

EXERCICE INTRODUCTION UNITY

À faire individuellement

CRÉER UN PROJET GITHUB

1. Appeler ce projet Rocket

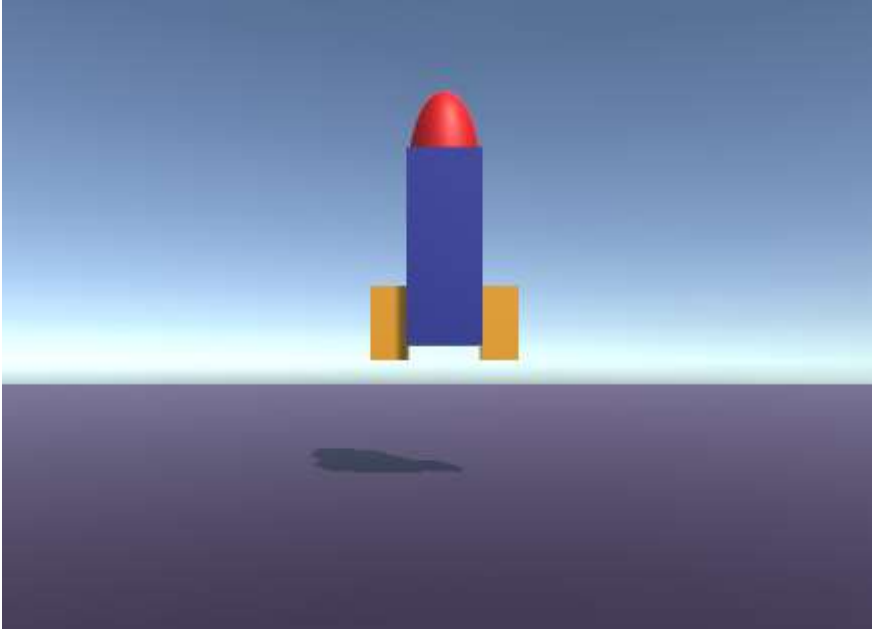
Vous êtes libre de faire vos commits lorsque vous le jugez opportun

CRÉER UN PROJET UNITY (3D)

1. Appeler ce projet Rocket
2. Renommer la scene « sandBox »

CRÉER UNE FUSÉE

1. Créer un gameObject vide, appeler le Rocket
2. À l'aide d'un cube, créer le corps d'une fusée, appeler le RocketBody
3. Mettre le RocketBody comme sous objet du gameObject Rocket
4. À l'aide d'un cube, créer un réacteur pour la fusée, appeler le Booster
5. Mettre le Booster comme un sous objet du gameObject Rocket
6. À l'aide de votre premier réacteur et de CTRL+D créer votre second réacteur. Il devrait être un sous objet du gameObject Rocket.
7. À l'aide de CTRL+Shift et des propriétés des réacteurs dans l'inspecteur, attacher correctement les réacteurs au corps de la fusée.
8. À l'aide d'une sphère et des propriétés de dimension (scale) créer le nez de votre fusée. Appeler le RocketNose
9. Mettre le RocketNose comme sous-objet de l'objet Rocket
10. À l'aide d'un cube, créer une plateforme.
11. Ajouter un des Box colliders à votre fusée ainsi qu'à ses réacteurs.
12. Ajouter un box collider à votre plateforme.
13. Ajouter un sphere collider à votre RocketNose
14. Ajuster le sphere collider pour qu'il ait une dimension acceptable
15. Ajouter un Rigidbody à Rocket
16. Donner une masse de 10
17. Votre fusée devrait être attirée par la gravité et entrer en collision avec votre plateforme
18. Faire un prefab de votre fusée
19. Créer 3 materials avec de la couleur : RocketNose, RocketBody et RokerBooster
20. Appliquer les materials sur les composantes de votre Rocket.



AJOUTER UN COMPORTEMENT À LA FUSÉE

1. Ajouter un script à Rocket
2. Appeler le script Rocket
3. Ouvrir le script avec VisualStudio
4. Créer une variable private de type Rigidbody
5. Dans la méthode Start, récupérer le Rigidbody de la Rocket pour le stocker dans la variable créer au point 3.
6. Dans la méthode update, utiliser input détecter si la barre d'espace est appuyée
7. Si la barre d'espace est appuyée, appliquer une force relative vers le haut à la fusée.
8. Transformer la détection de la barre d'espace en la détection de l'axe « Jump »
9. Utiliser le component rigidBody pour restreindre la fusée en position sur l'axe des Z et en rotation sur l'axe des Y et des Y

Votre fusée devrait décoller lors que la barre d'espace est enfoncée et retomber au sol lorsque la barre d'espace est relâchée.

FAIRE TOURNER VOTRE FUSÉE

1. Ouvrir le script Rocket
2. Détecter à l'aide de la variable « Input » la variation de l'axe horizontal
3. À l'aide de la méthode addForceAtPosition, appliquer une force à un endroit stratégique pour forcer une rotation vers la gauche ou la droite
4. Créer deux champ SerializeField pour donner une valeur dynamique au trust de la fusée et à la force de relation.

TESTER VOTRE JEU

1. Créer une nouvelle plateforme

2. À partir de votre point de départ, faire atterrir votre fusée sur votre nouvelle plateforme.

ORGANISER VOS ASSETS

1. Créer des répertoires et organiser vos assets.

REMISE

Me montrer le projet fonctionnel.

Conserver ce projet pour usage futur.

Transmettre un MIO avec le lien de votre repository du projet