

Quiz : Bonnes Pratiques de Débogage (Logs, Détection d'Erreurs Courantes)

1. Quelle est la principale fonction des logs dans le processus de débogage ?

- a) Créer des animations dans le jeu
- b) Suivre l'exécution du code et détecter les erreurs
- c) Ajouter des sons et effets visuels
- d) Gérer les entrées des utilisateurs

Réponse correcte : b) Suivre l'exécution du code et détecter les erreurs

2. Quel type de log est le plus adapté pour signaler un problème grave qui empêche le jeu de continuer normalement ?

- a) `Debug.Log()`
- b) `Debug.LogError()`
- c) `Debug.LogWarning()`
- d) `Debug.LogInfo()`

Réponse correcte : b) `Debug.LogError()`

3. Quelle est la différence entre `Debug.Log()` et `Debug.LogWarning()` dans Unity ?

- a) `Debug.Log()` affiche des erreurs, tandis que `Debug.LogWarning()` affiche des informations de débogage
- b) `Debug.Log()` est utilisé pour afficher des informations générales, tandis que `Debug.LogWarning()` signale des avertissements sans stopper l'exécution
- c) `Debug.Log()` est uniquement pour les logs d'erreurs, tandis que `Debug.LogWarning()` est pour les performances
- d) Il n'y a aucune différence

Réponse correcte : b) `Debug.Log()` est utilisé pour afficher des informations générales, tandis que `Debug.LogWarning()` signale des avertissements sans stopper l'exécution

4. Quelles informations peuvent être utiles à afficher dans un log de débogage lors de l'inspection d'un objet dans le jeu ?

- a) La taille de l'objet uniquement
- b) L'état des variables pertinentes, comme la position, la vitesse ou les interactions
- c) Le nom de l'éditeur du jeu
- d) Le genre de musique de fond qui est joué

Réponse correcte : b) L'état des variables pertinentes, comme la position, la vitesse ou les interactions

5. Lors de l'utilisation d'un débogueur, que permet un "point d'arrêt" ?

- a) Faire en sorte que le jeu s'arrête complètement
- b) Visualiser les textures du jeu
- c) Arrêter l'exécution du code à un endroit spécifique pour examiner les variables et le comportement du jeu
- d) Sauvegarder automatiquement les progrès du joueur

Réponse correcte : c) Arrêter l'exécution du code à un endroit spécifique pour examiner les variables et le comportement du jeu

6. Quel outil dans Unity permet de surveiller les performances en temps réel et d'identifier les goulots d'étranglement ?

- a) Profiler Unity
- b) Outils de modélisation 3D
- c) `Debug.Log()`
- d) Asset Store

Réponse correcte : a) Profiler Unity

7. Pourquoi est-il important de désactiver les logs en production ?

- a) Pour augmenter la qualité des graphismes
- b) Pour améliorer la performance en réduisant la charge sur le processeur et éviter la surcharge du système
- c) Parce que les logs sont toujours inutiles
- d) Parce qu'ils affectent le comportement du joueur

Réponse correcte : b) Pour améliorer la performance en réduisant la charge sur le processeur et éviter la surcharge du système

8. Quel type d'erreur signale un problème qui empêche l'exécution correcte du jeu, souvent lié à une exception ou un plantage ?

- a) Log d'information
- b) Log de débogage
- c) Log d'erreur
- d) Log critique

Réponse correcte : d) Log critique

9. Lorsqu'un objet dans le jeu traverse un autre objet ou se déplace de manière inattendue, quelle pourrait être la cause courante à vérifier ?

- a) Le type de musique joué dans le jeu
- b) La configuration incorrecte des **Colliders** et des **Rigidbody**s
- c) Les couleurs des objets
- d) La luminosité de l'environnement

Réponse correcte : b) La configuration incorrecte des **Colliders** et des **Rigidbody**s

10. Quel est l'objectif de l'utilisation de tests unitaires dans le processus de débogage ?

- a) Vérifier les performances graphiques du jeu

- b) Assurer que chaque composant individuel du code fonctionne correctement avant de l'intégrer au reste du jeu
- c) Ajouter des animations
- d) Créer un menu de démarrage

Réponse correcte : b) Assurer que chaque composant individuel du code fonctionne correctement avant de l'intégrer au reste du jeu