

**MP\_0489. Programación  
multimedia y dispositivos móviles**

**UF4. Análisis de motores de juegos**

**4.3. Escenas y Assets**

# Índice

---

☰	Objetivos	3
☰	Conceptos básicos	4
☰	Navegar entre escenas	6
☰	Definición y uso de assets	12
☰	Resumen	17

# Objetivos

---

Con esta unidad perseguimos los siguientes objetivos:

- 1 Comprender qué es una escena en Unity y conocer su utilidad.
- 2 Aprender a cargar y recargar una escena.
- 3 Saber navegar entre escenas.
- 4 Conocer qué son los *assets* y saber para qué se utilizan.

---

¡Ánimo y adelante!

# Conceptos básicos

---

Las escenas **contienen los objetos del juego**. Pueden ser usadas para crear un menú principal, niveles individuales y cualquier otra cosa.

Para utilizar las funciones de escenas en Unity usaremos la clase **SceneManager**.

Deberemos **incluir ese paquete en nuestros archivos C#**, como te mostramos a continuación:

```
using UnityEngine.SceneManagement ;
```

## Cargar escenas

Podemos cargar escenas de **dos maneras**:

- Utilizando su **índice**.
- O utilizando su **nombre**.

```
SceneManager.LoadScene(4)
SceneManager.LoadScene("escena4")
SceneManager.LoadScene("carpeta/escena4")
```

## Recargar una escena

No tenemos una función específica para recargar una escena; la manera de hacerlo podría ser esta:

```
SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex);
```

## Recuperar la escena actual

Unity cambió su convención de nomenclatura de "nivel" a "escenas" porque de este modo puede cargar varias escenas al mismo tiempo. Esto rompe el concepto de "nivel actual", que ha sido reemplazado por el de "escena activa".

En la mayoría de los casos, la escena activa es la última que se ha cargado. Podemos consultarla en cualquier momento usando:

```
Scene escena = SceneManager.GetActiveScene();
Debug.Log(escena.name); //devuelve el nombre de la escena actual
Debug.Log(escena.buildIndex); //devuelve el indice de la escena actual
```

# Navegar entre escenas

En el siguiente apartado te mostraremos **cómo cambiar de escenas pulsando un botón**, es decir, cómo navegar entre escenas.

## Crear dos escenas

1

Creamos un **nuevo proyecto**.

2

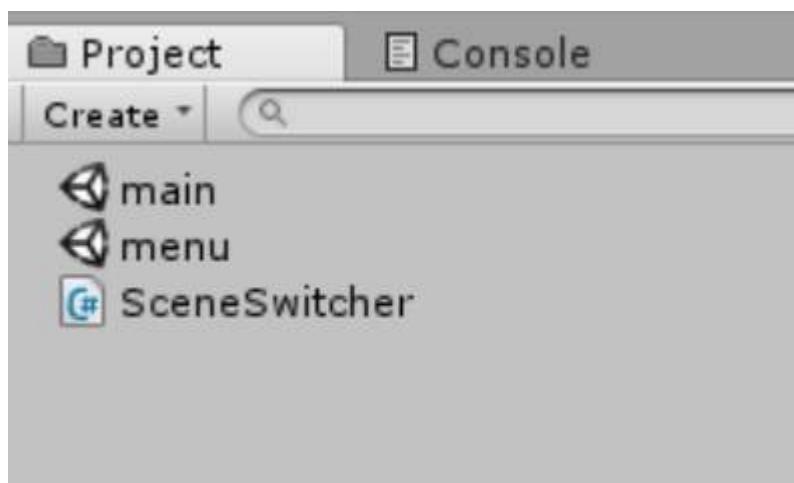
Guardamos la escena actual como "**main**".

3

Creamos **una nueva escena** en el menú de la vista *proyectos* y la nombramos "**menu**".

4

Creamos un **c# script** y lo nombramos como "**SceneSwitcher**".



## Código para abrir las escenas

1

Abrimos el archivo C# para escribir dos métodos públicos que abrirán, cada uno, una escena.

El código será el siguiente:

```
using UnityEngine;
using UnityEngine.SceneManagement;

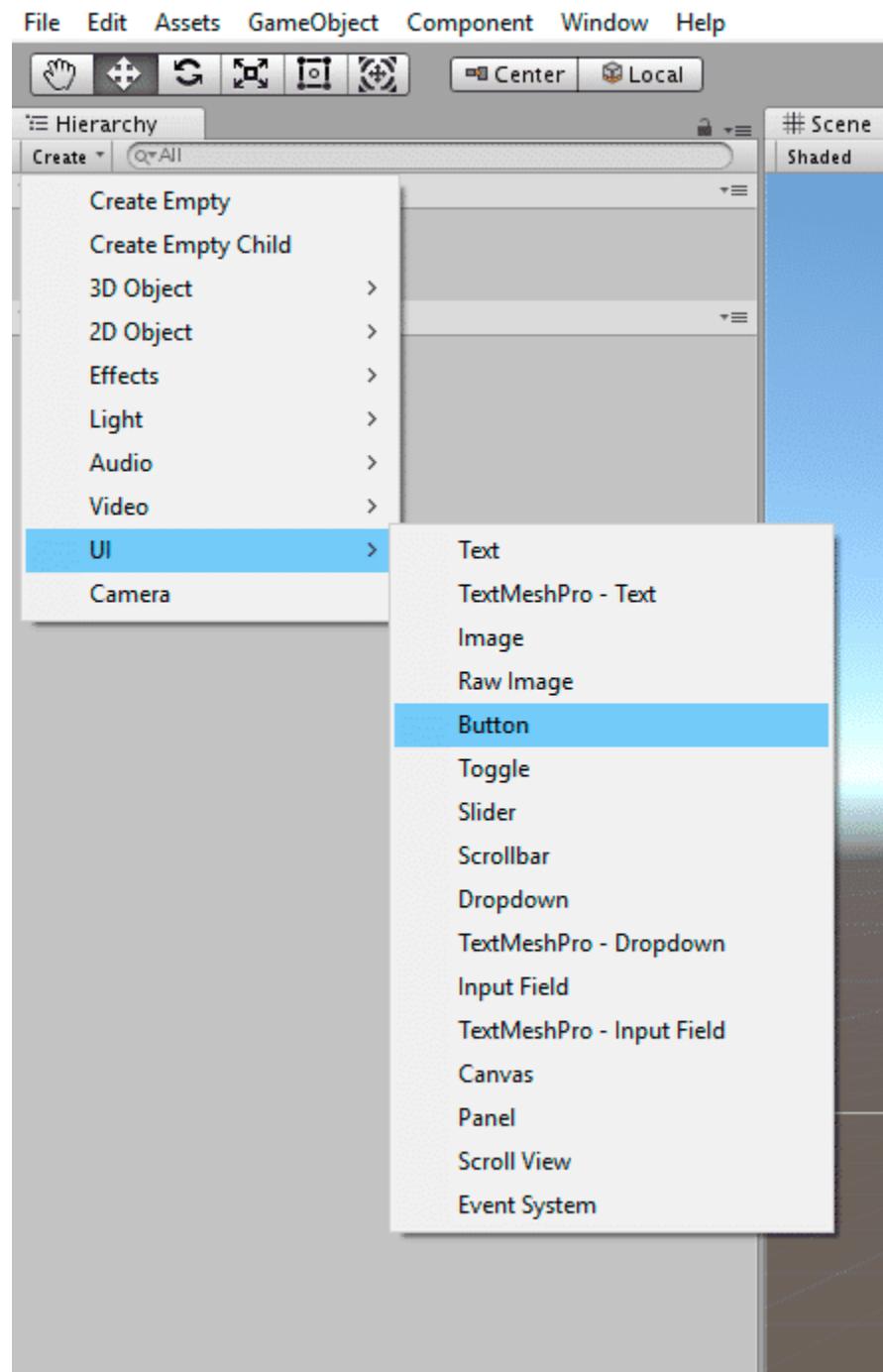
public class SceneSwitcher : MonoBehaviour
{
    public void GotoMainScene()
    {
        SceneManager.LoadScene("main");
    }

    public void GotoMenuScene()
    {
        SceneManager.LoadScene("menu");
    }
}
```

## Crear el botón

1

Abrimos la escena "menu" y, en el panel de *Jerarquía*, creamos un botón a través del menú.

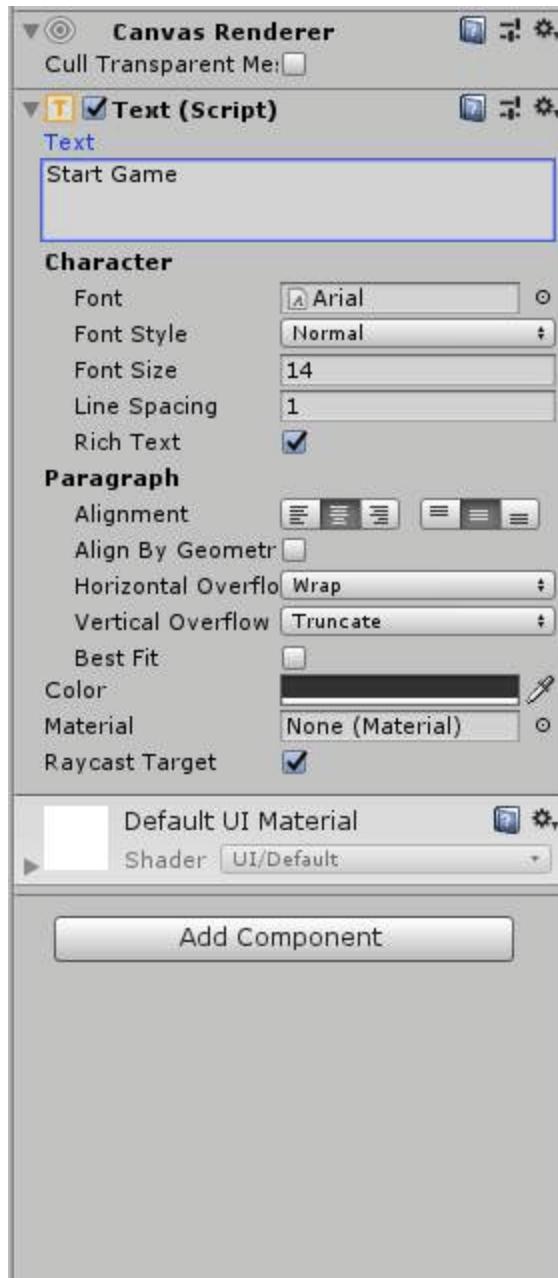


2

En el panel de *Jerarquía*, desplegamos el botón y seleccionamos *texto*.

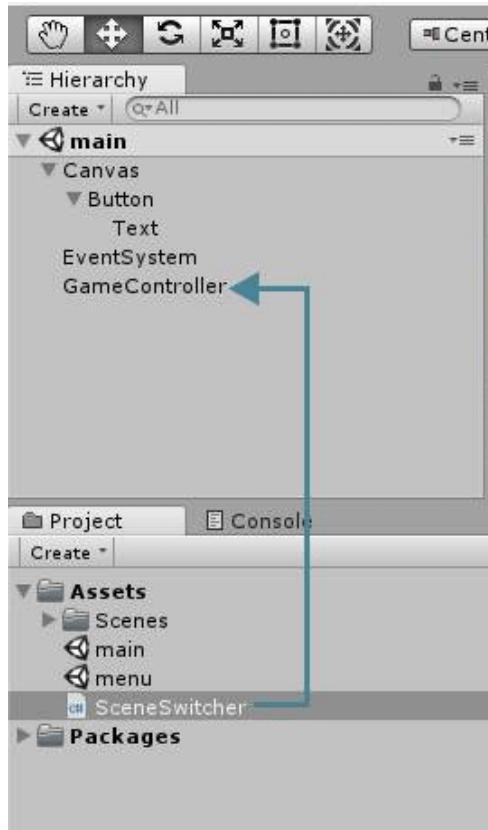
3

En el panel de *Inspector* escribimos "Start Game".



4

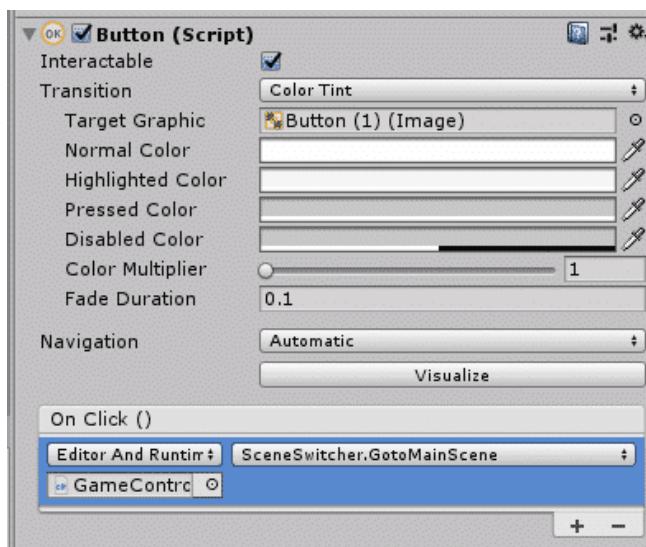
En el panel de *Jerarquía* creamos un *GameObject* vacío (*Create Empty*), lo renombramos como *GameController* y arrastramos el script *SceneSwitcher* hasta él.



5

Por último, seleccionamos el botón y, en el panel de *Inspector*, localizamos el evento *OnClick*, seleccionamos *Edit and Runtime* y después *GameController*.

En la función pondremos **SceneSwitcher > GotoMainScene**.



## Segunda escena

1

Simplemente repetiremos la misma operación en la escena "main", cambiando el texto del botón por "Menu".

2

En el evento *OnClick* la función elegida será: **SceneSwitcher > GotoMenuScene**.

3

Si lo probamos, veremos **cómo cambiamos de escena al pulsar el botón**.

# Definición y uso de assets

---

---

Un *asset* es una **representación de cualquier ítem que puede ser utilizado en un juego o proyecto.**

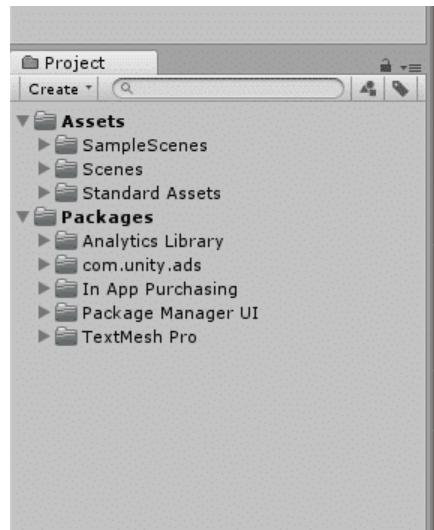
---

Un *asset* **puede venir de un archivo creado fuera de Unity**, como un modelo 3D, un archivo de audio, una imagen, o cualquiera de los tipos de archivos que Unity soporta. Y **también lo podemos crear dentro de Unity.**

## Importar Assets

 Recuerda que los *assets* que no se crean en Unity **se deben importar**.

Podemos importarlos, bien guardándolos dentro de la carpeta *assets* de nuestro proyecto, bien arrastrándolos a la carpeta *assets* que vemos en nuestro panel de proyecto.



Es importante tener en cuenta que cualquier cambio que hagamos en nuestra carpeta de *assets*, hará que se actualice el panel de proyectos. Por ejemplo, si eliminamos un *asset* desde el panel, desaparecerá en nuestra carpeta.

## Importar Standard Assets

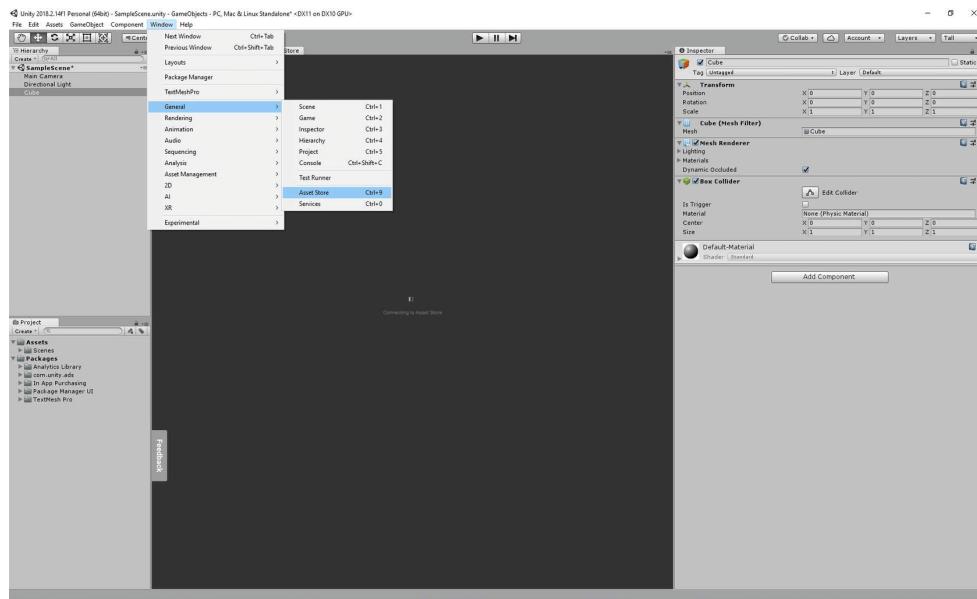
Existe un conjunto de *assets* que venían instalados por defecto en anteriores versiones de Unity y que suelen ser útiles para algunas tareas sencillas, o para realizar pruebas y ejemplos, por ejemplo.

Las últimas versiones de estos *assets* tenemos que importarlas de la *Assets Store*, como cualquier otro conjunto de *assets*.

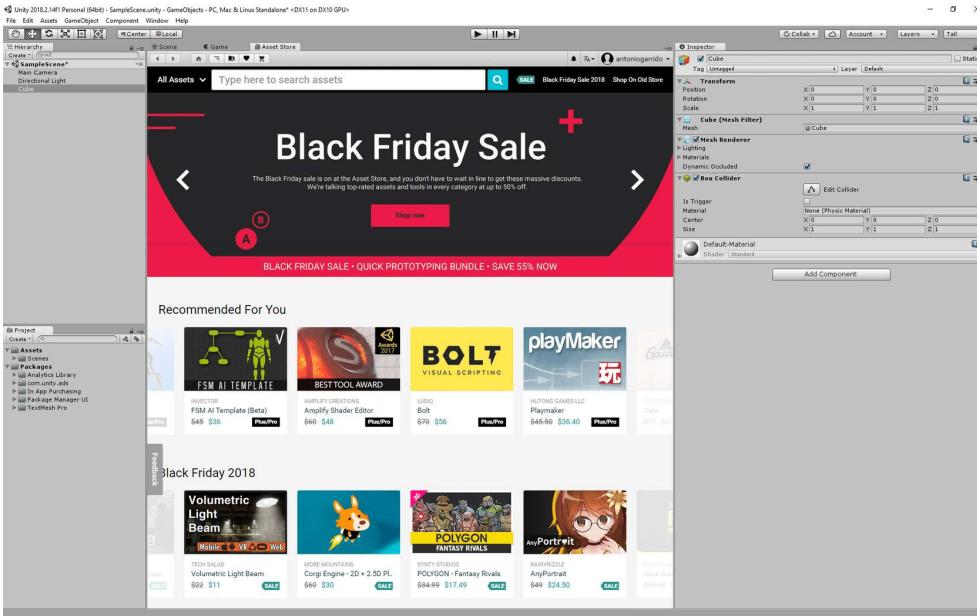
 **Assets Store** es una tienda en línea integrada en el programa (aunque ofrece también productos gratuitos, como vamos a ver) donde se pueden encontrar *assets* de todo tipo para nuestros juegos o proyectos: modelos 3D, audios, trozos de código, etc.

# Utilizar Asset Store

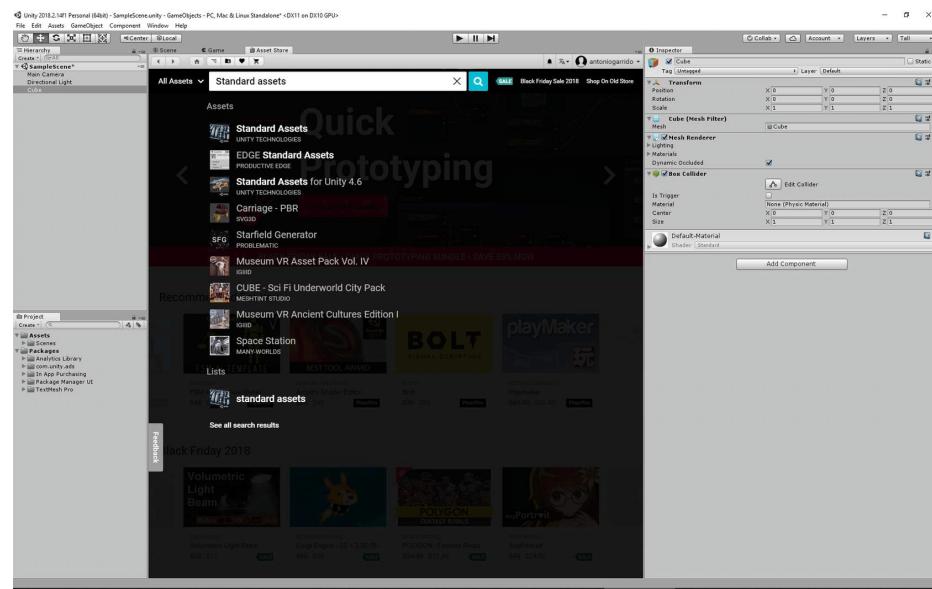
Veamos cómo utilizar la *Asset Store*, importando los *Standard Assets*.



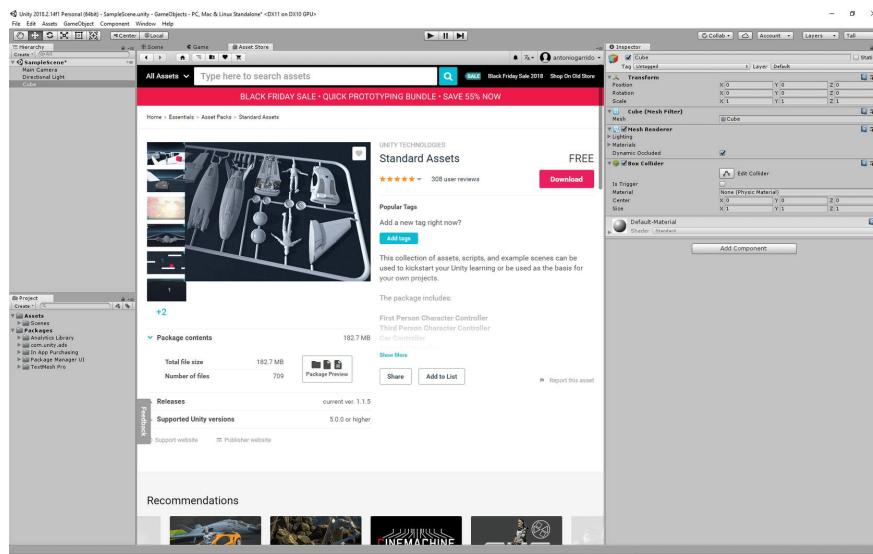
Dentro del menú *Window*, desplegamos *General* y pulsamos sobre *Asset Store*.



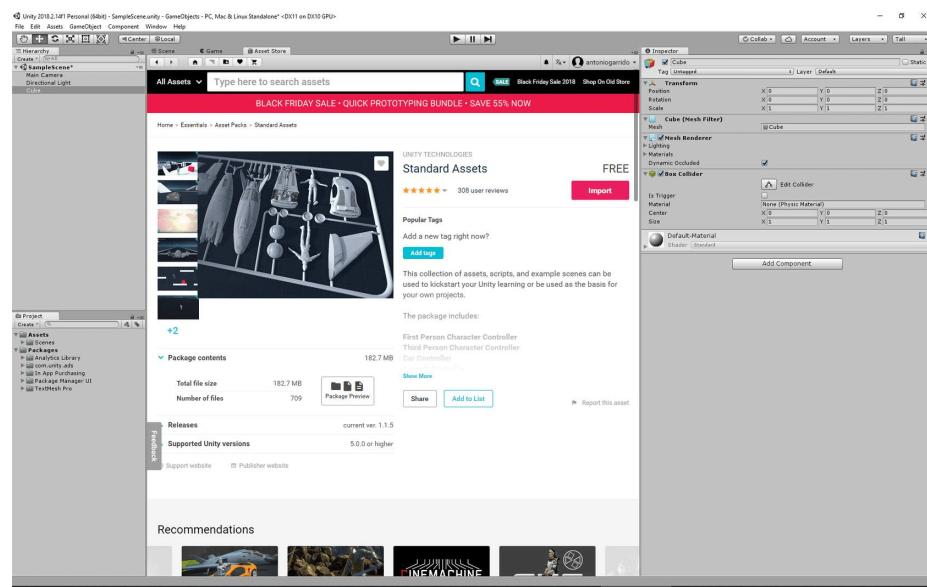
Aparecerá una nueva pestaña en el bloque central, donde cargará la *Asset Store*.



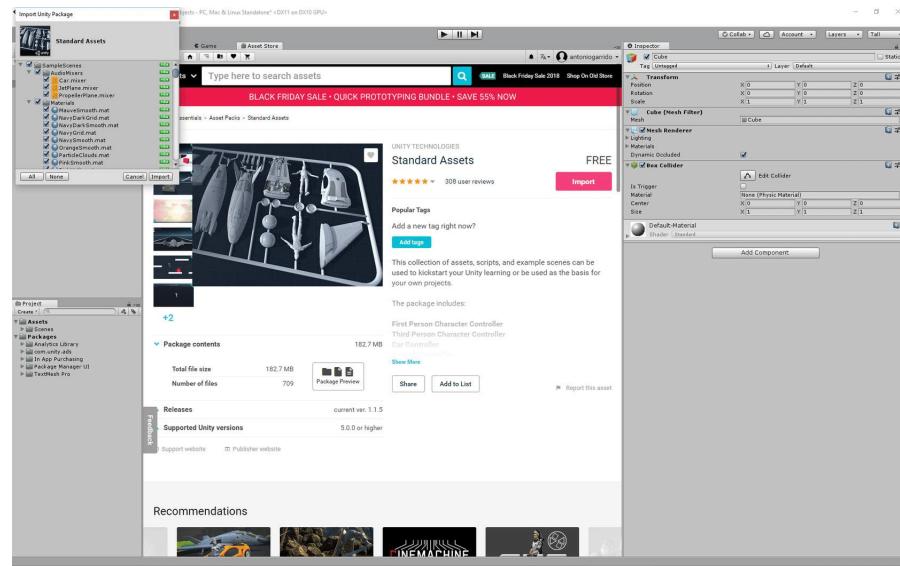
Utilizamos la herramienta **Buscar** para localizar los **Standard Assets**.



Se nos muestra la **ficha de Standard Assets**, donde podemos ver que son gratis (*Free*). Pulsamos en **Download** para descargarlos a nuestro ordenador.



Una vez descargados, el botón *Download* pasará a poner *Import*. Lo pulsaremos para importarlos a nuestro proyecto.



Aparecerá una ventana flotante en la que veremos todos los elementos y podremos elegir los que queremos importar; en este caso, **dejamos todos y pulsamos el botón Import**.

## ¡HECHO!

Ya tenemos los assets en nuestro panel de proyecto y podemos empezar a utilizarlos.

# Resumen

---

Hemos terminado la lección, repasemos los puntos más importantes que hemos tratado.

- En esta unidad hemos seguido analizando motores de juego y, para ello, hemos visto el concepto de **escena**, que es el **lugar donde insertaremos todos los objetos y elementos de nuestro juego**.
- Despues hemos aprendido a **cargar una escena**, averiguar cuál es la escena actual y a **recargar las escenas**.
- También hemos creado un ejemplo sencillo con el que poder **navegar entre escenas a través de botones**.
- Y, para finalizar la unidad, hemos descubierto qué son los **assets**, para qué sirven y cómo podemos importarlos a nuestro juego o proyecto.

