5.2 Jump (2a part): Aterratge

Si el player salta i cau sobre una plataforma, hi aterra i el joc continua. Si cau fora de la plataforma, mor.

La caiguda fora de la plataforma ja la tenim gestionada amb l'objecte **DeathTrigger**. Ara falta tractar el cas de caure sobre la plataforma.

Per establir el moment d'aterratge en la plataforma utilitzarem **colliders**. (Una alternativa seria comprovar si el Player té o no moviment vertical i suposar que, si no en té, està sobre el terra. Això no és del tot fiable perquè si es para durant un moment, el sistema detectarà erròniament que no està saltant.)

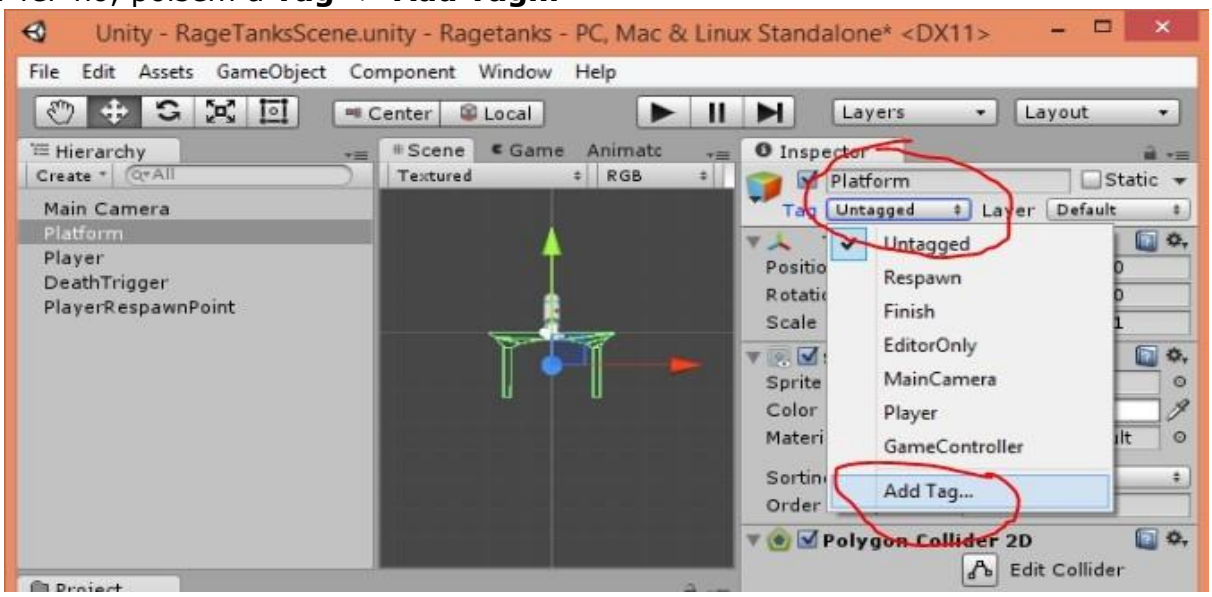
A més utilitzarem **tags**. Tots els objectes en Unity tenen una etiqueta, que s'anomena **tag**. Als tags els podem assignar valors i també llegir-los. **Una de les utilitats típiques dels tags és agrupar objectes**: els objectes que tenen un tag amb el mateix valor es consideren vinculats o que pertanyen a un mateix tipus.
Veure http://docs.unity3d.com/es/current/Manual/Tags.html

5.2.1 Gestionar tag de Platform

En aquest cas, utilitzarem els tags per saber si el Player ha aterrat sobre una plataforma (farem que totes les plataformes comparteixin un mateix valor de tag).

Per veure el tag de l'objecte Platform, primer de tot el seleccionem a la pestanya Hierarchy. A l'Inspector veiem que la propietat Tag té el valor "Untagged". Si polsem el desplegable veiem els **valors de tag que el sistema ofereix per defecte** (Respawn, Finish, etc). També **en podem afegir de particulars** per a la nostra aplicació. Ara afegirem un nou valor "Platform" que assignarem al tag dels objectes Platform.

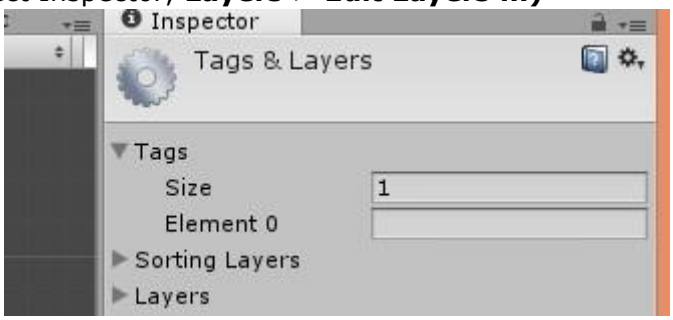
Per fer-ho, polsem a **Tag -> Add Tag...**



... i arribem a la **pestanya Tags & Layers**:

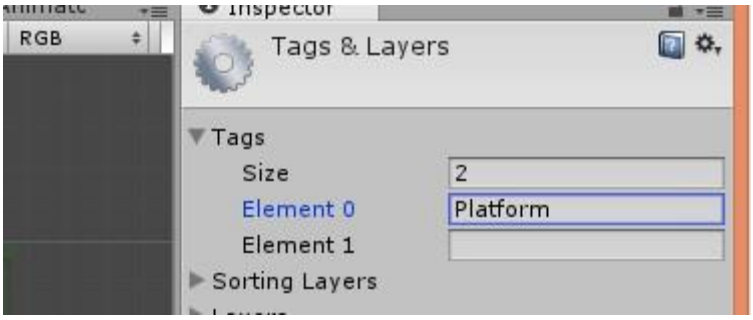
(També podriem arribar aquí des del menú principal **Edit > Project Settings > Tags and**

Layers o des de l'Object Inspector, **Layers > Edit Layers ...)**

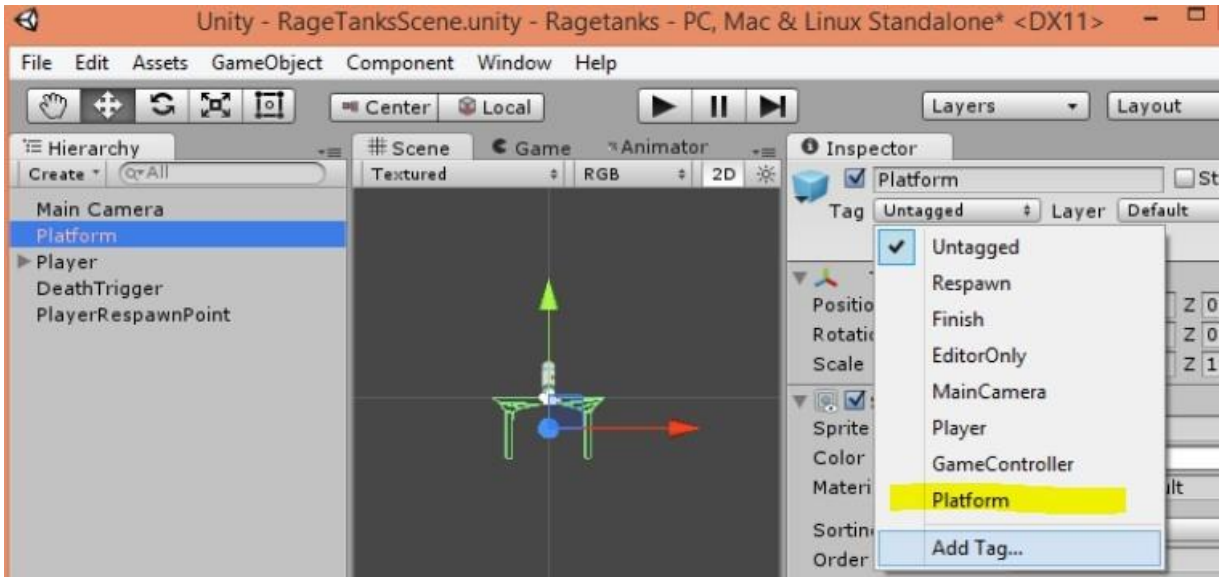


Els valors de tag definits per l'usuari s'emmagatzemen en un vector. Sempre hi apareix un element de més, que està buit, per poder afegir un nou valor.
A la imatge veiem que, inicialment, el vector de valors de tags d'usuari no té cap element informat, només l'element buit (en aquest cas "Element 0") i per això la seva mida és 1 (Size 1). Si hi afegim valors, el camp Size s'anirà incrementant.

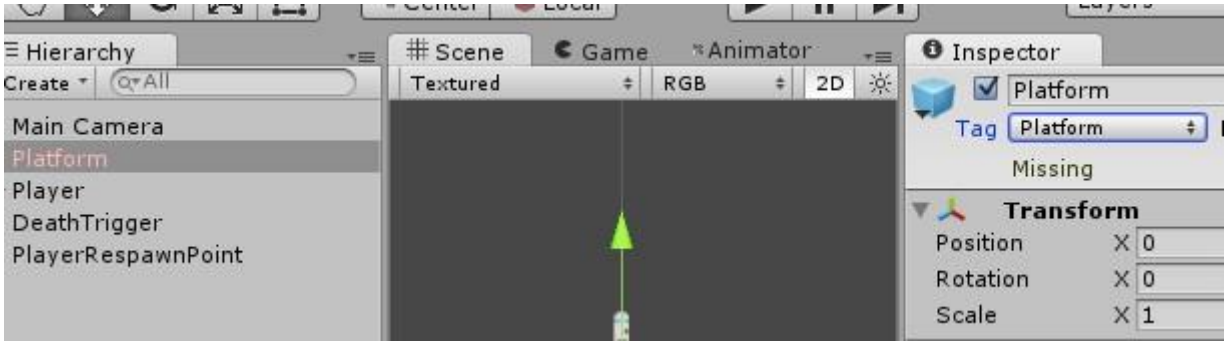
Creem ara un nou valor "Platform".



Seleccionem altra vegada l'objecte Platform a la Hierarchy i, a l'Inspector, veiem que el seu tag encara no té valor (és "untagged") però ja tenim disponible el nou valor "Platform".



El seleccionem i l'hi assignem.



Ara, quan detectem una col·lisió del Player, podrem preguntar si ha estat sobre un objecte Platform simplement consultant el tag de l'objecte col·lisionat.

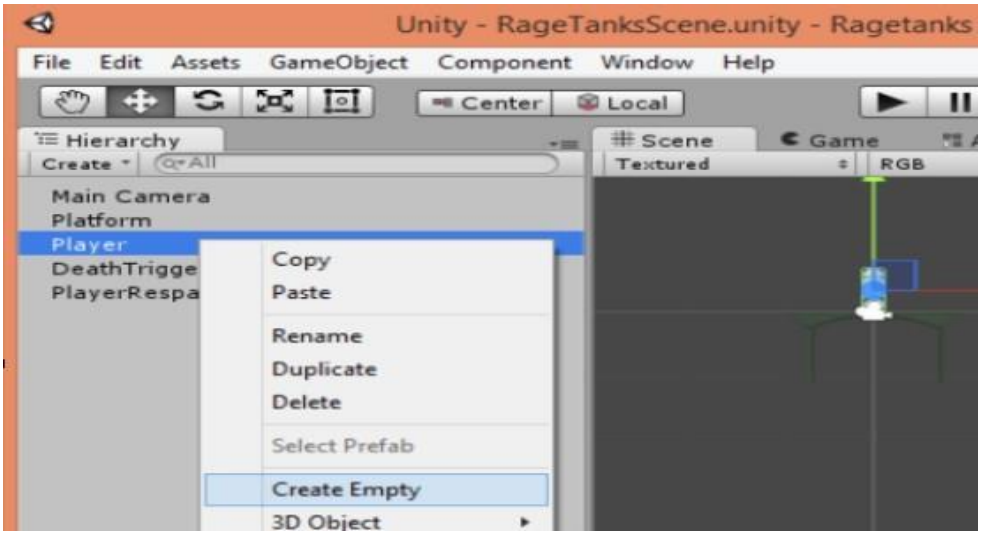
5.2.2 Detectar col·lisió del Player

Els components de Unity que permeten gestionar col·lisions són els **colliders**. Quan hem programat l'objecte **DeathTrigger** hem vist que, si activem la característica **IsTrigger** en un collider, quan hi col·lisionem, s'executa el mètode **OnTriggerEnter2D** (També existeix OnTrigger**Exit**2D, que s'executa quan es deixa de col·lisionar, i OnTrigger**Stay**2D, que s'executa a cada frame mentre duri la col·lisió).

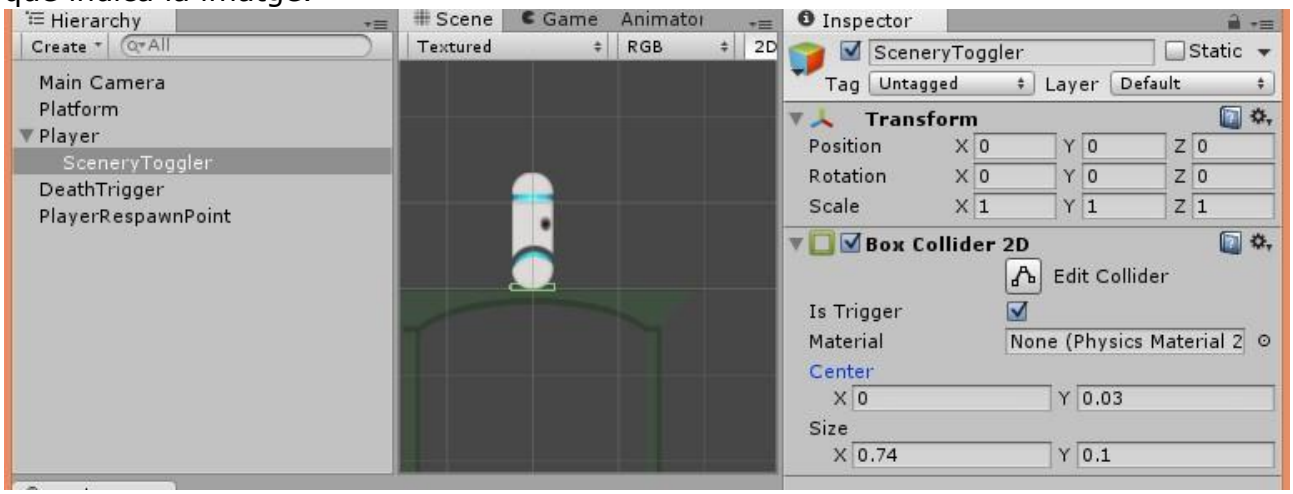
Ara procedirem de manera similar amb el Player per detectar l'aterratge sobre una plataforma. L'aterratge del Player el detectarem per mitjà d'un nou **collider** que, a més, serà **Trigger**. El tractament el farem en el mètode OnTriggerEnter2D (el component box collider 2D que ja té el Player, el deixem tal com està perquè la seva funció és que el Player sigui detectat per altres colliders).

El tractament de OnTriggerEnter2D el codificarem en un script. Com que els scripts s'associen a objectes i no a components, crearem un nou objecte buit (**fill del Player**) que tingui el nou collider 2D com a component.

Així que seleccionem el Player i, amb el botó dret, creem un objecte buit que, en la jerarquia, queda a sota del Player.



Anomenem **SceneryToggler** a aquest nou objecte i li afegim un component **BoxCollider2D** que marquem com a **IsTrigger**. Li donem els valors de Size i Center que indica la imatge.



Podem veure el collider del nou objecte com un rectangle en la base del Player.

Per tractar la col·lisió, creem l'script **PlayerColliderListener**, que haurem d'associar a l'objecte **SceneryToggler**.

El codi del nou script és:

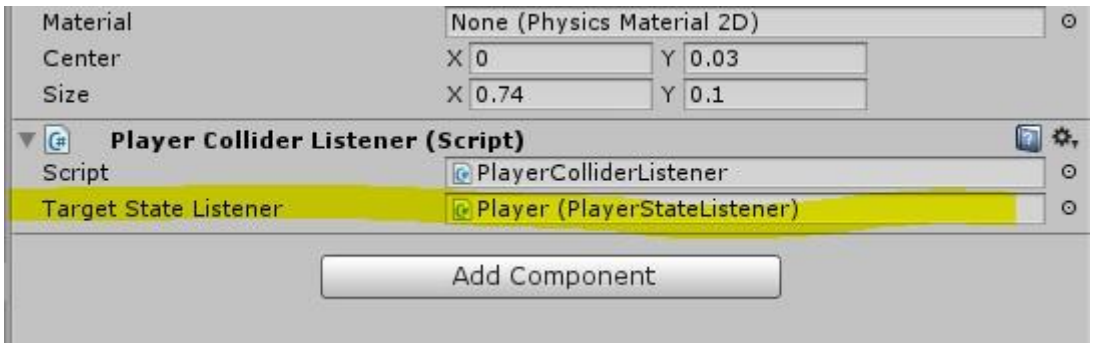
```
using UnityEngine; using
System.Collections;

public class PlayerColliderListener : MonoBehaviour
{
  public PlayerStateListener targetStateListener = null;

  void OnTriggerEnter2D( Collider2D collidedObject )
  { switch(collidedObject.tag)
      {   case "Platform":
                // Quan el Player cau en una plataforma, canviar l'estat.
            targetStateListener.onStateChange(
                        PlayerStateController.playerStates.landing);
            break;
      }
  }
}
```

Comentaris al codi:
Quan el Player cau en una plataforma (cosa que sabem pel valor del tag del collidedObject), simplement generem un event de canvi d'estat cap a l'estat **landing**.

S'ha de tenir compte d'associar l'script **PlayerColliderListener** al nou objecte **SceneryToggler**. Després, des de l'Inspector assignem l'objecte Player a la propietat TargetStateListener (la variable pública de l'script)


Material
None (Physics Material 2D)
Center
X 0
Y 0.03
Size
X 0.74
Y 0.1
Player Collider Listener (Script)
Script
PlayerColliderListener
Target State Listener
Player (PlayerStateListener)
Add Component

5.2.3 Gestió del nou estat "landing"

Hem d'afegir codi en els llocs habituals: a l'script **PlayerStateListener** afegim el "case" per l'estat **landing** a onStateCycle, onStateChange, checkForValidStatePair i checkIfAbortOnStateCondition.

OnStateCycle:

Crear el codi no farem res en el cas de landing.

OnStateChange:

En el cas de landing s'ha d'indicar a la variable playerHaslanded;

checkForValidStatePair:

Desde Landig només es pot pasar a idle, left o right.

checkIfAbortOnStateCondition:

De moment no tenim cap condició que no deixi estar en landing.

Exercicis de la 2a part

Implementa l'estat de landing indicat en aquest punt.