- The object Pool starts with a serialized field that holds the data for spawning the ducks ie. tag, prefab and size of the queue.

```CSharp
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

[System.Serializable]
2 references
public class Pool
{
    public string tag;
    public int size;
    public GameObject prefab;
}
```

- Within an object pooling script we make a queue and a dictionary for searching and holding the ducks we need to enable

```
public List<Pool> DuckPool;

Queue<GameObject> DuckObjPool;

public Dictionary<string, Queue<GameObject>> poolDictionary;
```

- Within the start function we initialize our queue of ducks and loop through it disabling them all since they are enabled when we spawn them.

```
void Start()
{
    poolDictionary = new Dictionary<string, Queue<GameObject>>();

    foreach(Pool pool in DuckPool)
    {
        DuckObjPool = new Queue<GameObject>();

        for(int i = 0; i < pool.size; i++)
        {
                GameObject duckObj = Instantiate(pool.prefab);
                duckObj.SetActive(false);
                DuckObjPool.Enqueue(duckObj);
        }
        poolDictionary.Add(pool.tag, DuckObjPool);
    }
}
```

- We then create a function for getting the tag of the duck in the queue that we would like to spawn then enable that specific duck and set its position and rotation we get from parameters
- Within Duck mover whenever we press space we will call the spawn duck function which gets us what duck we need to spawn and we input the position rotation and tag that it needs to find the duck and set it up.
- This enables our ducks that creates a rigid body for them so they can move
- The ducks are disabled when they are hit with the projectile the player fires.
- Instead of being destroyed they are instead deactivated
- This is more optimized since the constant instantiation and destruction of the ducks is more costly than simply recycling them by enabling and disabling them in a queue.
- This benefits the game since it will allow for better optimization of duck objects and can allow for many ducks being enabled and disabled on screen at once.
-