

- Raccourcis :

- ALT + Clic Gauche : rotation.
- Clic Molette : déplacer position caméra.
- Clic Droit : Déplacement Vue FPS Caméra.
- Touche F : Focus sur l'objet sélectionné.
- Touche W : mode axe de déplacement.
- Ctrl + Shift + F : aligner la caméra avec la vue de la scène.
- Ctrl K + Ctrl D : aligner automatiquement.

- Commandes :

- //texte : permet de faire des commentaires.
- && : permet de faire des combinaisons.
- If : permet de faire des boucles, des conditions.
 - Le résultat ne peut être que vrai ou faux (1 ou 0; booléen).
- `Debug.Log("")` : permet d'écrire dans la console de Unity.
- `Input.` = class.
- `Input.GetKeyDown` : touche enfoncée.
- `Input.GetKeyUp` : touche relâchée.
- `Input.GetKey` : tant que la touche enfoncée puis relâchée/maintenue.
- `Input.GetAxis` : permet de définir des touches en fonctions des paramètres attribués dans Input Manager. `Input.GetAxis("Vertical")`. Dans Input Manager.
- `keyCode` : Code de la touche du clavier : `rightArrow`, `A`, `1`, /...
- `transform.position = new Vector3(x,y,z)` : permet d'avoir accès aux propriétés Transform (Scale, Position, Rotation) et d'effectuer un vecteur. **Rajouter un f à la fin de (x,y,z) pour les floats. (x,y,zf).**
- `transform.Translate(x,y,z)` : permet d'effectuer un déplacement.
- `valeur * Time.deltaTime` :
 - permet de calculer une valeur en fonction des secondes et non des frames.
 - uniforme en fonction du support.
- `[SerializeField]` : permet d'afficher la valeur modifiable dans l'inspecteur.
`[SerializeField] float = 2.`
- `GameObject` : objet dans le jeu.
- `Vector3.MoveTowards(position actuelle.position finale.vitesse)`.
- `Vector3` : variable de type position.
- `.transform.position` : position.
- `.Fonction` : une fonction.
- `transform.` : permet de modifier la position, la taille, l'échelle.
- `variable` : permet d'attribuer une valeur à des caractères.
- `Destroy()` : faire disparaître un objet.
- `Destroy(gameObject)` : détruire l'objet.
- `transform.LookAt(nomobjet.transform)` : permet de suivre un ennemi.
- `!=` : n'est pas.
- `Instantiate(objetàcréer)` : permet de créer un objet.
- `.GetComponent<BulletMvmt>().;` = permet de récupérer un component.*
- `(Time.time)` : temps en jeu, chronomètre du jeu.
- `readonly` : permet de mettre une valeur uniquement depuis l'extérieur. Pour éviter d'être modifié depuis l'inspecteur.
- `void nomfonction` : créer une fonction.

- for : boucle pour un certain nombre de fois // sinon if.
- Return; : permet de remonter directement dans le code et de ne pas lire le code situé en dessous.
- Coroutine : permet de faire une pause ou un délai dans un script.
- Variable statique : variable accessible par tous mais accessible une seule fois dans la scène.