



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA
CAMPUS D'ALCOI

Communication between objects

Jordi Linares



Communication between objects

- ❖ There are many ways we can use to communicate between objects:
 - ❖ In a explicit way, with a public attribute of the type **GameObject** (or **Transform**) in the script, over which we will drag and drop an object of the scene
 - ❖ Using **transform.Find("GameObject name")**, that will look for the object in its parent/child hierarchy
 - ❖ Accessing to all its children: **foreach (Transform child in transform) { ... }**
 - ❖ Using **GameObject.Find("GameObject name")**, that will look for objects with this name. We can look also with tags: **GameObject.FindWithTag("tag")**.
 - ❖ Once we have the reference of the object, we can get access to its components (scripts included) using **GetComponent**
 - ❖ When looking for objects, try to do it in **Start()** or **Awake()**, getting the reference of the object and storing it in an attribute we will use afterwards in **Update/ FixedUpdate** etc.
 - ❖ We can also communicate among objects with **SendMessage, BroadcastMessage**

Communication between objects

- With **SendMessage** we can execute a method of any script that belongs to a particular GameObject
- With **BroadcastMessage** we can execute a method of any script that belongs to a particular GameObject and also with all the children the GameObject can have

```
using UnityEngine;
using System.Collections;

public class SendingMessage : MonoBehaviour {

    public GameObject receiver; // we drag and drop a gameObject in the editor

    // Update is called once per frame
    void Update () {

        //...
        // If the receiver has a script with a method called 'Die()', this method
        // will be executed
        receiver.SendMessage("Die");
        //...

    }
}
```



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA
CAMPUS D'ALCOI

Communication between objects

Jordi Linares

