

# Unity GameObject pool

Simple pool with a unity like signature of methods.

## Use:

### 1. Add:

```
using GameObjectPooling;
```

### 2. Insert the `GameObjectPool` . before calling the `Instantiate` and `Destroy` methods:

```
GameObjectPool.Instantiate(prefab, position, Quaternion.identity);  
Rigidbody instance = GameObjectPool.Instantiate<Rigidbody>(prefab, parent);  
GameObjectPool.Destroy(anyGameObject);
```

### 3. If necessary, reset variables in components after instance reuse:

```
Rigidbody instance = GameObjectPool.Instantiate(prefab, position, Quaternion.identity);  
instance.velocity = Vector3.zero;
```

## Methods:

```
public static T Instantiate<T>(T original) where T : Component;

public static GameObject Instantiate(GameObject original, Vector3 position,
    Quaternion rotation);

public static GameObject Instantiate(GameObject original, Vector3 position,
    Quaternion rotation, Transform parent);

public static GameObject Instantiate(GameObject original);

public static GameObject Instantiate(GameObject original, Transform parent);

public static T Instantiate<T>(T original, Transform parent,
    bool worldPositionStays) where T : Component;

public static T Instantiate<T>(T original, Transform parent) where T : Component;

public static T Instantiate<T>(T original, Vector3 position,
    Quaternion rotation, Transform parent) where T : Component;

public static T Instantiate<T>(T original, Vector3 position,
    Quaternion rotation) where T : Component;

public static GameObject Instantiate(GameObject original, Transform parent,
    bool instantiateInWorldSpace);

public static void Destroy(GameObject obj);

public static void Destroy(GameObject obj, [DefaultValue("0.0F")] float t);
```

---

Contacts: <https://t.me/zzwzp> (<https://t.me/zzwzp>).