D IDE Interface

- Hierarchy View – 所有存在目前場景的GameObject列表
- Project View – 目前 Project 所使用到的所有 Assets
 - 所有在 assets folder 下的檔案包括scenes, scripts, 3D models, textures, audio files. and prefabs所有檔案
- Inspector View – 顯示目前選到的GameObject的屬性表



Scene view (3D Space)

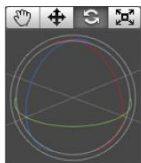


Hand tool : 位移視角(熱鍵:Q)，持續按住滑鼠中鍵有相同效果



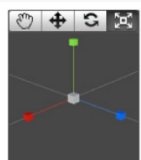
Translate Tool - Hotkey "W"

移動場景中的物體(熱鍵:W)



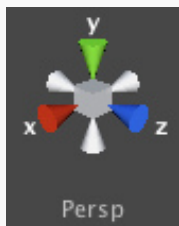
Rotate Tool - Hotkey "E"

旋轉場景中被選定的物件(熱鍵:E)

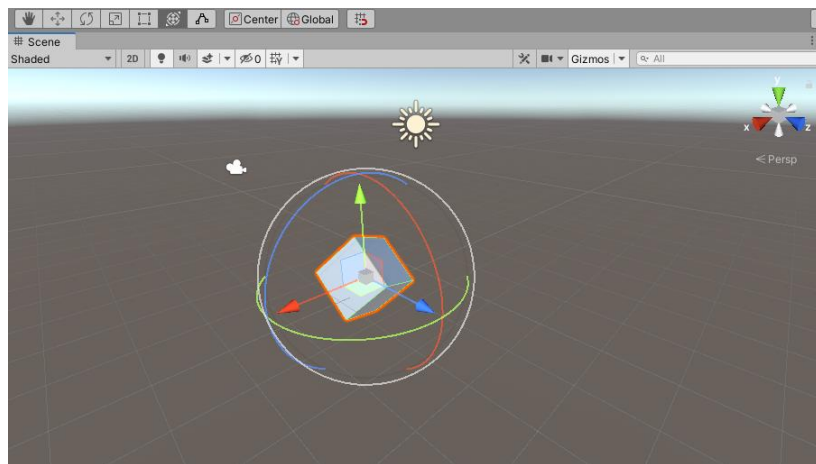


Scale Tool - Hotkey "R"

縮放場景中被選定的物件(熱鍵:R)



Scene Gizmo: 快速轉換視角(前視、後視、俯視...)

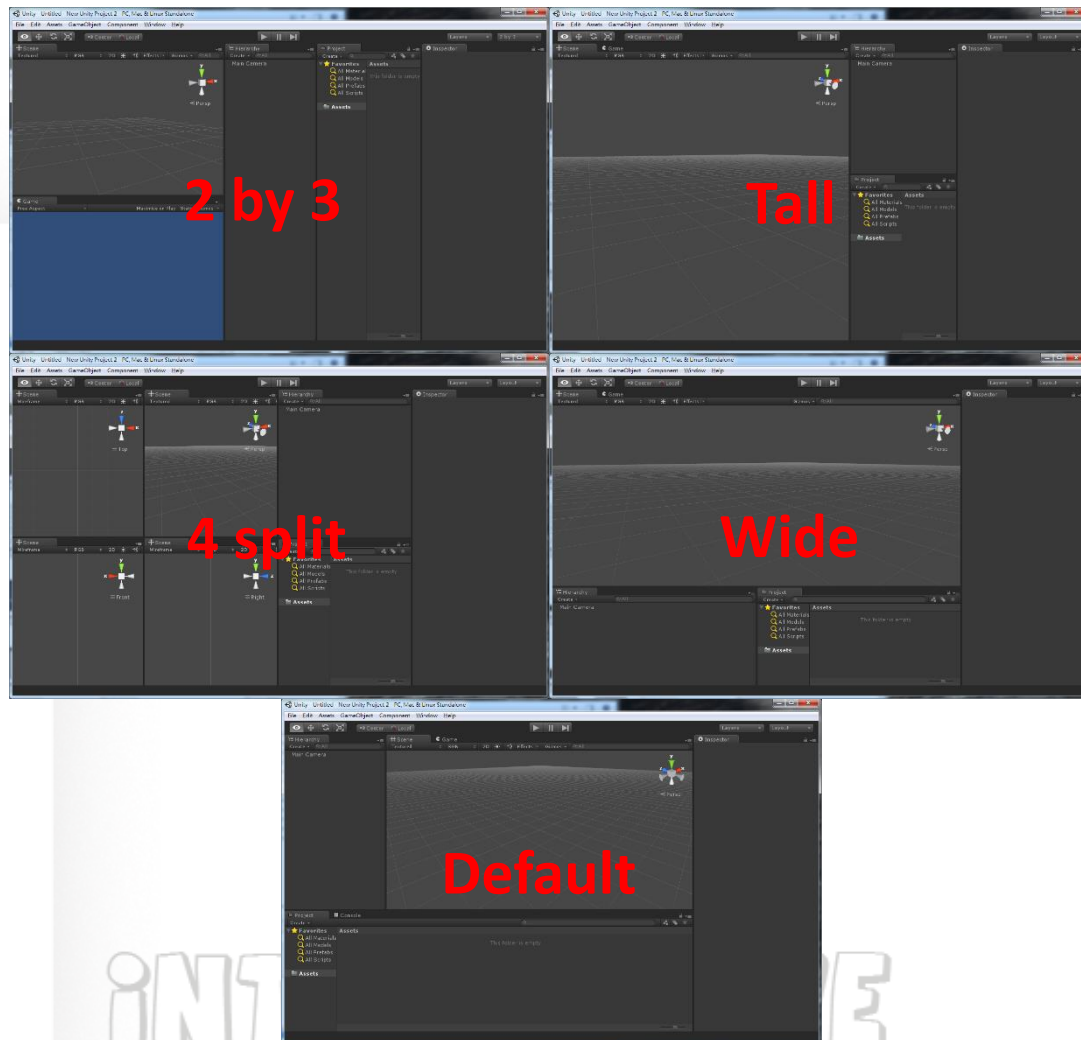
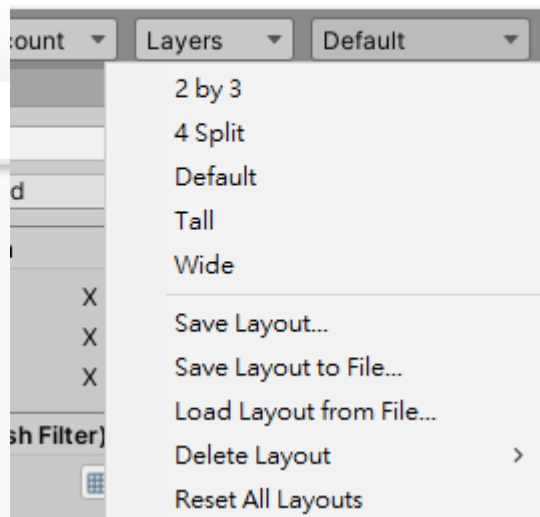


INTERACTIVE
MEDIA

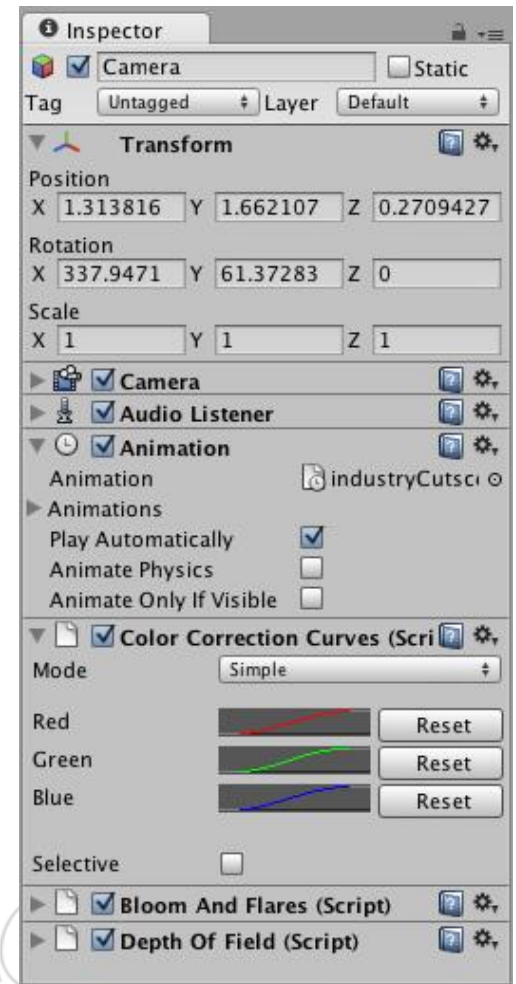
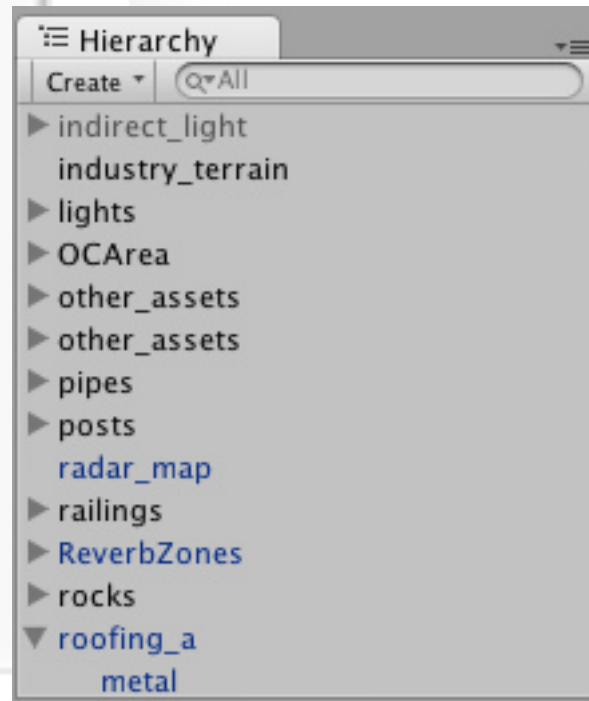
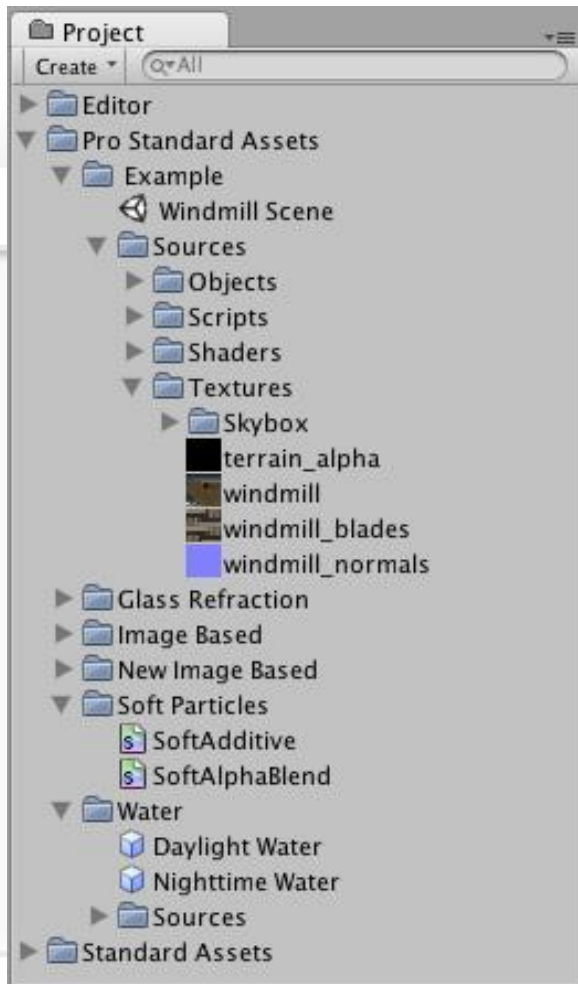
Scene view (3D Space)

旋轉場景視角	按住Alt + 滑鼠左鍵
平移場景視角	按住滑鼠中鍵
縮放場景視角	按住Alt + 滑鼠右鍵 or 滑鼠滾輪
移動場景視角 前進/後退/左移/右移	按住右鍵 + W/S/A/D

Layout



Project, Hierarchy, Inspector



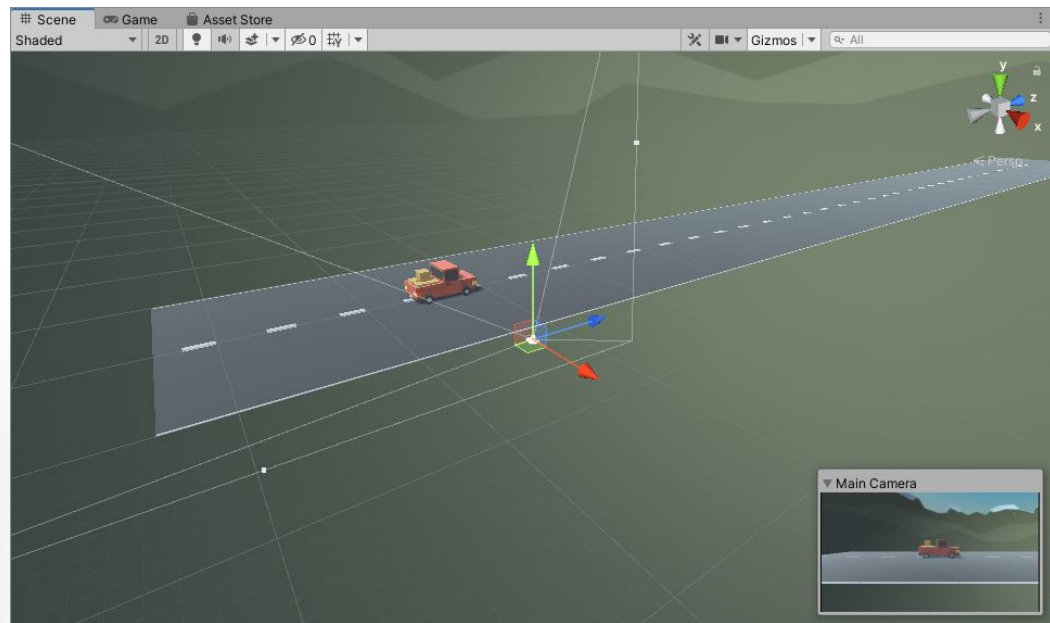
試玩



開新專案
– 下載 [package](#)

– Import!

– Asset > Scenes > Prototype 1



INTERACTIVE
MEDIA



Unity中重要的名詞介紹

iNTERACTIVE
MEDiA

重要名詞說明



Project vs Scene

– Project：遊戲專案, 如「坦克大決戰」



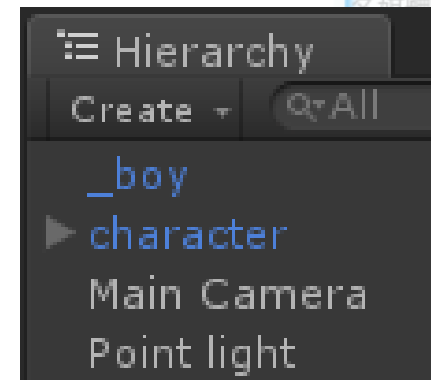
坦克大決戰
檔案資料夾



Level 1.unity
Unity scene file
11.4 KB

– Scene：關卡場景(Level)，一個坦克大決戰可能有五個場景關卡

重要名詞說明



GameObjects

- 任何在遊戲中的物件，如模型、光、攝影機、粒子效果...

Components

- 遊戲中的物件的組成元素，如音樂、碰撞體...，用來控制GameObject的行為，每個GameObject可以包含多個不衝突components

Transform

- Transform是GameObject的Component的一種，Transform描述GameObject的位置(Position)、旋轉(Rotate)、縮放(Scale)，每個GameObject都會包含這個元件

重要名詞說明



Asset

- 在project view會呈現的檔案物件，都稱為asset，包含 scenes, scripts, 3D models, textures, audio files



Prefabs

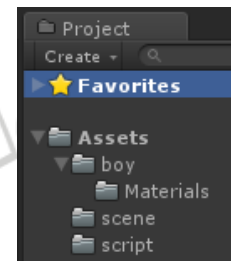
一種在場景中重複使用的物件，如子彈或敵人，通常會在遊戲執行中才動態產生或釋放



UnityPackage

Unity使用的package打包格式，可以把所有專案內用到的asset包含場景全部打包存成單一檔案，副檔名為 *.unitypackage

project view



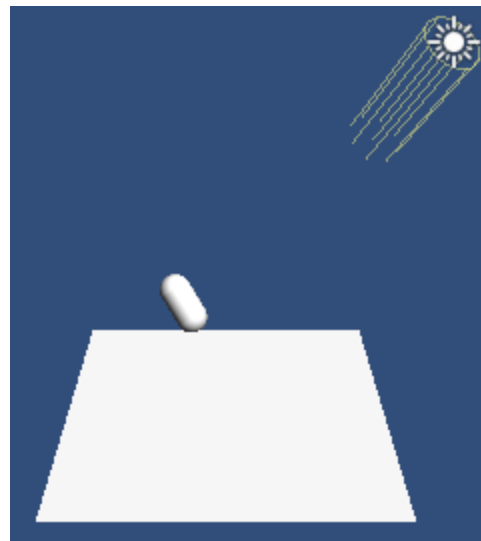
UnityProject.unitypackage

重要名詞說明

📎 Lights : 場景中需要有光源才能讓世界亮起來



No light



Add direction light

INTERACTIVE
MEDIA

重要名詞說明



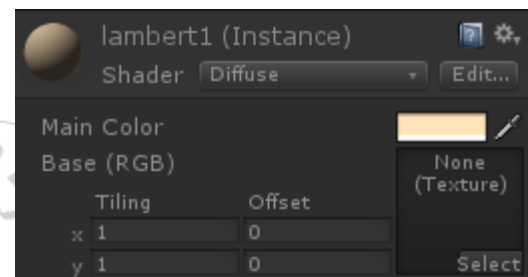
Rigid body & Collider

物理碰撞體, 給予物體後才會有基本的物理碰撞性質



Material

- 材質, 用來表現物體的性質, 通常材質是用來表現物體的表面, 如貼圖, 反光或效果,
- 另外有一種叫physic material, 用來表現物體物理性質, 如彈力, 摩擦力等





Unity遊戲設計架構

iNTERACTIVE
MEDiA

Unity遊戲設計架構

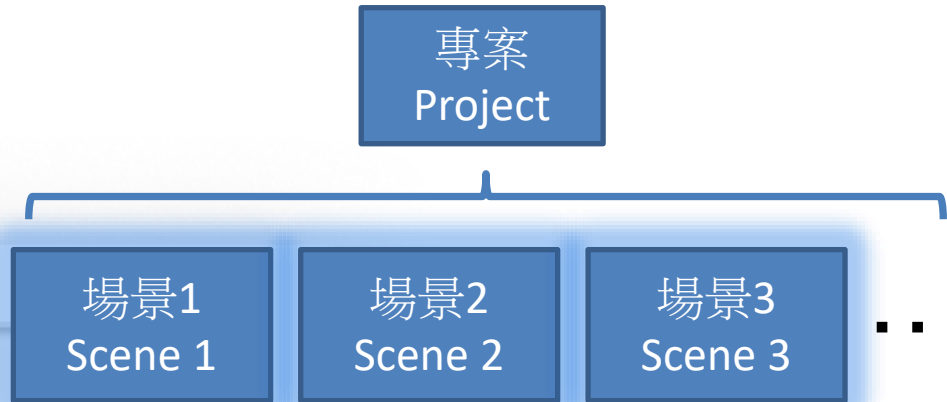


專案檔(Project)

專案
Project

- 一個遊戲
- 一個完整目錄的檔案架構，無單一代表檔案
- 一個Unity視窗同時只能開啟一個專案
- 一個專案同時只能由一個Unity視窗開啟

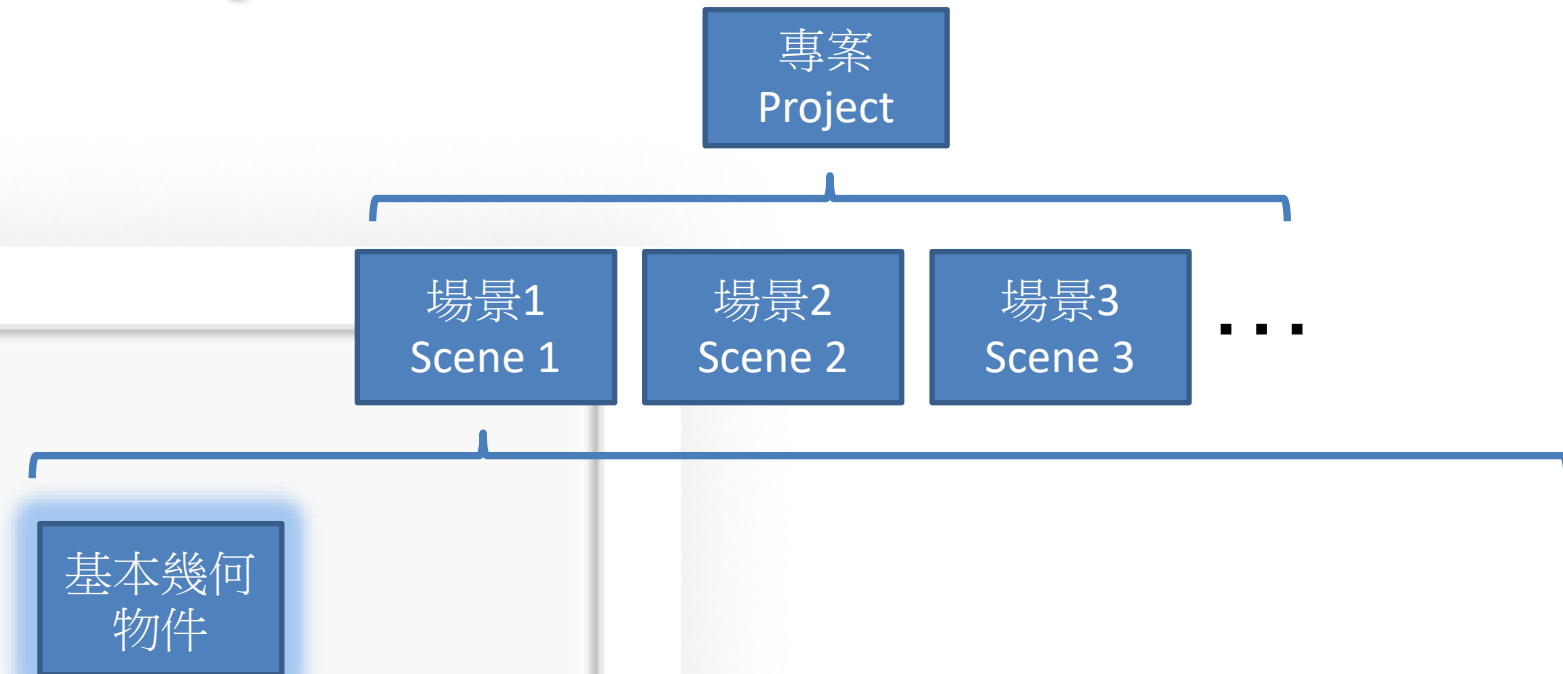
Unity遊戲設計架構



場景檔(Scene)

- *.Unity
- 單一關卡
- 關卡內的場景擺設及物件互動資訊的記錄
- 一個專案下可有多個場景

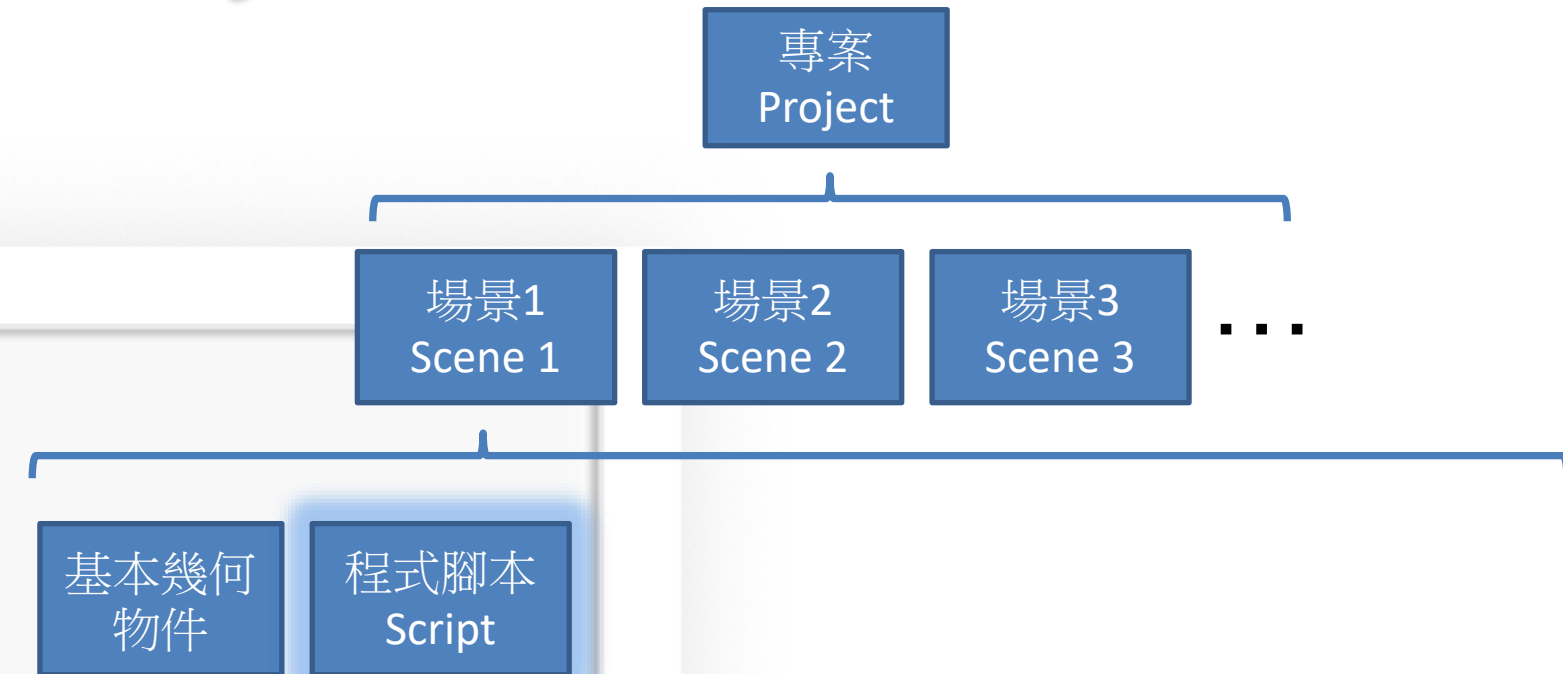
Unity遊戲設計架構



基本幾何物件

- 長方體、球體、膠囊體、圓柱體、平面...
- 可自由調整大小及位置

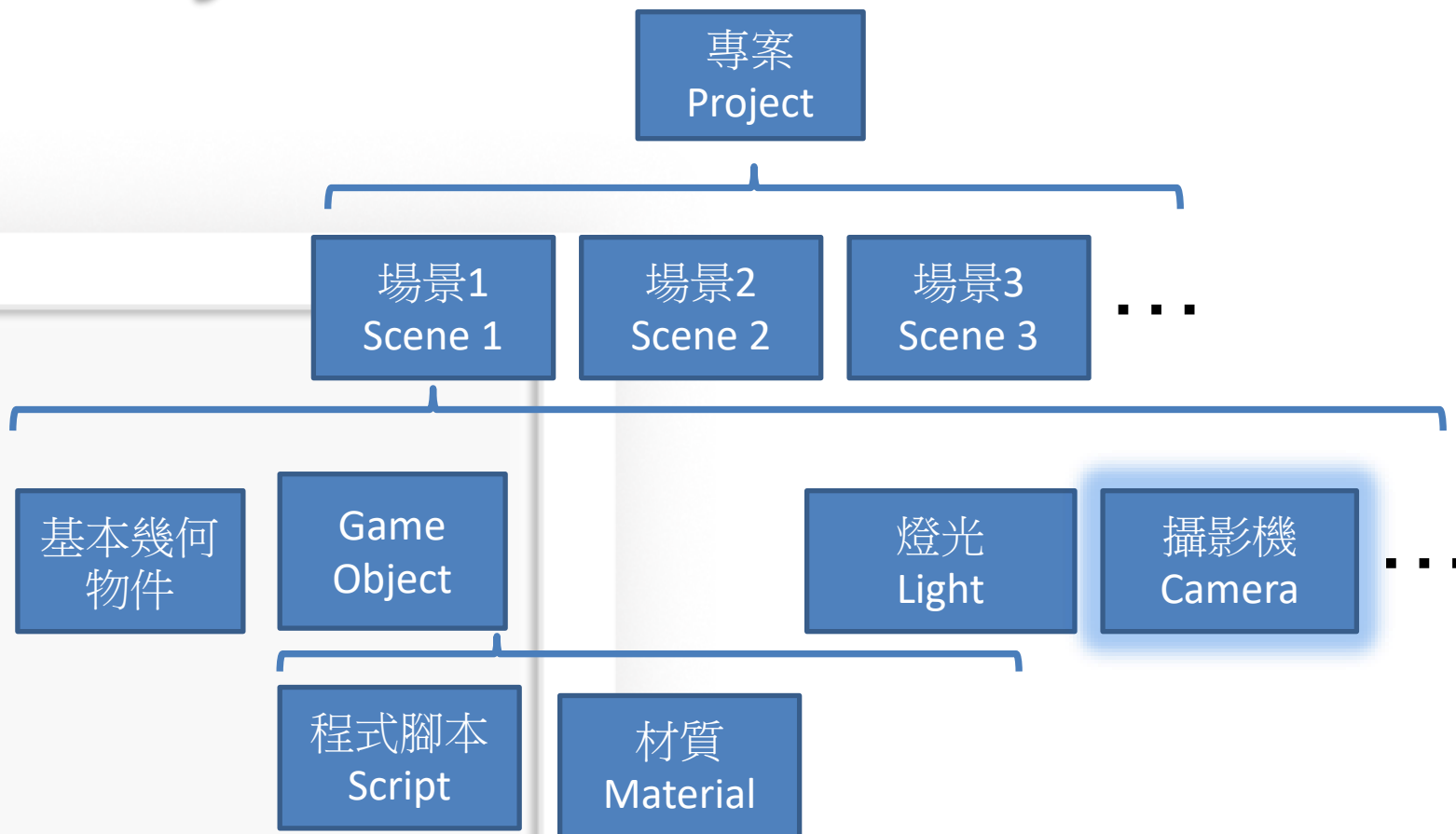
Unity遊戲設計架構



程式腳本(Script)

- 場景內元件的控制
- C#、Shader

Unity遊戲設計架構





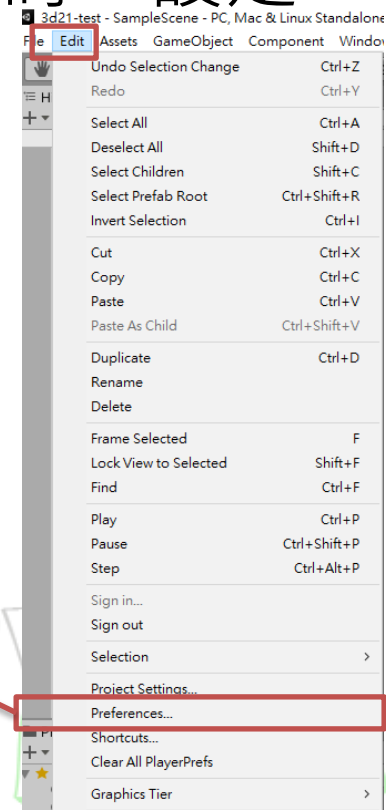
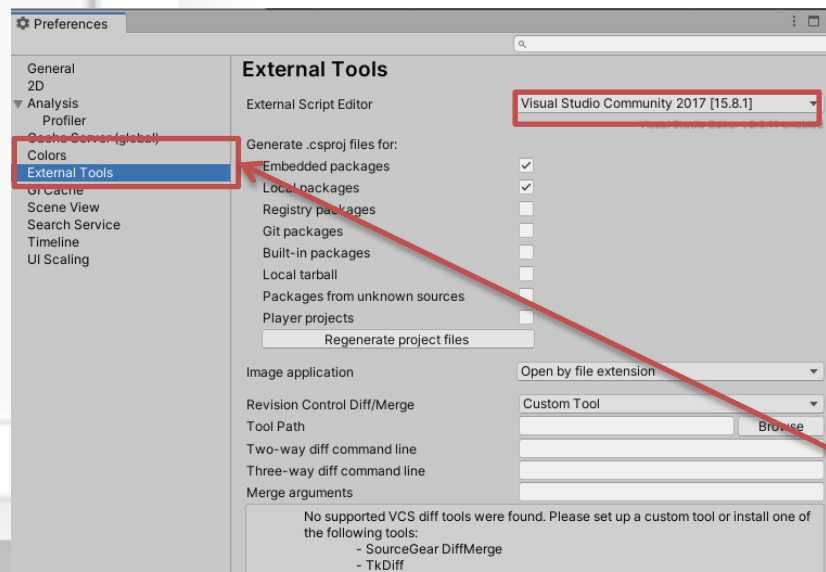
Script(C#)

INTERACTIVE
MEDIA

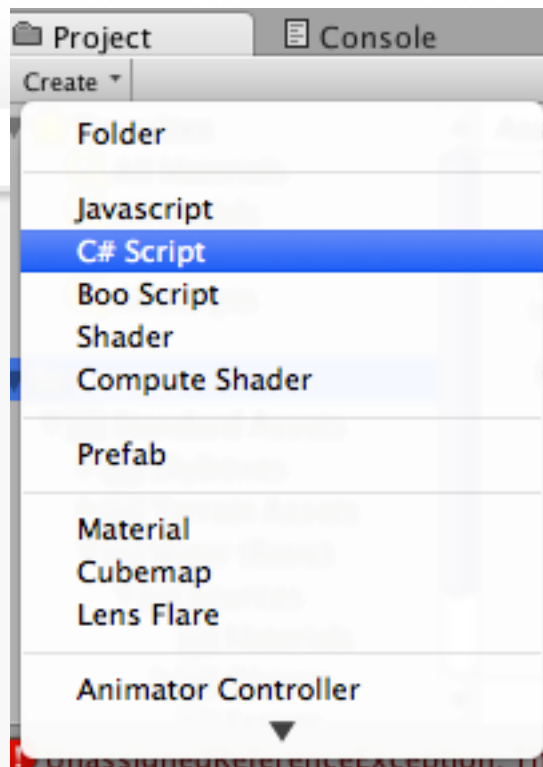


Script

- 大部份的物件都需要用腳本控制
- 有C#, Javascript(我們使用C#)
- 預設編輯器有Visual Studio (Unity下載時，設定add-on，一併下載)
- 可自訂預設的編輯器



Hello World!!



```
1 using UnityEngine;
2 using System.Collections;
3
4 public class HelloWorld : MonoBehaviour {
5
6     // Use this for initialization
7     void Start () {
8         print ("Hello World!!");
9         Debug.Log ("Hello World!!");
10    }
11
12    // Update is called once per frame
13    void Update () {
14
15    }
16 }
17
```

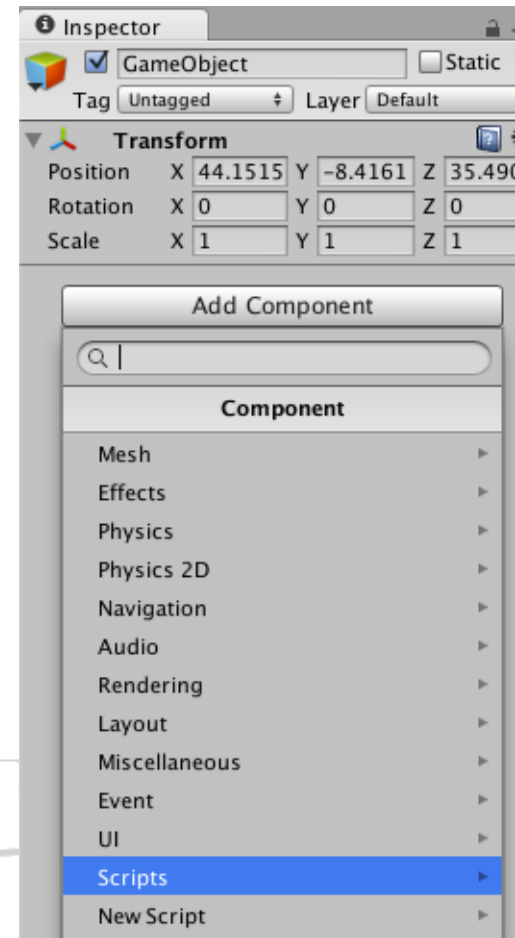
Start裡面兩句結果是一樣的
只是呼叫的函式不同

Hello World!!

✎ 我們要讓腳本能運作就一定要讓它依附在一個物件上

✎ 我們讓他依附在Empty這個物件上有兩個方法

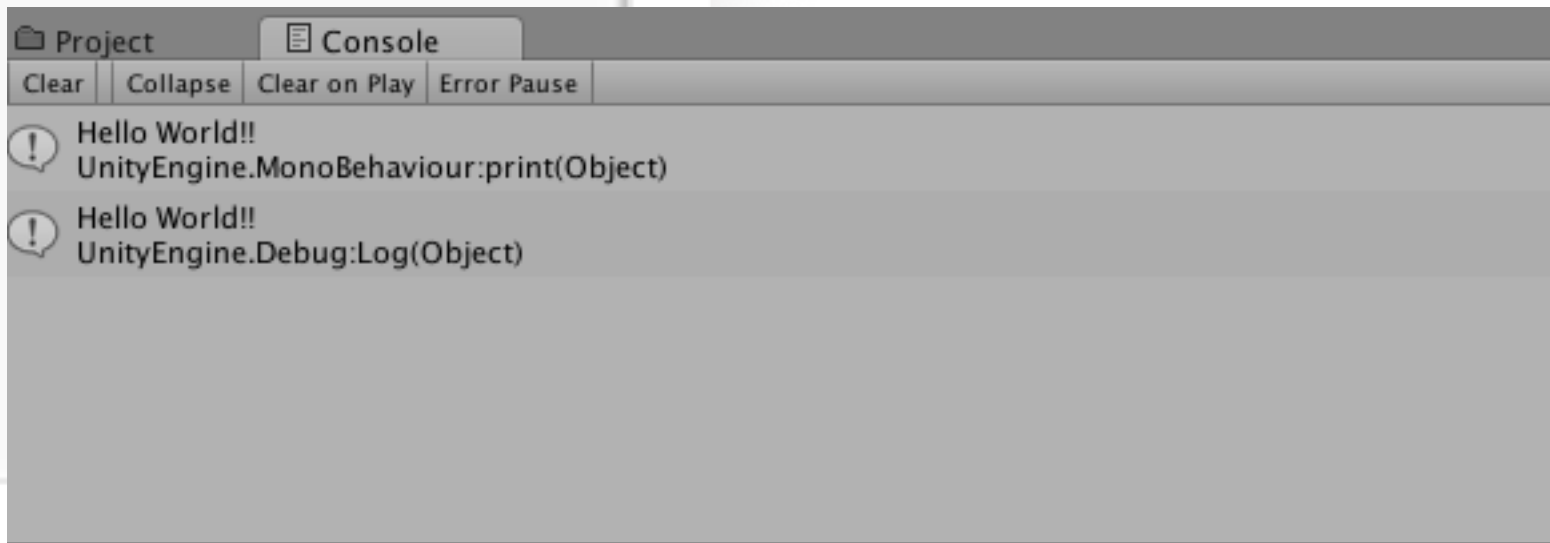
1. 從add component加
2. 直接把腳本拉到物件上



INTERACT

Hello World!!

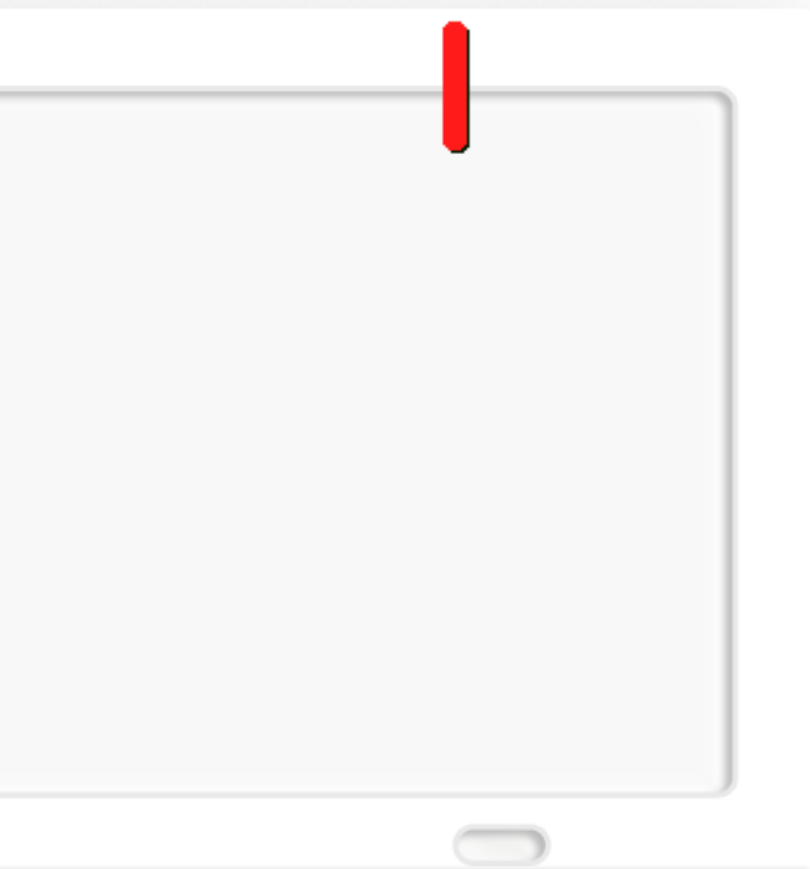
- 執行之後所有會印出來的東西都會在 **Console** 裡面顯示
- Error 和 Warning 也是在這裡顯示



Assets



[falling.zip](#)

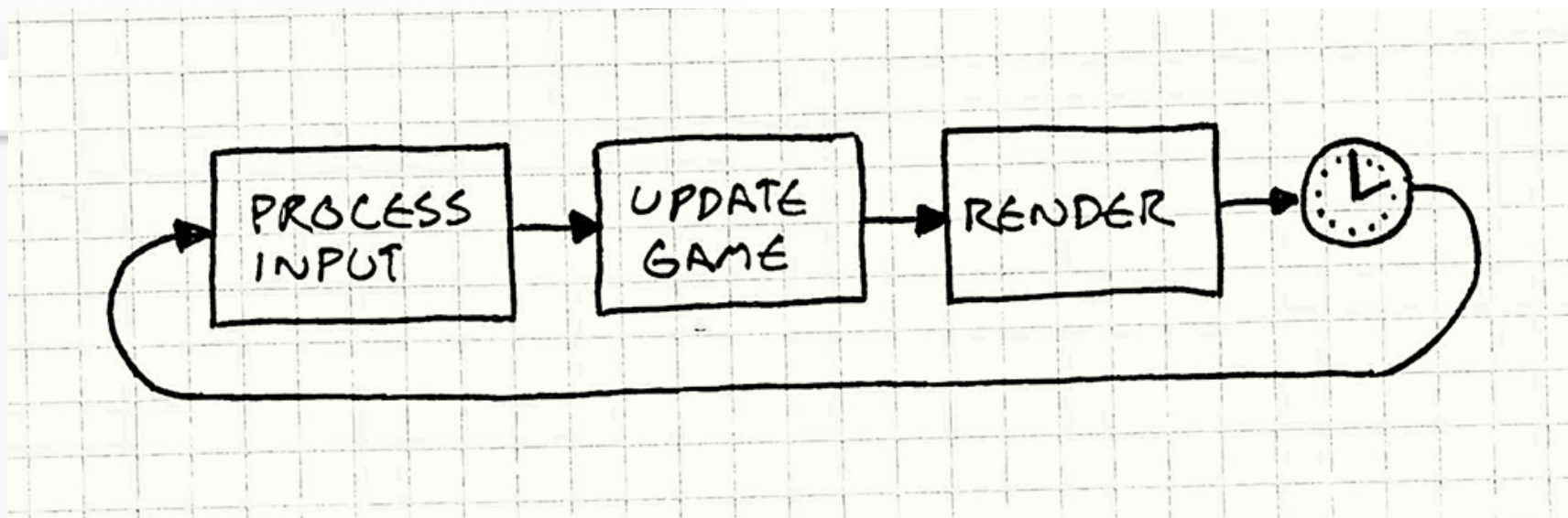


iNTERACTIVE
MEDiA

Transform class

- 📎 `this.transform.position = new Vector3(0, 0, 0)`
//將此物件位置設在(0, 0, 0)
- 📎 `this.transform.position.x` //取得此物件x座標
- 📎 其他使用方法可參考以下網址
<https://docs.unity3d.com/ScriptReference/Transform-position.html>

Game loop



<http://gameprogrammingpatterns.com/game-loop.html>

基本功能函式

函數名	被呼叫時機
Awake()	在脚本(Script)檔被創建並載入場景的時候呼叫。
Start()	在創建後的下一個Update()被呼叫前執行。
Update()	每個禎(frame)更新時呼叫。
FixedUpdate()	在固定的時間呼叫。
OnGUI()	用於處理GUI事件，每個禎(Frame)更新時執行。
OnCollisionEnter()	碰撞體(Collider)/剛體(Rigidbody)和其他的碰撞體(Collider)/剛體(Rigidbody)接觸的一開始執行。
OnCollisionExit()	當碰撞體(Collider)/剛體(Rigidbody)和其他的碰撞體(Collider)/剛體(Rigidbody)終止接觸的瞬間執行。
OnCollisionStay()	當碰撞體(Collider)/剛體(Rigidbody)和其他的碰撞體(Collider)/剛體(Rigidbody)終止接觸的瞬間執行。
OnTriggerEnter()	當本身碰撞體進入觸發器(Trigger)時執行。
OnTriggerExit()	當本身碰撞體離開觸發器(Trigger)的瞬間執行。
OnTriggerStay()	本身碰撞體與觸發器(Trigger)持續接觸的情況下，每個禎(Frame)執行。

輸入控制

- GetKey(): 取得按鍵(持續按會連續動作)
- GetKeyDown(): 取得按鍵(持續按只會動作一次)
- KeyCode.W: W鍵(W可以換成不同的按鈕)
- gameObject.transform: 對這個物件做指令
後面跟OpenGL一樣是接translate(), rotate(), scale()

```
if (Input.GetKey (KeyCode.A)) {  
    this.transform.Rotate(0,-5,0);  
}  
  
if (Input.GetKeyDown (KeyCode.F)) {  
    this.transform.localScale=new Vector3(0.5f,0.5f,0.5f);  
}
```

```
if (Input.GetKey (KeyCode.S)) {  
    this.transform.Translate(0,0,-0.1f);  
}
```




Reference

iNTERACTIVE
MEDiA

Reference

 Unity3D website: unity3D.com
– <https://unity3d.com/>

 UnityCommunity
– <https://unity.com/community>