

Unity Εργαστήριο 03

Μετακίνηση προς επιλεγμένο στόχο

- Ξεκινάμε με τη **σκηνή** και το **script** από το Lab 02
- http://www.image.ece.ntua.gr/courses_static/cg/

- Δημιουργούμε **δύο κύβους**
- **Tags:** tower
- **Name:** bottomleft, bottomright
- Η ιδέα είναι ότι τα καινούρια αντικείμενα πρέπει να κατευθύνονται στους κύβους

Μετακίνηση προς ένα στόχο

```
function MoveObject (thisTransform : Transform, startPos : Vector3, endPos :  
    Vector3, time : float) {  
    var i = 0.0;  
    var rate = 1.0/time;  
    while (i < 1.0) {  
        i += Time.deltaTime * rate;  
        thisTransform.position = Vector3.Lerp(startPos, endPos, i);  
        yield;  
    }  
}
```

```
if(Input.GetMouseButtonDown(0))
    {
        ray = Camera.main.ScreenPointToRay(Input.mousePosition);
        if (Physics.Raycast(ray, hit, 100)) {
            if (hit.collider.name=="Plane") newHero = Instantiate(hero1, hit.point +
Vector3(0, 1, 0), Quaternion.identity);
                }

            newHero.name = "Hero"+nextNameNumber;
            objectToMove = GameObject.Find("Hero"+nextNameNumber);
            Debug.Log ("New object name: "+ objectToMove.name);
            target = GameObject.Find("bottomleft");
            nextNameNumber++;

            hero1.transform.position = Vector3.MoveTowards(transform.position,
target.gameObject.transform.position, 10);
            //MoveObject (objectToMove.transform, objectToMove.transform.position,
target.gameObject.transform.position, 1);

        }
    }
```

```
function FindClosestEnemy (whichObject : GameObject) : GameObject {
    // Find all game objects with tag Tower
    var gos : GameObject[];
    gos = GameObject.FindGameObjectsWithTag("tower");
    var closest : GameObject;
    var distance = Mathf.Infinity;
    var position = transform.position;
    // Iterate through them and find the closest one
    for (var go : GameObject in gos) {
        var diff = (go.transform.position - whichObject.transform.position);
        var curDistance = diff.sqrMagnitude;
        if (curDistance < distance) {
            closest = go;
            distance = curDistance;
        }
    }
    return closest;
}
```

```

if(Input.GetMouseButtonDown(0))
    {
        ray = Camera.main.ScreenPointToRay(Input.mousePosition);
        if (Physics.Raycast(ray, hit, 100)) {
            if (hit.collider.name=="Plane") newHero = Instantiate(hero1, hit.point + Vector3(0, 1, 0),
Quaternion.identity);
                }

            newHero.name = "Hero"+nextNameNumber;
            objectToMove = GameObject.Find("Hero"+nextNameNumber);
            //target = GameObject.Find("bottomleft");
            enemy = FindClosestEnemy(objectToMove);
            Debug.Log (enemy.name);
            Debug.Log ("object " + objectToMove.name + " will move to " + enemy.name);
            nextNameNumber++;

            //hero1.transform.position = Vector3.MoveTowards(transform.position,
target.gameObject.transform.position, 10);
            MoveObject (objectToMove.transform, objectToMove.transform.position,
enemy.transform.position, 1);

        }
    }

```

Final declarations

```
var hero1 : GameObject;  
var ray: Ray;  
var hit: RaycastHit;  
var gridSpacing = 1.0;  
var target: GameObject;  
var newHero : GameObject;  
var objectToMove : GameObject;  
var nextNameNumber = 0;  
var enemy : GameObject;
```