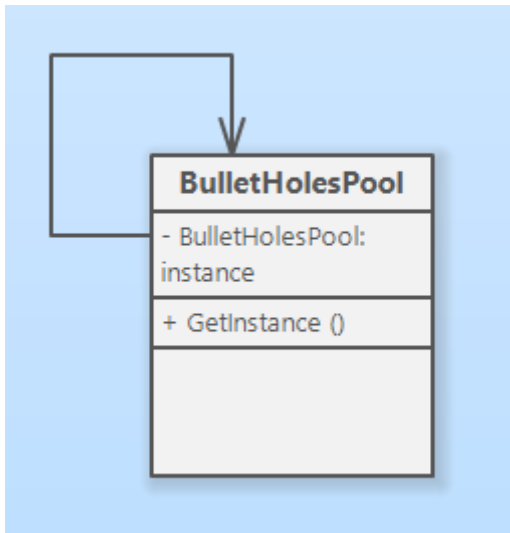


Одинак - це породжувальний патерн проєктування, який гарантує, що у класу є тільки один екземпляр, і надає до нього глобальну точку доступу.

В моєму проєкті є декілька прикладів цього патерну. Наприклад клас **BulletHolesPool**, який містить в собі *басейни дірок від пуль. Мені в сцені потрібне лише 1 такий об'єкт, тому тут я використовую метод **Одинак**.



В мене є статичне поле **instance**, яке відповідає за “одинокість” компонента. Також є метод **GetInstance**, який повертає **instance**.

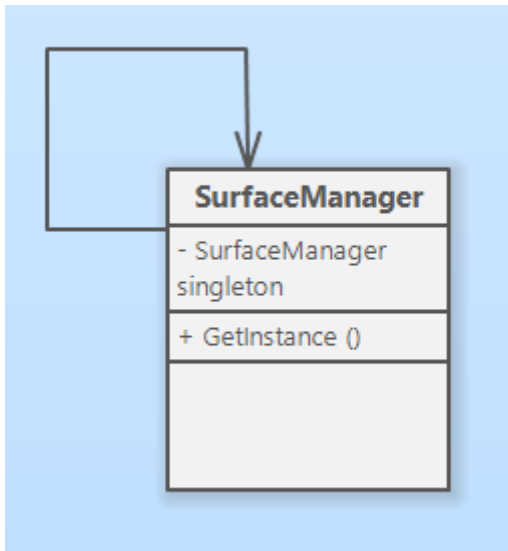
```
static BulletHolesPool instance;
0 references
static public BulletHolesPool GetInstance()
{
    return instance;
}

Unity Message | 0 references
void Start()
{
    if (instance && instance != this)
        Destroy(this);
    instance = this;
}
```

В методі **Start** (коли гра запущена) в мене йде перевірка на “одинокість” компонента, якщо компоненти не є головним компонентом (`instance != this`), то цей компонент видаляється.

Також другим прикладом в мене може слугувати клас **SurfaceManager**, який відповідає за виявлення типу поверхні під гравцем. Клас не повністю мій, але досить сильно модифікований.

Працює він аналогічно як і клас вище.



```
Unity Message | 0 references
void Start() {
    if(!singleton) singleton = this;
    else if(singleton != this) Destroy(gameObject);
}
```

***Басейни** - це породжувальний патерн, який попередньо створює всі об'єкти, що знадобляться вам у певний момент, ще до початку гри. Це усуває необхідність створювати нові об'єкти або знищувати старі під час гри.

Класи можете знайти за шляхом:

Assets/FreeAssets/FREE Footsteps System/scripts/SurfaceManager.cs

Assets/Scripts/Weapon/ShootingWeapon/PoolableObjects/BulletHolesPool.cs