

---

# PR N°3 MERCREDI 25 OCTOBRE 2023

- Retour sur le PR n°2, jusqu'à scène 3, étape 10
- Objectif du PR n°3
- Contenu du PR n°3
- Rendu du PR n°3
- Survol de Unity 3D, partie 3

---

# PR N°3 MERCREDI 25 OCTOBRE 2023

## Retour sur le PR n°2, jusqu'à scène 3, étape 10

- Transformation du 1<sup>er</sup> script ClavierScriptTP1.cs en 3 scripts ClavierScriptTP2scene1.cs, MaterielScript.cs et SelectionScriptTP2.cs compatibles avec la séparation entre les couches Matérielle et I3D

### ClavierScript

```
public MaterielScript mat

Start()
{ Mise à 0 de mat.increment_IR }

Update()
{
    Vector3 new_increment
    Si on appuie sur 1 des 6 touches alors
        new_increment est affecté
    Sinon new_increment vaut (0,0,0)
    mat.increment_IR = new_increment
}
```

### MaterielScript

```
public Vector3 increment_IR

Start()
{}

Update()
{}
```

### SelectionScript

```
public MaterielScript mat
public Gere_bras bras_robot
public increment_VR

Start()
{}

Update()
{
    increment_VR = mat.increment_IR // MVS
    bras_robot.increment = increment_VR
}
```

---

# PR N°3 MERCREDI 25 OCTOBRE 2023

## Retour sur le PR n°2, jusqu'à scène 3, étape 10

- Ajout de la gestion de la souris et transformation de MaterielScript

### SourisScript

```
public MaterielScript mat

Start()
{ Mise à false de mat.B1,mat.B2 et mat.B3 }

Update()
{
    Si clique souris, on met à true mat.B1 ou mat.B2 ou mat.B3
    Sinon, on met à false mat.B1, mat.B2 et mat.B3
}
```

### MaterielScript

```
public Vector3 increment_IR

public bool B1, B2, B3

Start()
{ }

Update()
{ }
```

---

# PR N°3 MERCREDI 25 OCTOBRE 2023

Retour sur le PR n°2, jusqu'à scène 3, étape 10

- Ajout de NavigationScript

NavigationScript

```
public MatérielScript mat
public Vector3 increment_trans_VR
public Vector3 increment_rot_VR
public float MAGNITUDE_TRANS
public float MAGNITUDE_ROT
public BougeBaseMobile gestion_base_mobile
Start()
{}
Update()
{
    increment_rot_VR.x = mat.increment_IR.x * MAGNITUDE_ROT
    increment_trans_VR.y = mat.increment_IR.y * MAGNITUDE_TRANS
    gestion_base_mobile.increment_rotation = increment_rot_VR
    gestion_base_mobile.increment_translation = increment_trans_VR
}
```

---

# PR N°3 MERCREDI 25 OCTOBRE 2023

## Retour sur le PR n°2, jusqu'à scène 3, étape 10

- La classe ToColor

```
public class ToColor : MonoBehaviour {
private GameObject cube;
private Color old_color;
private Color cur_color;
// Use this for initialization
void Start () {
SetColor (Color.gray);
}
// Update is called once per frame
void Update () {
}
```

```
//-----
public void SetColor( Color color)
{
old_color = cur_color;
cur_color = color;
cube = transform.Find("cube").gameObject;
cube.GetComponent<Renderer>().material.color
= color;
}
//-----
void RetrieveColor( )
{
SetColor (old_color);
}
}
```

---

# PR N°3 MERCREDI 25 OCTOBRE 2023

## Retour sur le PR n°2, jusqu'à scène 3, étape 10

- La classe SelectionnableDistScript

```
public class SelectionnableDistScriptTP2scene3 :  
MonoBehaviour {  
    public bool EstSelectionnable=false;  
    public float threshold = 0.1F;  
    public ToColor gestionCouleurs;  
    private GameObject Tcube; // Sommet du cube  
    private Vector3 Tcube_init_position;  
    private GameObject Otool;  
    // Use this for initialization  
    void Start () {  
        Tcube = transform.Find("Tcube").gameObject;  
        Tcube_init_position = Tcube.transform.position;  
        Otool = GameObject.Find("Otool");  
    }  
}
```

---

# PR N°3 MERCREDI 25 OCTOBRE 2023

## Retour sur le PR n°2, jusqu'à scène 3, étape 10

- La classe SelectionnableDistScript

```
// Update is called once per frame
```

```
void Update () {  
    Vector3 Otool_position = Otool.transform.position;  
    Vector3 Tcube_position = Tcube.transform.position;  
    float norm;  
    norm = (Otool_position - Tcube_position).sqrMagnitude;  
    EstSelectionnable = (norm < threshold*threshold);  
    if (EstSelectionnable)  
        gestionCouleurs.SetColor(Color.red);  
    else  
        gestionCouleurs.SetColor(Color.grey);  
}  
}
```

---

# PR N°3 MERCREDI 25 OCTOBRE 2023

## Objectif du PR n°3

### ❑ **Achever la version 2 du projet**

- Séparation en 3 couches Matériel, I3D et Application
- Écriture des scripts gérant le matériel, gérant les techniques d'I3D, gérant l'Application
- Traçage des données
- Affichage de l'état du système sur l'écran

**Restriction 1** levée: Dans le cas de 10 objets

**Restrictions 2 et 3** levées: physicalisation des objets <-> Sélectionnable par utilisation du moteur physique

**Restriction 4** levée: Trois caméras



---

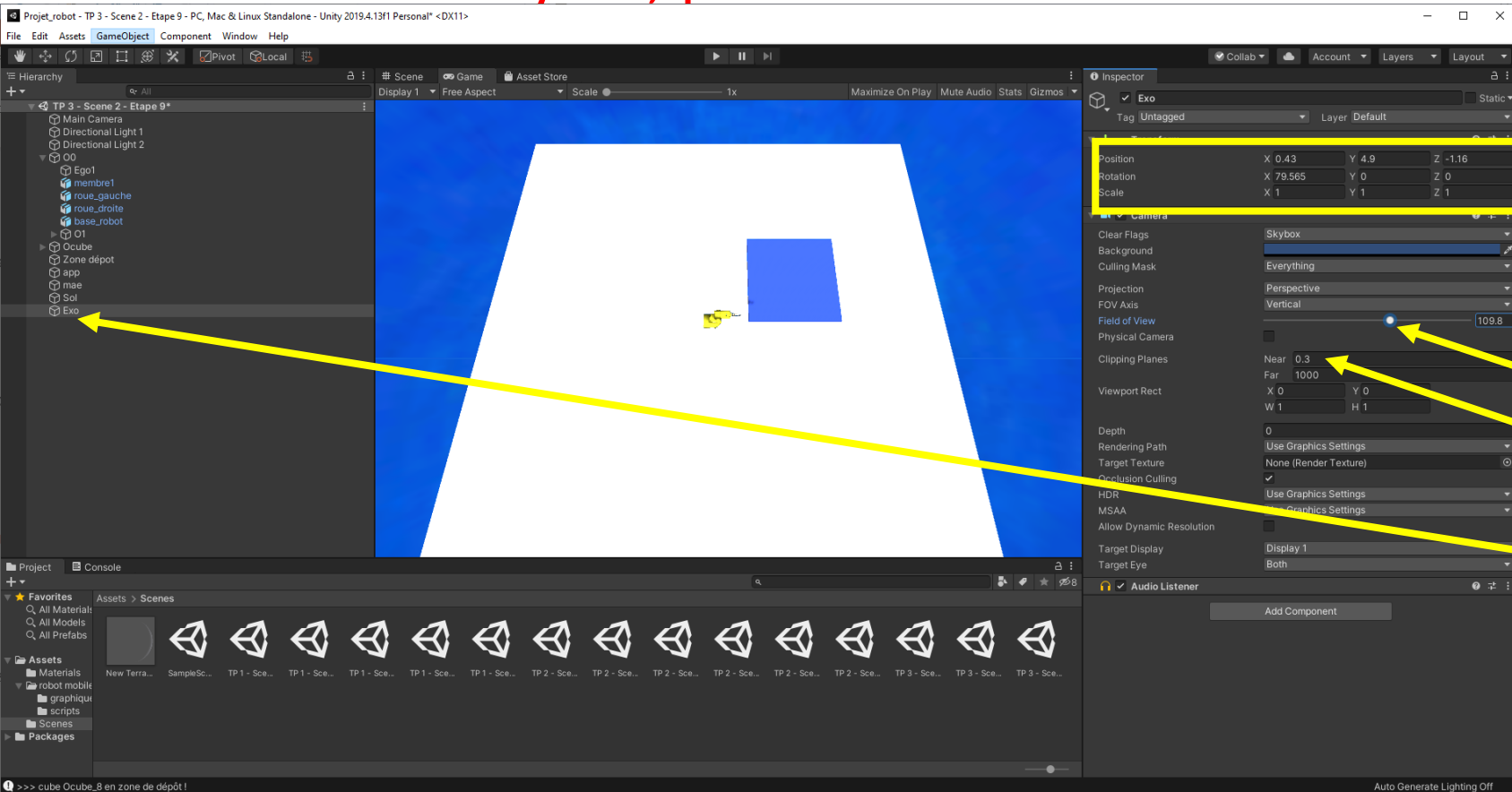
# PR N°3 MERCREDI 25 OCTOBRE 2023

## Contenu du PR n°3

- ❑ Le sujet de PR n°3 : [https://www.ibisc.univ-evry.fr/~fdavesne/ens/ens\\_efrei\\_m1/proj/doc/Projet\\_RV\\_M1\\_2021\\_TP3.pdf](https://www.ibisc.univ-evry.fr/~fdavesne/ens/ens_efrei_m1/proj/doc/Projet_RV_M1_2021_TP3.pdf)
- ❑ Vous êtes amenés à chercher par vous-même, en groupe. Il n'y aura pas de correction donnée à la fin du PR, mais uniquement à la fin du projet (pour les PR n°2 et n°3)

# PR N°3 MERCREDI 25 OCTOBRE 2023

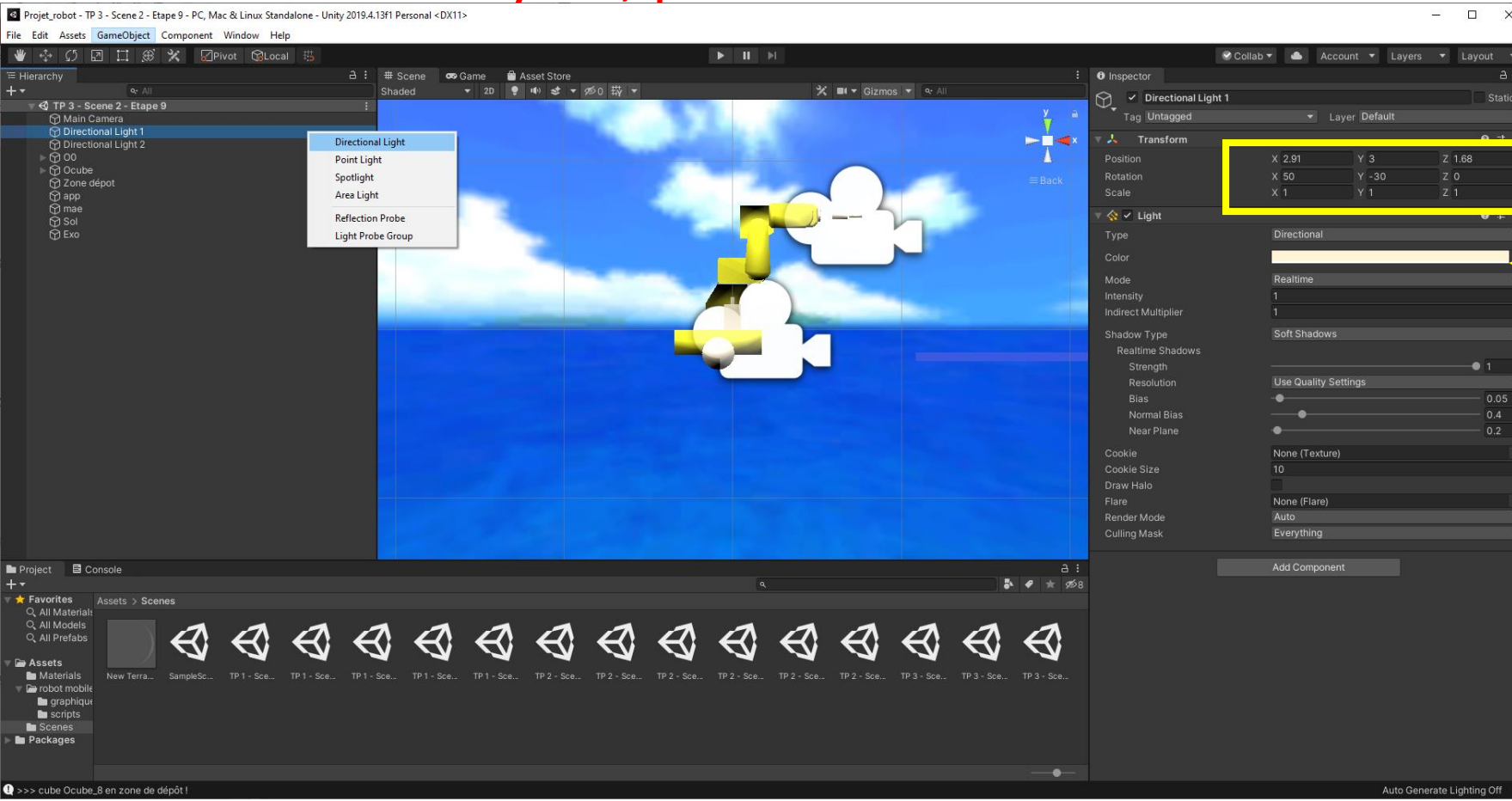
## Survol de Unity 3D, partie 3 – Création des caméras



- GameObject -> Caméra
- Transform à modifier
  - Position 3d
  - Rotation 3d
- Field of view à ajuster
- Clipping planes
  - Near à modifier
  - Far
- Exo, fille du Monde

# PR N°3 MERCREDI 25 OCTOBRE 2023

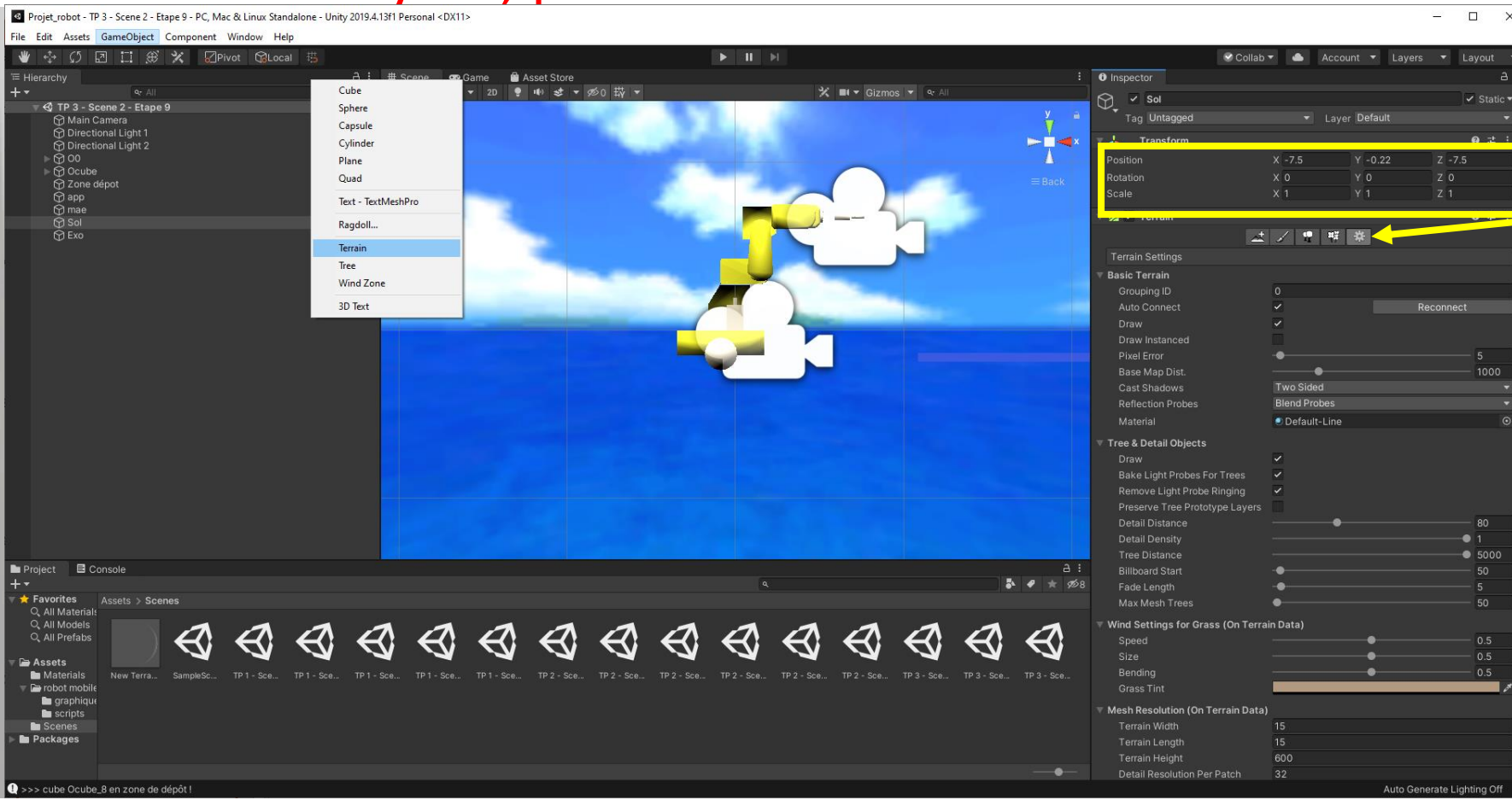
## Survol de Unity 3D, partie 3 – Création des lumières



- ☐ GameObject -> Light
  - Directional Light
- ☐ Transform à modifier
- ☐ Color à modifier

# PR N°3 EN DISTANCIEL, MARDI 24 NOVEMBRE 2020

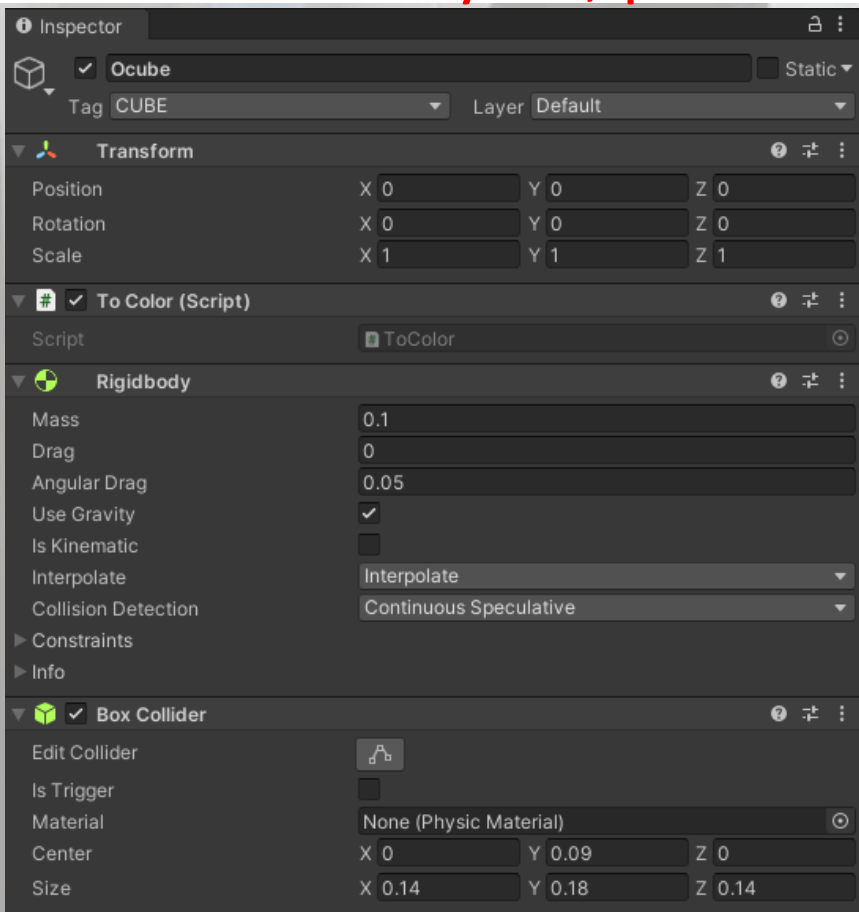
## Survol de Unity 3D, partie 3 – Création du sol



- GameObject -> Terrain
- Transform à modifier
- Terrain settings à modifier
  - Terrain Width
  - Terrain Height

# PR N°3 MERCREDI 25 OCTOBRE 2023

## Survol de Unity 3D, partie 3 – Physicalisation



- Physicalisation de l'EmptyObject Ocube
- Il faut rajouter des Components à Ocube
  - Component -> Physics -> Rigidbody
    - Mass, Use Gravity
  - Component -> Physics -> BoxCollider
    - Is Trigger gère l'événement de collision dans la fonction OnTriggerEnter(Collider other) du script supporté par l'objet courant ou dans la fonction OnTriggerExit(Collider other)
    - other.gameObject permet de connaître le GameObject ayant collisionner l'objet courant