**Основы C#. HideInInspector и SerializeField.**

**Лекция 4.1.**

**Атрибуты.**

В одном из прошлых уроков мы познакомились с атрибутами доступа полей (private, public). Сейчас на примере одного скрипта нужно рассмотреть разные атрибуты инспектора для полей, которые так или иначе будет полезно использовать во время написания кода и создания собственной игры. Для начала рассмотрим простой скрипт, который называется «CharacterStats», он хранит в себе различные данные персонажа.

Атрибут **[HideInInspector]** будет полезен, когда Вам необходимо скрыть общедоступную переменную (public) из инспектора, но оставить возможность изменять её состояние из других скриптов.

Атрибут **[SerializeField]** не позволяет изменять поля из другого скрипта, т. к. это опасно и может привести к багам и непредсказуемым последствиям.

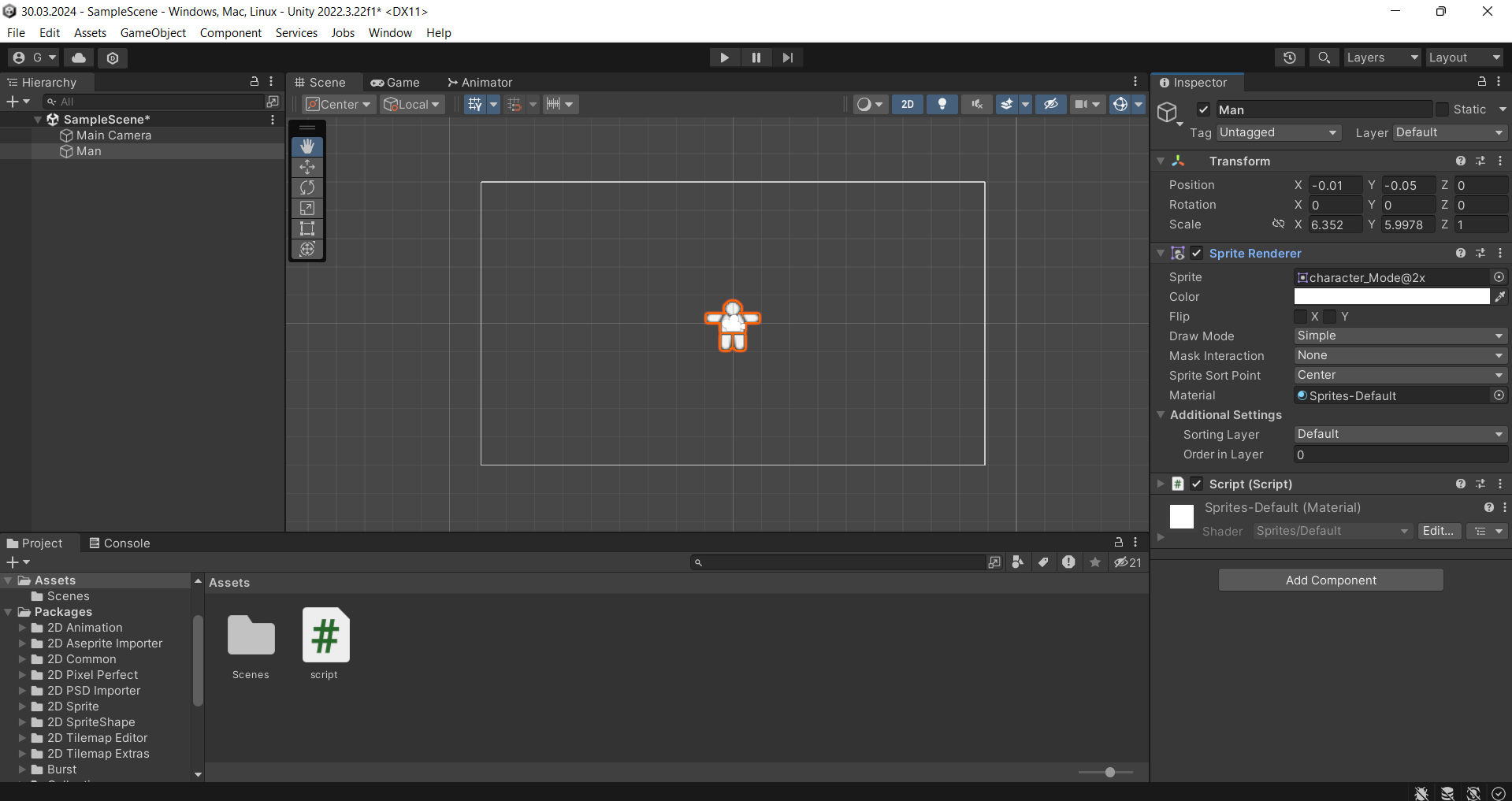
1. **Создание объекта:**  
   Создаем пустой объект “Man”, в котором мы расположим Sprite Render для отображения модели объекта и Script с его характеристиками.  
     
   

Рисунок 1 – Объект.

1. **Создание скрипта:**

Откроем наш скрипт в Visual Studio и пропишем характеристики персонажа.

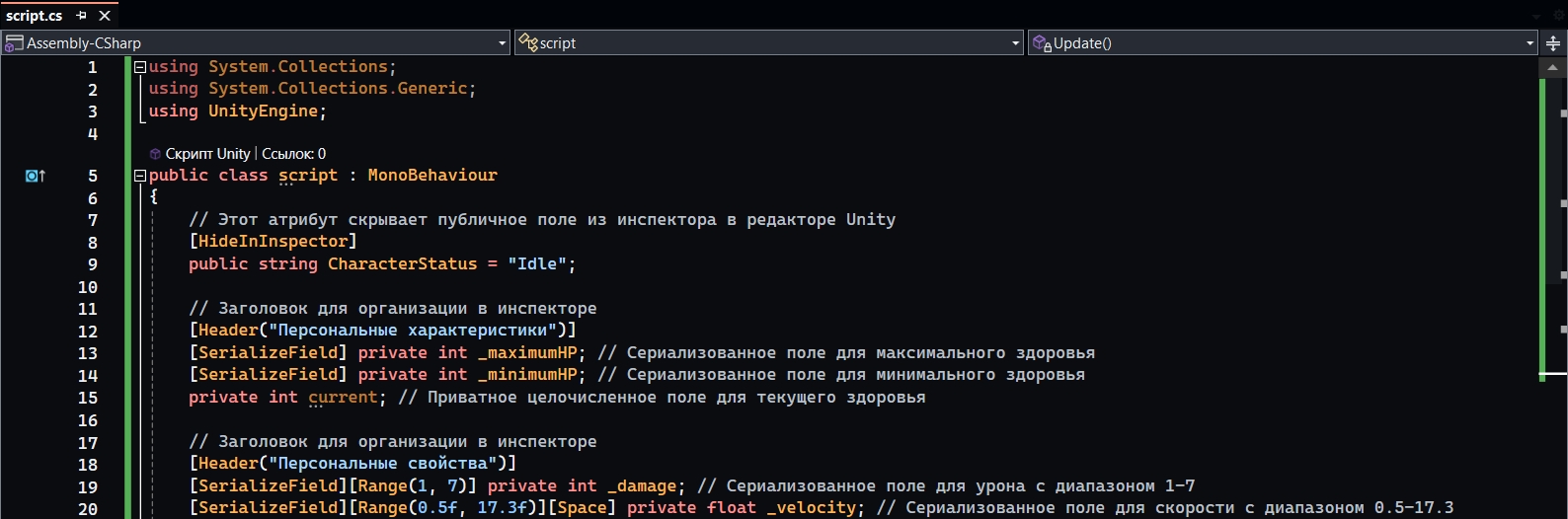


Рисунок 2 – Скрипт.

1. **Характеристики:**

После добавления к объекту нашего скрипта, у него появляются характеристики, такие как:

1. «CharacterStatus» (публичная строковая переменная, которая описывает текущее состояние персонажа).

2. «\_maximumHP» (приватная целочисленная переменная, которая хранит в себе максимальное количество жизней персонажа).

3. «\_minimumHP» (приватная целочисленная переменная, которая хранит в себе минимальное количество жизней персонажа).

4. «currentHP» (приватная целочисленная переменная, которая хранит в себе текущее количество жизней персонажа).

5. «\_damage» (приватная целочисленная переменная, которая хранит в себе урон, который может наносить персонаж).

6. «\_velocity» (приватная численная переменная с плавающей запятой, которая хранит в себе скорость персонажа).

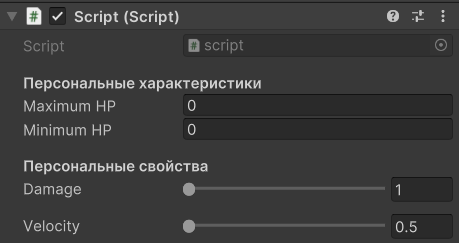


Рисунок 3 – Характеристики.

1. **Вывод:**

В итоге, использование атрибутов HideInInspector, SerializeField и Range позволяет более гибко настраивать отображение и доступ к переменным в инспекторе Unity, делая процесс разработки более удобным и безопасным.