

Gestion de caméras

Consignes pour l'exercice

Dans cet exercice, vous devez avoir 4 caméras ainsi qu'un gestionnaire de caméra qui permet de changer la vue active avec les touches 1,2,3 et 4.

- **Caméra première personne** : Positionnez la caméra présente dans le *Hierarchy* devant l'hélicoptère (ou à l'intérieur) et parentez-la à l'hélico en la glissant sur l'objet hélico. (nommée CameraFPS).
- **Caméra surveillance**: Ajoutez une caméra de type "surveillance" (nommée CameraSurveillance) dans le ciel à une distance **d'environ** 50 de l'hélicoptère.
 - Créez le script de cette caméra. Ce script permet à la caméra de pointer toujours vers l'hélicoptère sans se déplacer (voir `transform.LookAt()`). L'action se fait dans `Update()`.
- **Caméra à distance fixe**: Créez une autre caméra de type "caméra à distance fixe" (nommée CameraDistanceFixe) qui permet de suivre l'hélico à une distance fixe. (style LOL ou Diablo)
 - Créez le script de cette caméra. Utilisez une variable de type `Vector3` pour déterminer la distance désirée. Cette variable peut être initialisée dans l'inspecteur.
- **Caméra 3e personne**: Créez la caméra "3e personne avec mouvement de suivi fluide" et son script à l'aide de la technique vue dans les notes de cours des caméras.
 - Créez le script de cette caméra.
 - Utilisez une variable de type `Vector3` pour déterminer la distance entre la caméra et l'hélico (ex: `distanceCamera`) qui sera utilisée par la fonction `TransformPoint()`,
 - **Remarque**, un bug avec `TransformPoint()` : si le scale de l'hélico n'est pas (1,1,1) alors votre caméra peut se trouver trop loin de l'hélico. La distance est multipliée par le scale. Alors, divisez les valeurs par 100. exemple: au lieu de 10, il faut 0.1f et au lieu de 5 il faut 0.05f. Ce bug n'existe pas si le scale est fixé sur le modèle importé!
 - Utilisez une autre variable (ex: `ValeurTransition`) de type `float` pour déterminer le facteur d'amortissement pour la fonction `Lerp` qui positionnera la caméra. Faites des tests et ajustez sa valeur.
- Créez un objet vide pour gérer les changements de caméras (nommé GestionnaireCameras)
 - **Important: désactivez toutes les caméras sauf celle avec laquelle le jeu doit commencer.**
 - Ajoutez le **TAG MainCamera** sur chaque caméra
 - Créez le script de cet objet qui doit permettre de :
 - activer la caméra FPS à l'aide de la touche 1
 - activer la caméra 3e personne à l'aide de la touche 2
 - activer la caméra à distance fixe à l'aide de la touche 3
 - activer la caméra de surveillance à l'aide de la touche 4

Défis supplémentaires : (optionnels)

- Créez un script pour la caméra FPS qui permet de la faire trembler un peu en modifiant sa position de façon aléatoire (exemple: de 0.1 unité) à chaque frame en utilisant `transform.Translate(...Random.Range ...)` ou `transform.position = ...` ou
 - la caméra vibre seulement si les hélices tournent.
- Optimisez le code des gestions des caméras en utilisant un tableau de `GameObject`:
 - `public GameObject[] tableauCameras;`
 - Chaque élément du tableau est un `GameObject` alors on peut faire:
`TableauCameras[0].transform.position` ou `TableauCameras[0].SetActive(true)`, etc.
 - Identifiez la taille du tableau et associez les caméras aux éléments du tableau dans l'inspecteur.
 - Modifiez la caméra selon la touche appuyée:
 - `if(Input.GetKeyDown("1")) ActiverCamera(1);` //le **numéro** de la caméra est passé à la fonction

```
....  
void ActiverCamera(???)  
{  
    ....  
}
```