Unity Data Structures

Data Structures

- Array
  - What we've used so far
  - Fixed size, fast
  - Unity video tutorial
- List (aka Generic List)
  - Arrays that you can add to and remove from *dynamically*
  - Unity video tutorial
- Dictionary (aka Generic Dictionary)
  - Store information using a "key" (an associative array)
  - Dynamic
  - Unity video tutorial
- Wiki: Choosing the Right Data Type
- Blog: Arrays, Hashtables and Dictionaries Explained

Creating a Dictionary

```
using System.Collections;
using System.Collections.Generic; // <- Dictionaries are in this namespace
using UnityEngine;

public class DictionaryDemo : MonoBehaviour {

    public GameObject Wolf;
    public GameObject Red;

    private Dictionary<string, GameObject> Characters;

    void Start () {
        Characters = new Dictionary<string, GameObject>();
    }

    void Update () {

    }
}
```

```
private Dictionary<string, GameObject> Characters;

void Start () {
    Characters = new Dictionary<string, GameObject>();

    // Adding elements to a dictionary
    GameObject wolf = (GameObject) Instantiate(Wolf);
    Characters.Add("Big Bad Wolf", wolf);
    GameObject tinyWolf = (GameObject) Instantiate(Wolf);
    Characters.Add("Big Bad Wolf's Brother", tinyWolf);
    GameObject red = (GameObject) Instantiate(Red);
    Characters.Add("Red Riding Hood", red);

    // Counting the elements in a dictionary
    Debug.Log(Characters.Count);

    // Removing an element from a dictionary
    Characters.Remove("Big Bad Wolf's Brother");
}
```

```
private Dictionary<string, GameObject> Characters;

void Start () {
    Characters = new Dictionary<string, GameObject>();

    // Adding elements to a dictionary
    GameObject wolf = (GameObject) Instantiate(Wolf);
    Characters.Add("Big Bad Wolf", wolf);
    GameObject tinyWolf = (GameObject) Instantiate(Wolf);
    Characters.Add("Big Bad Wolf's Brother", tinyWolf);
    GameObject red = (GameObject) Instantiate(Red);
    Characters.Add("Red Riding Hood", red);
}

void Update () {
    // Retreive a single element
    GameObject red = Characters["Red Riding Hood"];

    // Loop over all elements
    foreach(KeyValuePair<string, GameObject> element in Characters) {
        Debug.Log("The key is: " + element.Key);
        Debug.Log("The value is: " + element.Value);
    }
}
```