



HLC

2º DAW

The background of the slide features a dark blue gradient. On the left, a blue and green geometric shape is partially visible. Two futuristic cars are shown in motion: a blue car on the left leaving a thick white smoke trail, and a red car on the right leaving a large, billowing orange and yellow smoke trail. The text is centered and white, providing information about the Unity Asset Store.

Asset Store

La Asset Store es una tienda de recursos de todo tipo:

Código
Texturas
Materiales
Modelos
Animaciones
Utilidades...

De pago y gratuitos.

Unity ofrece muchos packs de ejemplo como parte de sus tutoriales y documentación:

Standard Assets
2D Game Kit
3D Game Kit
Particles Pack
UI sample

Si hemos entrado con nuestra cuenta de Unity podremos acceder a:

Favoritos
Mi Perfil

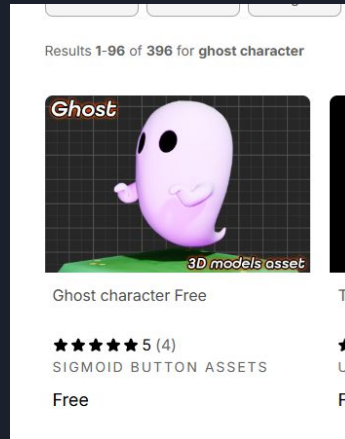
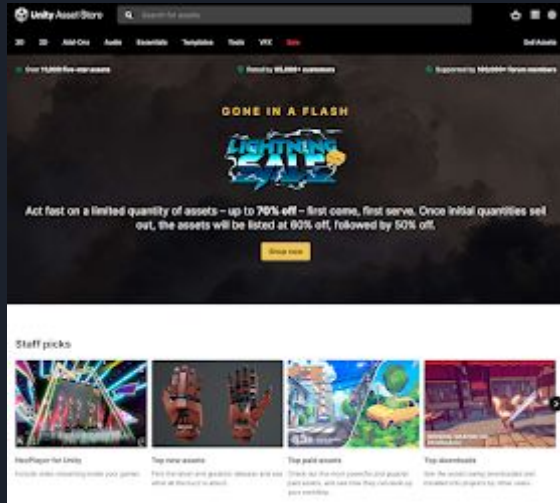
Assets ya descargados o comprados

Todas estas opciones están disponibles tanto en la web como en el propio Unity, a través de la ventana
Asset Store (Window/Asset Store)

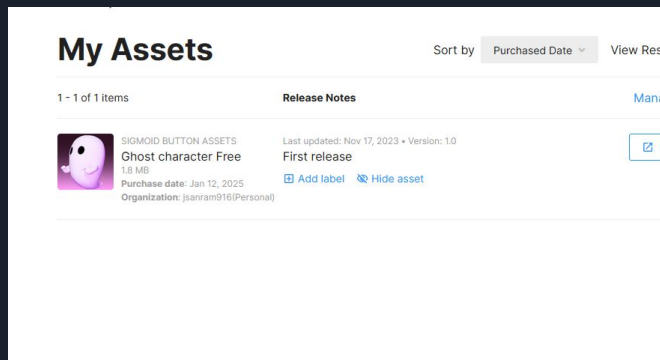
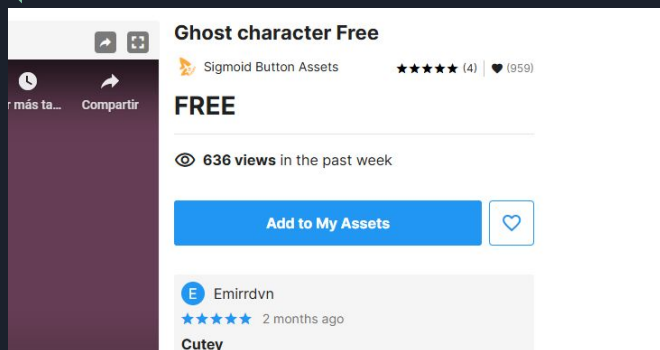
Asset Store

<https://assetstore.unity.com/>

Iremos a la web de Asset Store y buscaremos un elemento, por ejemplo: *Ghost character*



Asset Store

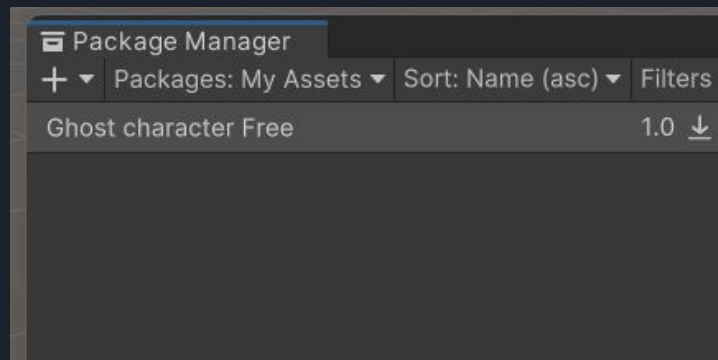


✓ Added to My Assets

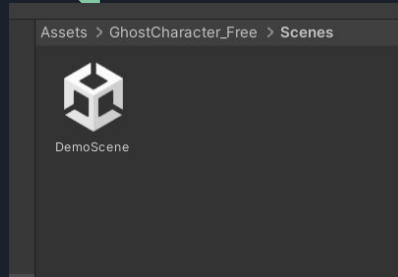


SIGMOID BUTTON ASSETS
Ghost character Free

Desde Unity, Window > Package Manager > My Assets. E importamos.



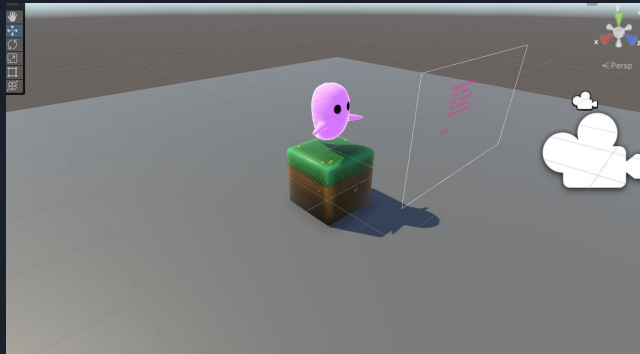
Asset Store



Abrimos la **escena** de demostración que nos trae el fantasma.

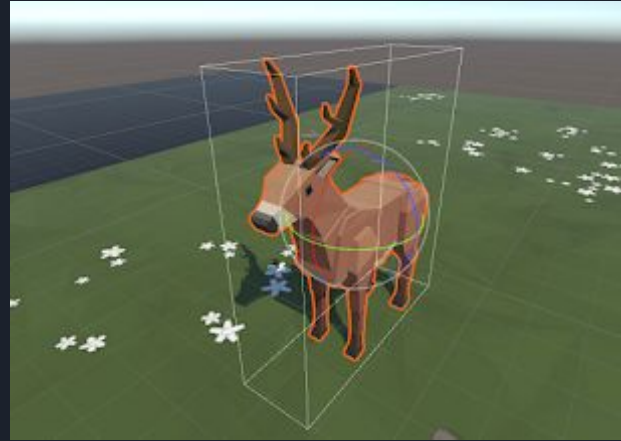
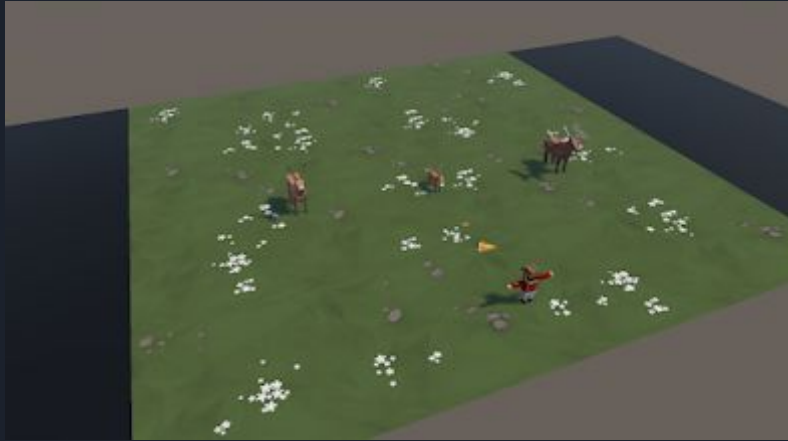
En Unity, una scene (escena) es un contenedor que organiza todos los elementos y configuraciones necesarios para una parte específica de nuestro juego o aplicación. Podemos pensar en una escena como un nivel, una pantalla o un estado del juego.

Por ejemplo, podríamos tener una escena para el menú principal, otra para el primer nivel del juego, otra para el nivel final, y así sucesivamente.



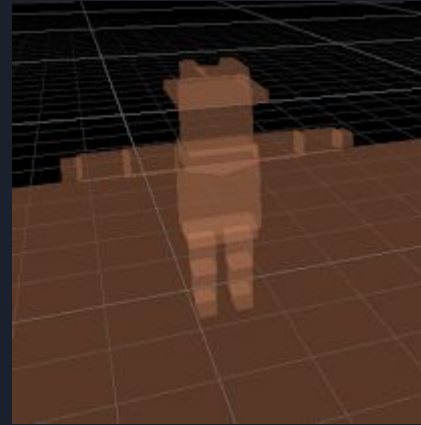
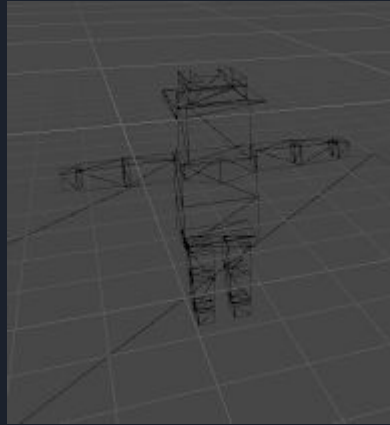
Proyecto. Creación de un juego real

Con los conocimientos básicos que tenemos en Unity y C#, vamos a crear un juego completo, con movimiento de jugadores, disparos y mapa.



Proyecto. Creación de un juego real

En primer lugar, vamos a crear un nuevo proyecto, en el que vamos a importar un conjunto de recursos que encontraremos en Moodle. **Prototype-2-Starter-File**
En este juego, un granjero disparará comida a los animales para eliminarlos.

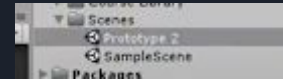


Proyecto. Creación de un juego real

Importamos el paquete descargado
Añadimos la escena Assets>Scenes>Prototype 2

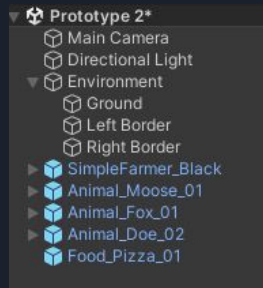


Podemos eliminar la escena de muestra Sample Scene



Si nos aparece algún elemento en rosa: Edit>Rendering>Materials>Convert

Desde Assets>Course Library > Humans, podemos añadir un granjero a nuestro plano.
Desde Assets>Course Library > Animals > Forest, por ejemplo, añadimos 3 animales.
Y también desde Assets>Course Library > Food añadiremos un trozo de pizza.





Proyecto. Creación de un juego real

Vamos a añadir movimiento lateral al personaje del granjero. Al cual vamos a renombrar desde la vista de jerarquía a **Jugador**.

Para poder usar este movimiento lateral crearemos un script en una carpeta que vamos a crear:

Assets>CourseLibrary>Scripts>ControlJugador.cs.
Este script se lo asignaremos (arrastrando por ejemplo) a nuestro Jugador.

Con esto, ya podemos mover nuestro granjero, pero se sale por los lados del mapa.

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

public class ControlJugador : MonoBehaviour
{
    public float desplazamientoHorizontal;
    public float velocidad = 10.0f;

    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {

        desplazamientoHorizontal = Input.GetAxis("Horizontal");
        //Mover al jugador horizontalmente
        transform.Translate(Vector3.right *
desplazamientoHorizontal * Time.deltaTime * velocidad);

    }
}
```