

Guida agli Script

Indice

1. Character
2. Camere
3. Teleport
4. Torcia e Pick Up
5. Oggetti Nascosti

Character

Il Personaggio ha il suo prefab che si trova nella cartella:

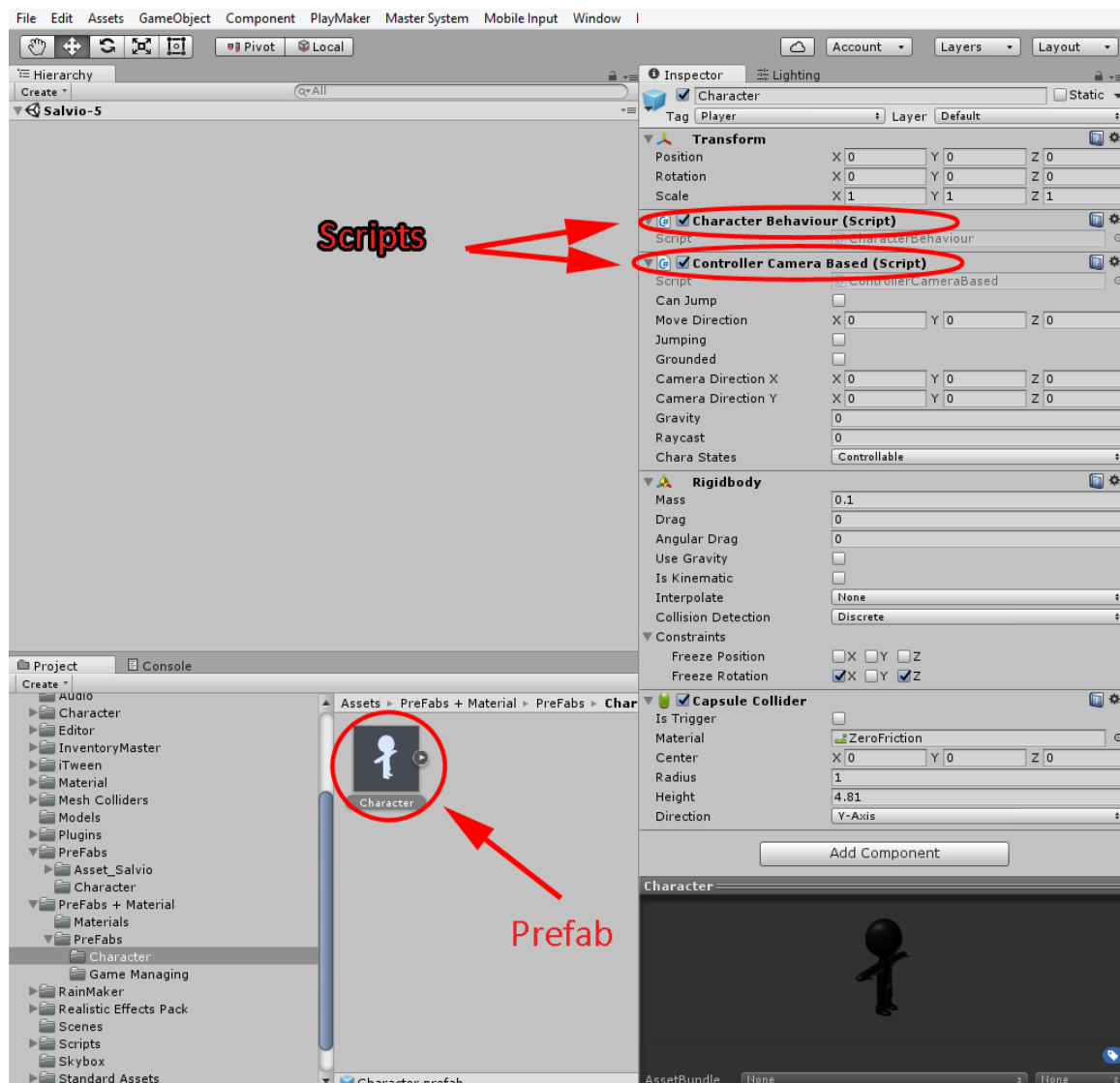
Asset/PreFabs + Material/PreFabs/Character

Il Prefab ha già i 2 script chiave per il personaggio:

CharacterBehaviour.cs e ControllerCameraBased.cs

In più il rigidbody e il capsule collider*

Non ci può essere più di un Character nella scena



* Posso esserci modifiche in fasi successive di scripting (la mesh verrà aggiornata col quella del personaggio definitivo)

Camere

Sono state scriptate 2 macro categorie di camere:

Platform (Plt) o Puzzle (Pzl):

Le camere Puzzle vengono divise in altre categorie:

Frontali (FRT) o Lateralì (SIDE)

E ulteriormente in altre 2:

Look legato alla posizione del personaggio (Free)

Look legato all'altezza del terreno (Lock)

Gli Script corrispondenti ai vari tipi di camere sono:

Platform:

PltCamBehaviour.cs

Puzzle:

PzlCamFreeFRTBehaviour.cs

PzlCamFreeSIDEBehaviour.cs

PzlCamLockFRTBehaviour.cs

PzlCamLockSIDEBehaviour.cs

Caratteristiche delle camere:

- Platform

La camera Platform segue il personaggio su tutti gli assi di movimento (X,Y,Z) e mantiene una distanza fissa dal personaggio che viene salvata al momento che la camera viene attivata, quindi basta posizionarla in modo tale che la già dalla preview si è certi della prospettiva e distanza che la camera manterrà in runtime.

- Puzzle

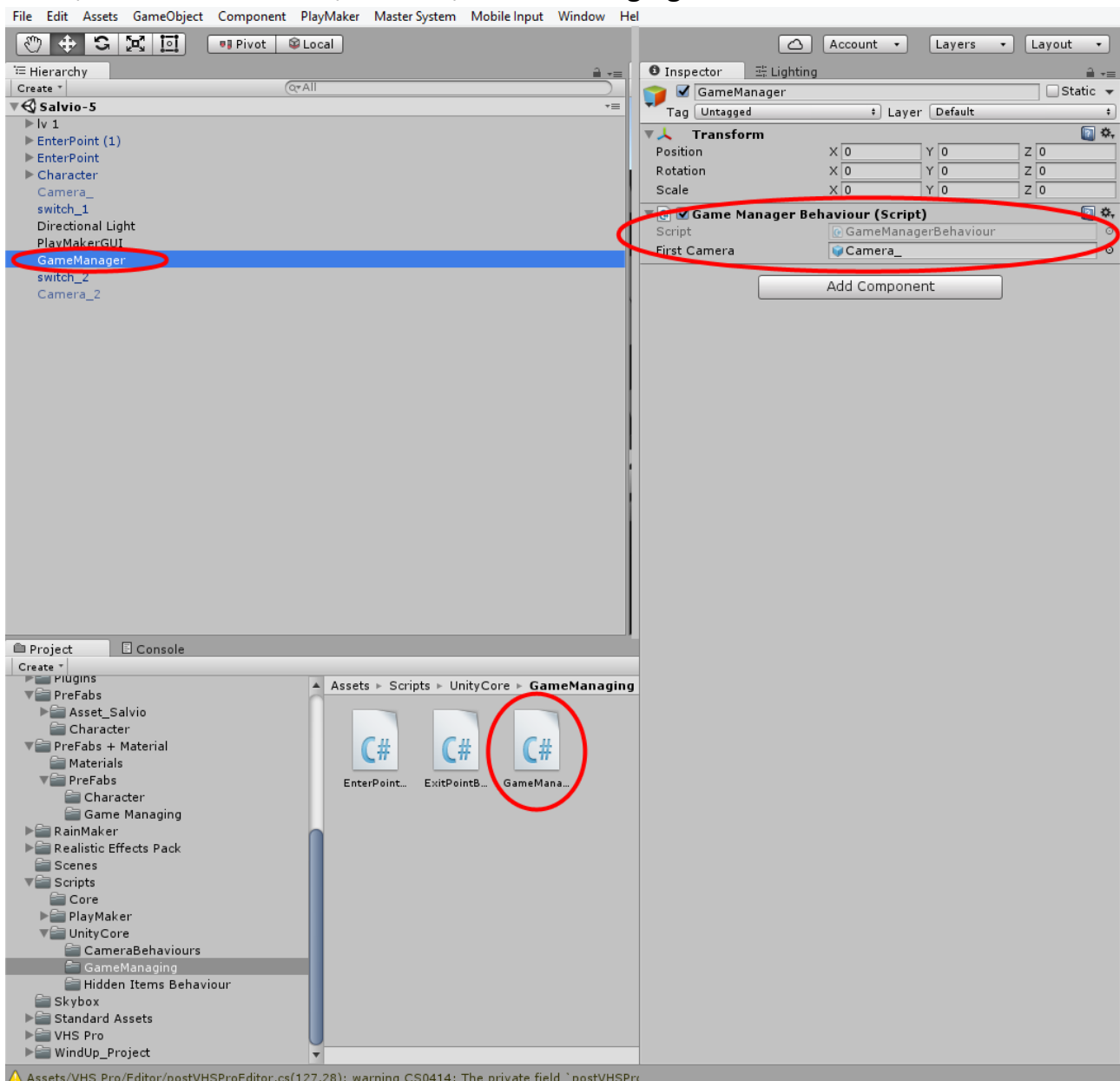
La camere Puzzle invece hanno un'asse di movimento bloccato. In base se sono Front o Side (dettato dalla rotazione della camera) lo script seguirà il personaggio solo in maniera laterale. Per sapere se la camera che si intende utilizzare è Front o Side basta posizionarla, spostarla sulla sua asse laterale e vedere quale valore di Position cambia: Se cambia la X è Front, se cambia la Z è Side.

La scelta tra Free e Lock è unicamente a scopo di level design, dipende se vi è utile ruotare l'inquadratura quando il personaggio si alza da terra oppure tenerla fissa.

Come settare le camere all'interno della scena:

- Game Manager

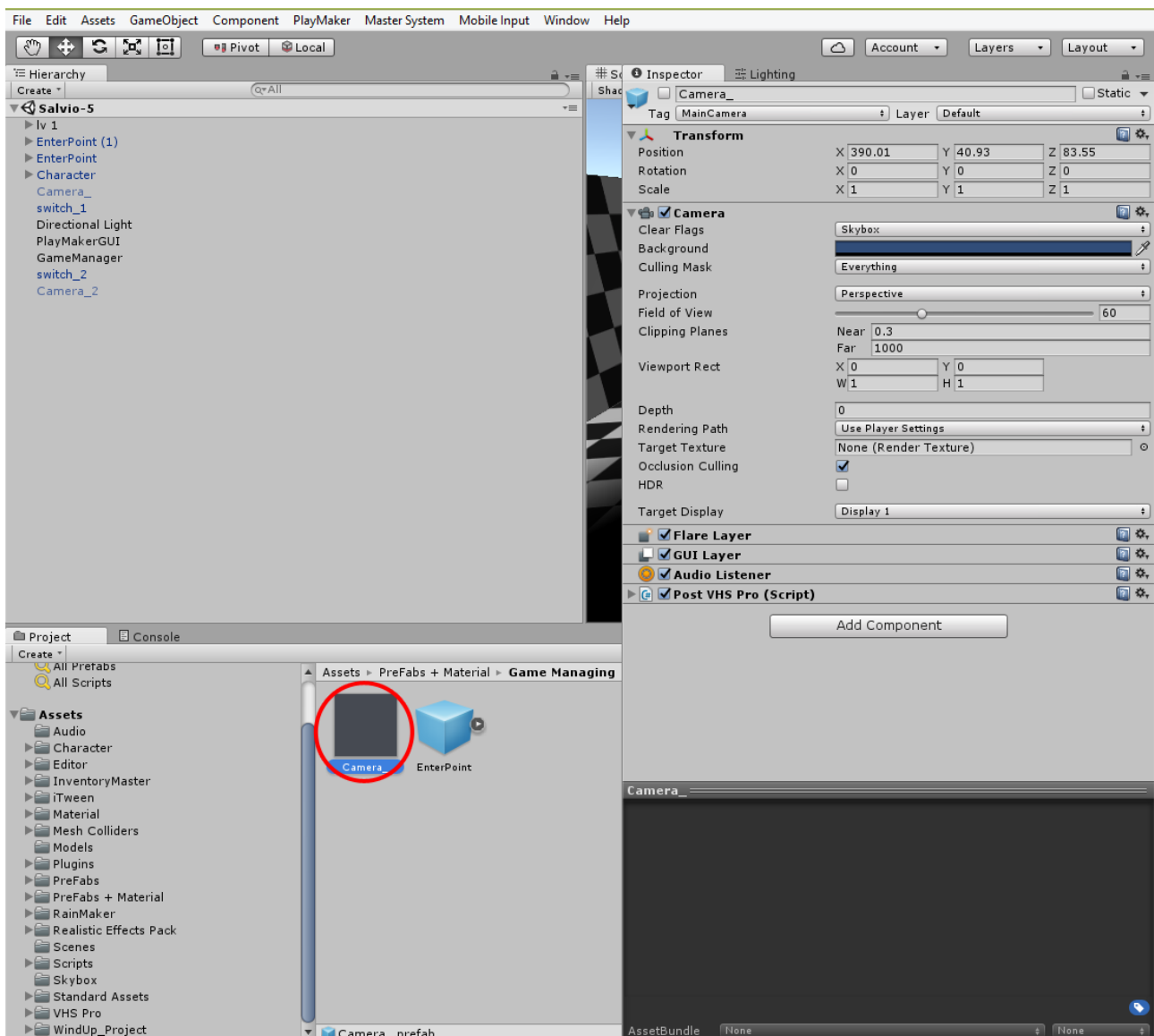
È fondamentale che ogni scena abbia il Prefab “GameManager” che si trova cartella:
Asset/PreFabs + Material/PreFabs/Game Managing



Nello script del Game Manager al parametro “First Camera” passate direttamente dalla gerarchia la prima Camera che deve essere attiva all’avvio della scena.

- Inserire nuove camere:

Per inserire una nuova camera inserite nella scena una istanza del Prefab “Camera_” all’interno della cartella Asset/PreFabs + Material/PreFabs/Game Managing (Trascinando il prefab dalla cartella alla scena)



Rinominate la Camera con “Camera_X” dove X è il numero della camera (Es: la prima camera che inseriremo sarà “Camera_1”, la successiva “Camera_2” e così via).

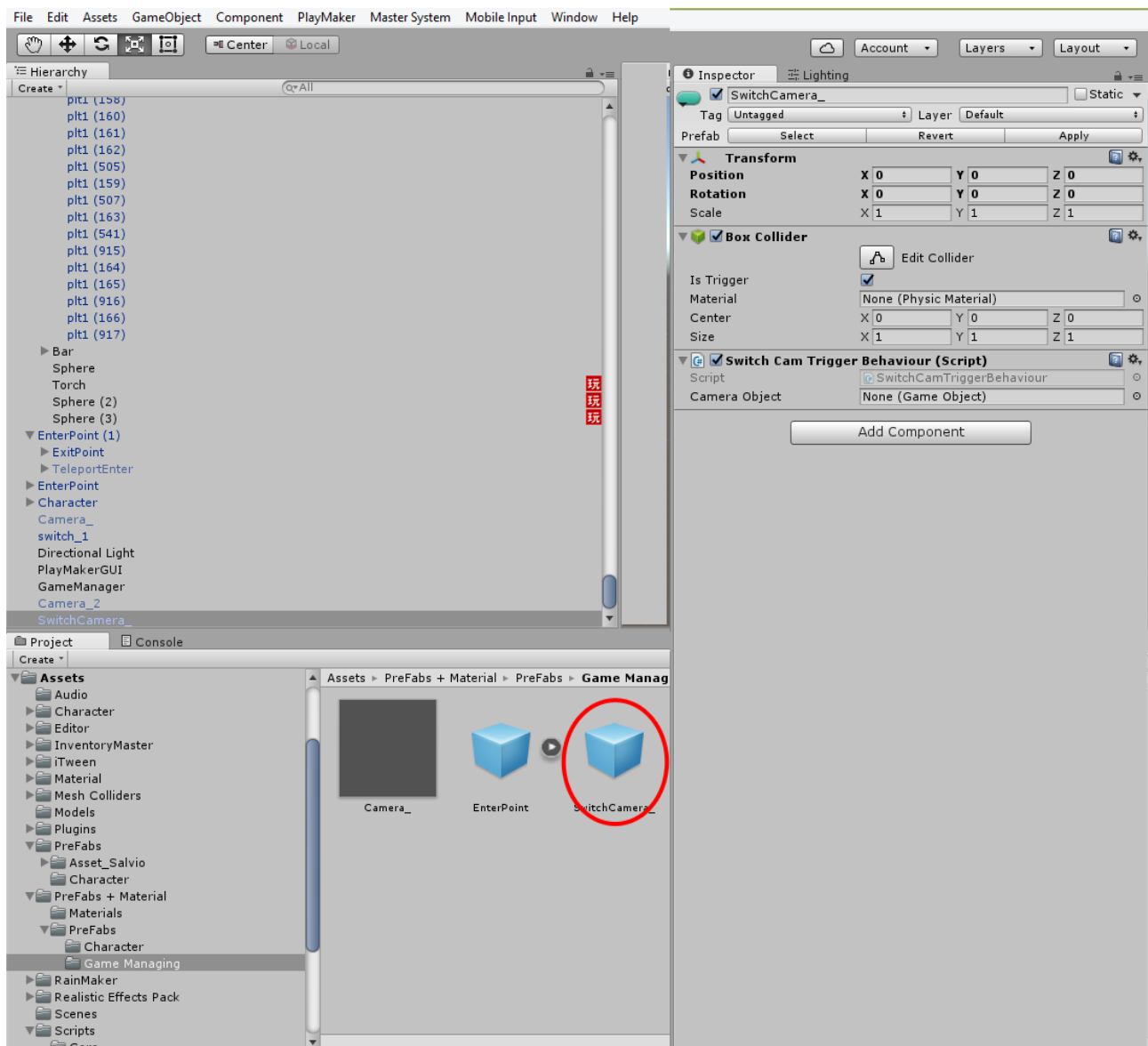
Posizionate la Camera come desiderate e assegnatevi lo script corrispondete al tipo di movimenti che dovrà fare una volta attivata.

NOTA BENE:

Il prefab di base ha il gameobject disattivato, per vedere la camera preview bisogna attivarlo. Una volta posizionata a dovere RICORDATE di disattivare il gameobject (la spunta di fianco al nome) altrimenti ci saranno più camere attive che causeranno molteplici errori.

- Cambiare telecamere In-Game:

Una volta posizionate le camere va definito in che parti della scena vanno scambiate. Per fare questo utilizzate il prefab "SwitchCamera_" nella cartella Asset/PreFabs + Material/PreFabs/Game Managing:



Rinominate mettendo come numero dopo “_” lo stesso della Camera che deve attivare.

Assegnate al parametro “Camera Object” dello Script SwitchCamTriggerBehaviour.cs la camera che deve essere attivata da questo trigger direttamente dalla gerarchia.

Successivamente posizionate il trigger nella posizione che vi serve con le dimensioni tali da coprire l’area interessata.

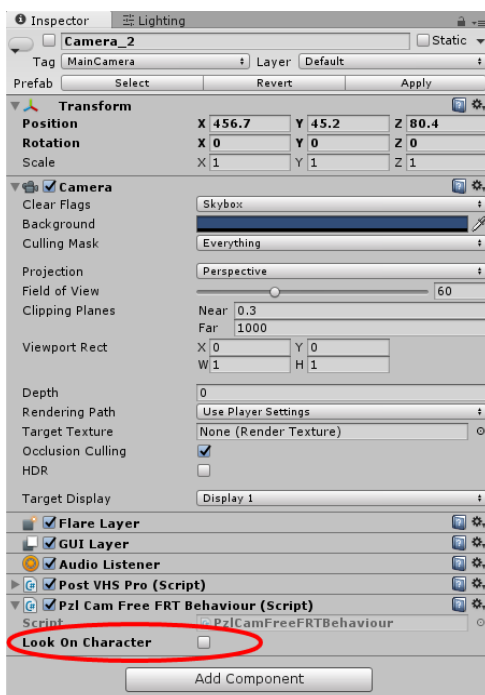
Quando il giocatore entrerà nel trigger che ha una camera impostata differente dall’attuale lo script farà lo scambio.

NOTA BENE:

È utile mettere un trigger anche per l’area di partenza con la prima camera, nel caso il giocatore avanzi ma poi torni indietro, anche se inizialmente è il Game Manager ad attivarla. Ed evitate di mettere SwitchCamera_ che si toccano tra loro.

- Look On Character

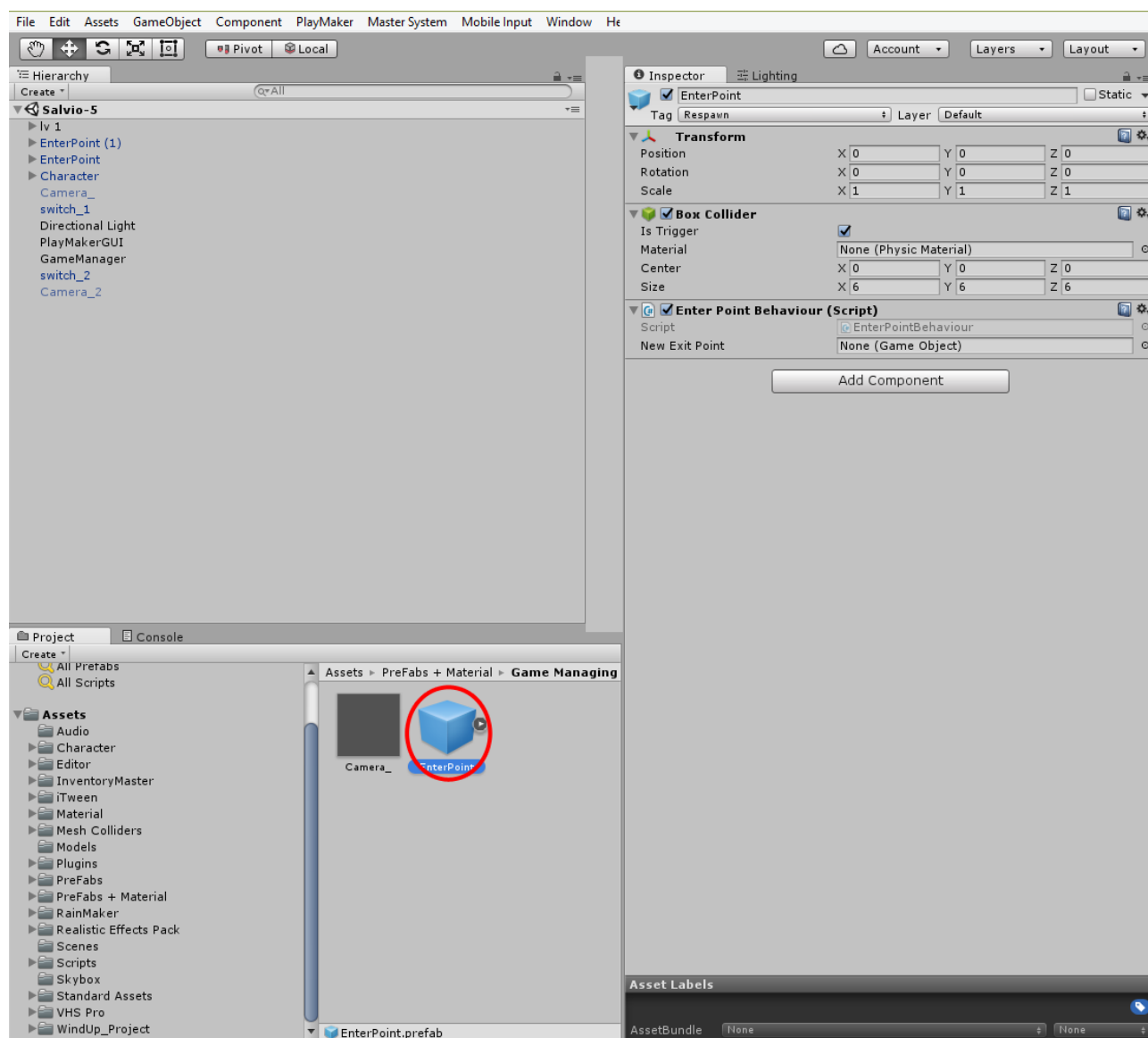
Può capitare che voglia utilizzare una telecamera in modalità Ortogonale, per evitare la rotazione della camera se utilizzo uno script Puzzle, mi basta de-spuntare il paramentro “Look On Character” presente nello script.



Teleport

Per inserire un teletrasporto che sposti il personaggio in un'altra area della scena istantaneamente utilizza il prefab "EnterPoint" nella cartella:

Asset/PreFabs + Material/PreFabs/Game Managing



Quando si inserisce nella il Prefab utilizzate un oggetto 3D (Cubo, Sfera o simili) come padre del Prefab nella scena, in assenza del modello del teletrasporto vero e proprio. Quando verrà aggiunto agli asset il Prefab sarà aggiornato con la mesh già inserita.

(Se inserite prima l'EnterPoint nella scena e poi lo parentate ricordate di azzerare la position per metterlo nello stesso punto del padre)

- Struttura EnterPoint

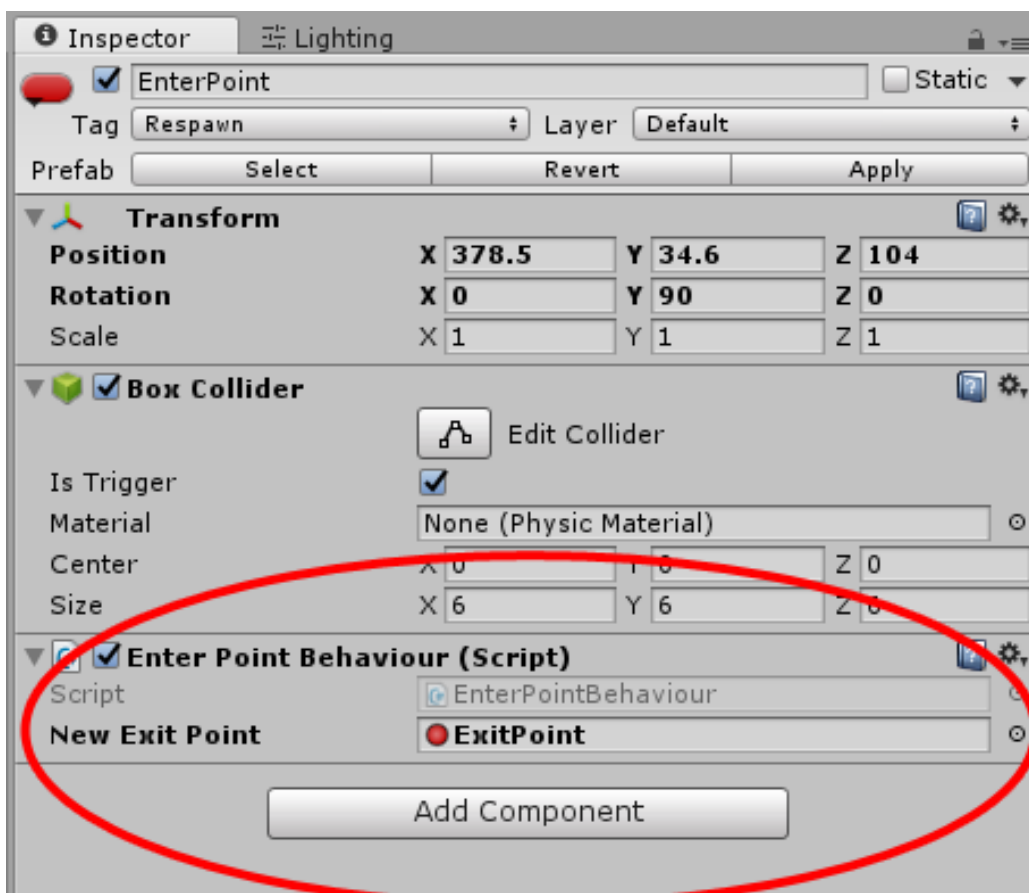
L'EnterPoint al suo interno contiene un ExitPoint, punto di riferimento quando ci si teletrasporta.

Ci sono 2 modi di usare l'EnterPoint:

1. A senso unico
2. EnterPoints comunicanti

1.

Per il teleport di sola andata basta posizionare l'ExitPoint figlio del prefab nella posizione dove riapparirà il personaggio e successivamente passare al prefab nel parametro dello script "EnterPointBehaviour.cs" il gameobject dell'ExitPoint figlio



2.

Per far comunicare tra loro i teleport basta assegnare ad ogni EnterPoint come paramentro l'ExitPoint dell'altro Prefab, in maniera che esempio entrando nell'EnterPoint 1 esco nell'ExitPoint dell'EnterPoint 2 ed entrando nell'EnterPoint 2 esco da quello dell'1. (È anche possibile creare una catena di teletrasporti dove il primo porta al secondo, il secondo al terzo e il terzo al primo per esempio soltanto assegnando gli ExitPoint nell'ordine giusto)

NOTA BENE:

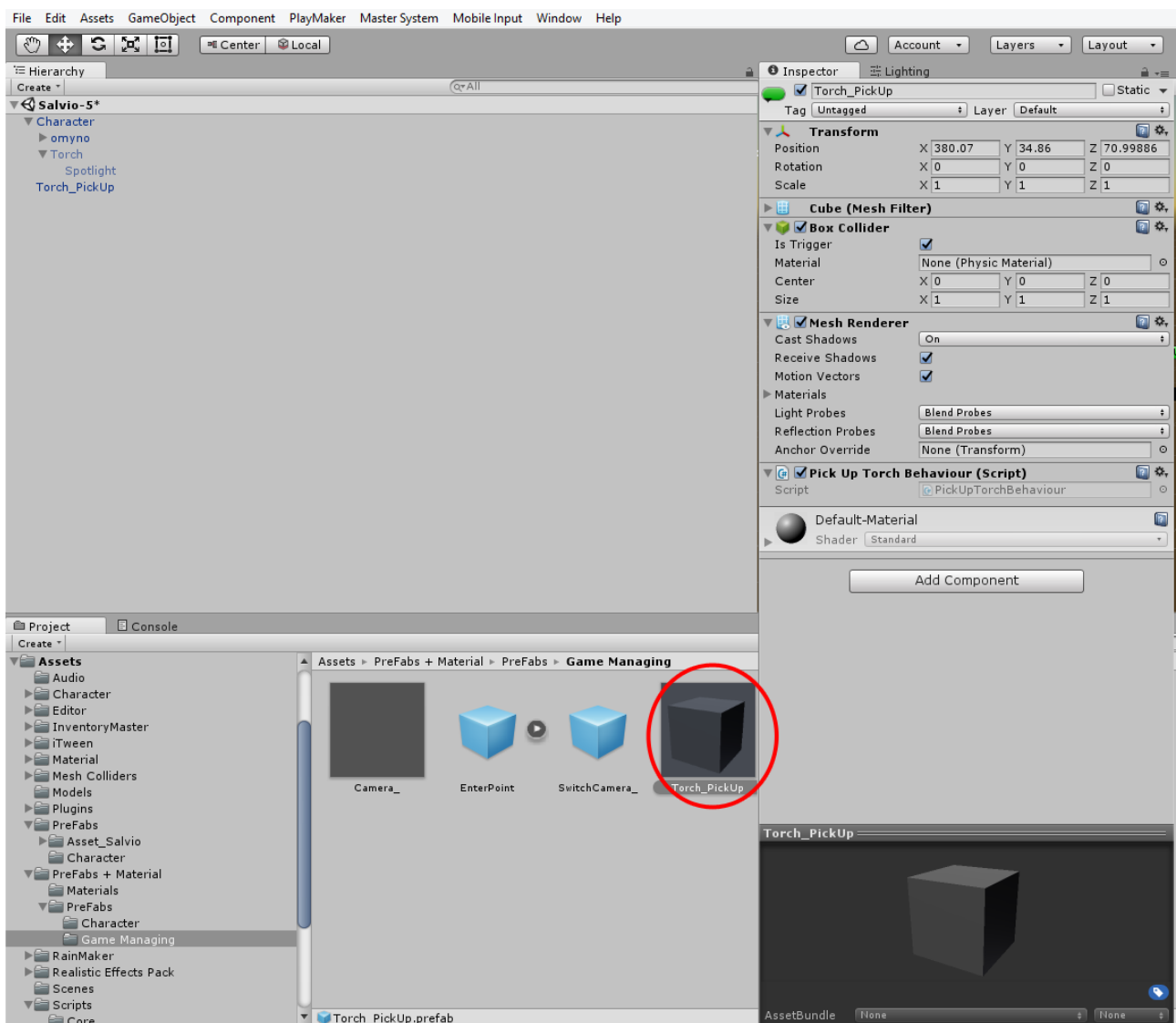
Quando il personaggio viene teletrasportato in una nuova area è consigliabile che tale area abbia già il trigger per lo switch per la camera per evitare movimenti di camera anomali e dare lo scambio di camera simultaneamente al teletrasporto del personaggio.

Torcia e Pick Up

La torcia (o visore) è già presente nel prefab del Character, ma è inizialmente disabilitata. All'interno del livello il giocatore dovrà raccogliere l'oggetto che attiverà la torcia.

- Pick Up

Il Pick Up è un oggetto in scena che una volta il personaggio tocca gli attiva la torcia e si distrugge. Per usarlo basta mettere in scena il Prefab "Torch_PickUp" e farlo arrivare a prendere al personaggio tramite il Level Design.



La Mesh definitiva verrà caricata in seguito.

Oggetti Nascosti

Esistono 2 tipi di oggetti nascosti:

1. Una volta puntati dalla torcia rimangono visibili per sempre
2. Sono visibili solo mentre sono puntati dalla torcia

Per questi 2 tipi ci sono 2 scripts:

1. ItemHiddenBehaviour.cs
2. ItemPermaHiddenBehaviour.cs

Ognuno associato al tipo di oggetto nascosto.

Per rendere un oggetto in scena un oggetto nascosto basta assegnargli lo script corrispondente tramite la voce "Add Component" dell'inspector.

- Muri Segreti

I muri segreti utilizzano uno script che quando sono puntati dalla torcia diventano semi-trasparenti e intangibili così da mostrare passaggi segreti.

Settaggio:

Una volta creato il muro nella scena (che dovrà avere il materiale opaco in comune con i muri della scena *una volta creato il prefab per le strutture non ci sarà bisogno di quest'accortenza*) Bisogna aggiungere al muro che si intende far scomparire lo script : "WallHiddenBehaviour.cs" tramite "Add Component" da Inspector. Successivamente come paramentro dello script il materiale trasparente dagli asset (ancora da creare in maniera definitiva).