Lesson Summary

Key Points

- Just like classes and variables, functions can be **private** or **public.**
- The return type decides what type of data the function will give back when it is called.
- A function that doesn't return any data has a return type of **void.**
- **Parameters** refer to the information you pass into a function, typically the data that you want the function to use.
- It's possible to create different version of the same function by using different parameter patterns, known as **overloading**.
- The **out** and **ref** keywords can be used to give data back through parameters, inside of passing it in.
- To call a function, type the name of the function and pass in its parameters using parentheses (or leave them empty if the function doesn't accept any parameters.
- Use **return**, to stop a function early,

Damageable Object

```
using UnityEngine;

public class Damageable : MonoBehaviour
{
    [SerializeField] float hp = 100;

    public void TakeDamage()
    {
        float defaultDamage = 50;

        TakeDamage(defaultDamage);
    }

    public void TakeDamage(float damageAmount)
    {
        hp -= damageAmount;

        if (hp < 0)
        {
            // Object is destroyed
            Destroy(gameObject);
        }
    }
}
```

Enemy Object

```
using UnityEngine;

public class Enemy : MonoBehaviour
{
    private void OnCollisionEnter(Collision collision)
    {
        GameObject hitObject = collision.collider.gameObject;

        if (hitObject.TryGetComponent(out Damageable damageable))
        {
            damageable.TakeDamage();
        }
    }
}
```