

# POURQUOI UNE CLASSE ABSTRAITE

Prenons l'exemple suivant : vous avez une classe *Humain*, à partir de laquelle dérivent la classe *Homme* et la classe *Femme*. En toute logique, *Homme* et *Femme* sont instanciables (les objets créés ont une existence en soi), mais la classe *Humain* sera déclarée abstraite car un objet *Humain* n'existe pas en tant que tel, puisqu'il manque l'information sur le sexe. Ici, la classe *Humain* servira à implémenter des méthodes qui seront utilisées à la fois pour *Homme* et pour *Femme*.

## INTERFACE VS CLASSE ABSTRAITE

- Les interfaces ne peuvent pas avoir de propriétés, alors que les classes abstraites le peuvent.
- Toutes les méthodes des interfaces doivent être publiques, tandis que les méthodes des classes abstraites sont publiques ou protégées.
- Toutes les méthodes d'une interface sont abstraites, elles ne peuvent donc pas être implémentées dans le code et le mot-clé `abstract` n'est pas nécessaire.
- Les classes peuvent implémenter une interface tout en héritant d'une autre classe en même temps.