

# QCM : Vers un code Immuable

**Lien du Kahoot :** <https://play.kahoot.it/v2/?quizId=3ab4d3ba-42b6-419d-bc65-3730c851e23a&hostId=00605cf2-2e2e-407a-83ff-3d1144fb10cc>

## 1. Qu'est-ce qu'un objet immuable ?

- a) Un objet dont les propriétés peuvent changer après sa création
- b) Un objet dont les propriétés ne peuvent pas changer après sa création
- c) Un objet qui ne peut pas être créé
- d) Un objet qui utilise des collections

## 2. Quels sont les avantages de l'immuabilité ?

- a) Facilite la gestion de la mémoire
- b) Permet la modification des données
- c) Réduit les erreurs en environnement multi-threadé
- d) Augmente le code "boilerplate"

## 3. Pourquoi les objets immuables sont-ils adaptés aux environnements multi-thread ?

- a) Ils peuvent modifier des données partagées sans conflit.
- b) Leur état ne peut pas être modifié après leur création.
- c) Ils implémentent des mécanismes de synchronisation internes.
- d) Ils interdisent la création de nouvelles instances par plusieurs threads.

## 4. Pourquoi l'immuabilité est-elle importante en programmation ?

- a) Elle permet de modifier les objets à tout moment
- b) Elle rend le code plus complexe
- c) Elle facilite l'héritage
- d) Elle évite les bugs liés à l'accès concurrent aux objets

## 5. Quelle est une différence majeure entre une classe traditionnelle et un record en Java ?

- a) Exige que tous les champs soient déclarés comme `public`.
- b) Peut avoir des setters pour ses attributs.
- c) Est plus lent qu'une classe traditionnelle.
- d) Génère automatiquement des méthodes `equals()`, `hashCode()` et `toString()`

## 6. Quel code crée un record en Java représentant un point dans un plan avec des coordonnées x et y ?

- a) `public class Point(int x, int y) {}`
- b) `public record Point(int x, int y) {}`
- c) `Point(x, y) {}`
- d) `record Point { int x; int y; }`