Read Me

Thank you for downloading Lean Pool!

If you have any questions, feel free to e-mail me at: carlos.wilkes@gmail.com

Step 1 - Replace your Instantiate calls

Find your code that heavily relies on:

```
Instantiate( … );
```

And replace it with:

```
Lean.LeanPool.Spawn( ... );
```

For example: `var enemy = Lean.LeanPool.Spawn(enemyPrefab);`

Step 2 - Replace your Destroy calls

Find your code that heavily relies on:

```
Destroy( … );
```

And replace it with:

```
Lean.LeanPool.Despawn( ... );
```

For example: `Lean.LeanPool.Despawn(enemy);`

Step 3 - Done!

Wasn't that easy?

How do I alter the settings and stuff?

Right click in your Hierarchy window, and select Lean / Pool. You should now see a Lean Pool GameObject in your scene that's automatically selected.

Next, drag and drop your prefab into the Prefab slot, and you can play around with the Preload, Capacity, etc.

Help, my Rigidbody velocity isn't being reset!!

By default Lean Pool doesn't modify any components, but for rigidbodies you can just add the LeanPooledRigidbody component to your prefab that has a Rigidbody, and its velocities will automatically get reset.

How do I create custom pooling behaviour for the _____ component?

Take a look at the LeanPooledRigidbody script, you can see that it defines the OnSpawn and OnDespawn message receivers. These automatically get called when the prefab clone is spawned and despawned.

How do I increase the performance even more?

You may notice that Lean.LeanPool.Spawn and Lean.LeanPool.Despawn have a little bit of extra overhead. This is because they're static methods, and they search all your active pools to find the right one.

If this is an issue for you then you need to store a reference to the Lean.LeanPool itself, and then you can call: yourPoolReference.FastSpawn and .FastDespawn. These are the direct pooling methods a more efficient than the static ones, but they require more programming skill to use.

Want More Assets?

If you like this asset then please check out my website here: https://sites.google.com/site/carloswilkes/