

J01-02 – Exercices

Objectifs

- Découvrir la gestion du moteur physique

Gérer des inputs

Exercice 1

Créer un script qui affiche quelque chose dans la console quand on appuie sur la barre espace. Attacher ce script à un GameObject sur la scène et vérifiez que votre texte est bien affiché

Exercice 2

Créer un script très simple permettant de diriger un personnage selon les axes X et Z :

- En appuyant sur la flèche haut, le GameObject doit se déplacer en positif selon l'axe Z
- En appuyant sur la flèche bas, le GameObject doit se déplacer en négatif selon l'axe Z
- En appuyant sur la flèche droite, le GameObject doit se déplacer en positif selon l'axe X
- En appuyant sur la flèche gauche, le GameObject doit se déplacer en négatif selon l'axe X

Gestion de la physique

Exercice 3

Créer une nouvelle scène. Sur cette scène, posez des obstacles via des primitives afin de créer une sorte de flipper. Ajouter plusieurs sphères en haut de ce flipper et leur ajouter tout ce qui est nécessaire pour qu'elles tombent et ne passent pas à travers les obstacles.

Exercice 4

Créer des zones de détection en bas du flipper qui doivent réagir lorsqu'une bille passe dedans. Utilisez les événements du moteur physique pour cela et écrivez dans la console un message lorsqu'une bille passe dans une de ces zones.

Exercice 5

Créer un nouveau script représentant un objet collectionnable que le joueur pourra ramasser. Utiliser le même principe que pour l'exercice précédent. Lorsque l'objet est ramassé, il faut supprimer cet objet de la scène.

Créer une nouvelle scène sur laquelle poser plusieurs objets « collectionnables ». Ajouter tout ce qui est nécessaire afin que le joueur puisse déplacer le personnage et ramasser les objets.

Exercice bonus

Modifier le contrôleur de personnage pour gérer la rotation

- Appuyer sur la touche gauche ou droite fait tourner le personnage selon l'axe Y
- Appuyer sur la flèche haut ou bas fait avancer ou reculer le personnage dans la direction vers laquelle il regarde

Aide : cherchez comment utiliser le vecteur « forward » présent sur le composant Transform.