

# Quiz avec Solutions : Ajout d'un Système de Score et Affichage dans l'UI

1. Quelle méthode dans Unity permet d'afficher dynamiquement un score à l'écran ?

- a) `UpdateText()`
- b) `Text.SetText()`
- c) `Text.text`
- d) `SetScoreDisplay()`

Réponse correcte : c) `Text.text`

---

2. Dans quel script doit-on ajouter une référence au composant `Text` pour afficher le score dans l'UI ?

- a) `ScoreManager`
- b) `PlayerController`
- c) `GameManager`
- d) `UIController`

Réponse correcte : a) `ScoreManager`

---

3. Quel est l'objectif principal de la méthode `AddScore()` dans le script `ScoreManager` ?

- a) Incrémenter le score du joueur.
- b) Sauvegarder le score dans un fichier.
- c) Afficher le score à l'écran.
- d) Réinitialiser le score du joueur.

Réponse correcte : a) Incrémenter le score du joueur.

---

#### 4. Quelle classe Unity est utilisée pour manipuler le texte dans l'interface utilisateur (UI) ?

- a) `GameObject`
- b) `Text`
- c) `TextMeshPro`
- d) `Slider`

Réponse correcte : b) `Text`

---

#### 5. Que fait la méthode `PlayerPrefs.SetInt("PlayerScore", score)` dans Unity ?

- a) Elle sauvegarde le score dans un fichier texte.
- b) Elle récupère le score sauvegardé.
- c) Elle définit le score dans le système de sauvegarde de Unity.
- d) Elle sauvegarde le score dans le registre système.

Réponse correcte : c) Elle définit le score dans le système de sauvegarde de Unity.

---

#### 6. Comment afficher le score en temps réel dans l'interface utilisateur ?

- a) En appelant `UpdateScore()` chaque fois que le score change.
- b) En modifiant la variable `score` directement dans le script `Text`.
- c) En assignant la valeur du score à `scoreText.text`.
- d) En appelant la fonction `SetText()` du composant `Text`.

Réponse correcte : c) En assignant la valeur du score à `scoreText.text`.

---

#### 7. Quelle méthode permet de charger le score précédemment sauvegardé dans PlayerPrefs ?

- a) `PlayerPrefs.LoadInt("PlayerScore")`
- b) `PlayerPrefs.GetInt("PlayerScore")`
- c) `PlayerPrefs.Load("PlayerScore")`
- d) `PlayerPrefs.FetchInt("PlayerScore")`

Réponse correcte : b) `PlayerPrefs.GetInt("PlayerScore")`

---

## 8. Quel est l'usage principal de la méthode

`Destroy(gameObject)` dans le script `CollectibleItem` ?

- a) La destruction de l'objet collecté après avoir ajouté au score.
- b) La destruction du score à chaque nouvelle collecte.
- c) La désactivation de l'objet collecté pour ne pas l'afficher.
- d) La sauvegarde du score de l'objet collecté.

Réponse correcte : a) La destruction de l'objet collecté après avoir ajouté au score.

---

## 9. Dans quel cas utiliseriez-vous `PlayerPrefs.Save()` dans le cadre de la gestion du score ?

- a) Pour sauvegarder l'interface utilisateur.
- b) Pour finaliser et enregistrer toutes les données dans PlayerPrefs.
- c) Pour afficher le score à l'écran.
- d) Pour réinitialiser le score.

Réponse correcte : b) Pour finaliser et enregistrer toutes les données dans PlayerPrefs.

---

## 10. Comment incrémenter le score chaque fois qu'un objet collectable est touché ?

- a) Appeler `AddScore()` dans la fonction `Update()`.
- b) Appeler `AddScore()` lorsque l'objet touche un ennemi.
- c) Appeler `AddScore()` lorsque l'objet est collecté par le joueur.
- d) Appeler `SaveScore()` chaque fois que l'objet est collecté.

Réponse correcte : c) Appeler `AddScore()` lorsque l'objet est collecté par le joueur.

---

## Révision des Concepts Clés

1. **Système de Score** : Le score du joueur peut être manipulé à l'aide de méthodes comme `AddScore()` et affiché via le composant `Text` ou

`TextMeshPro` .

2. **Sauvegarde du Score** : Unity offre des outils comme `PlayerPrefs` pour stocker le score, permettant au joueur de reprendre sa progression lors des prochaines sessions.
3. **Interaction avec l'UI** : Le score est généralement mis à jour et affiché en temps réel dans l'UI en assignant la nouvelle valeur à un composant `Text` .