

---

# JAVA – ORM : Comprendre la persistance des objets relationnels

---

Paul EDIDE  
contact@gmail.com

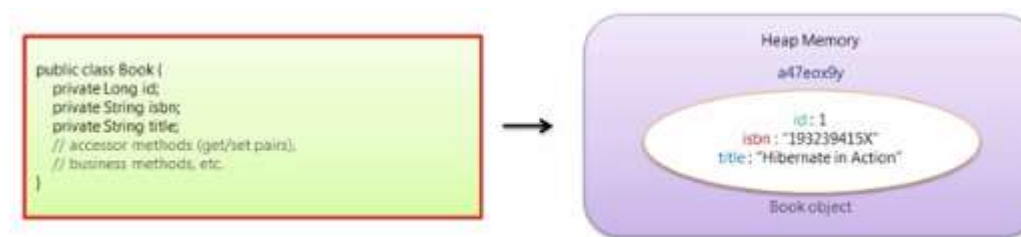
# PLAN



- 
- I- La persistance des objets
  - II- Modèle objet et modèle relationnel
  - III- Cas pratique – installation d'Eclipse

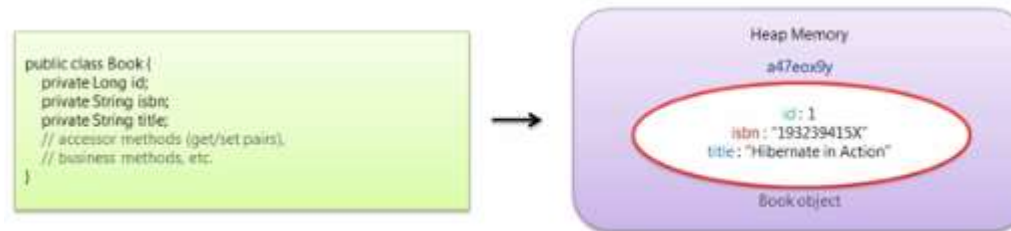
# I- La persistance des objets

- Dans cette session, nous allons parler de persistance des objets.
- Alors, qu'est-ce la persistance d'un objet ?
- Essayons de le comprendre à l'aide d'un exemple.
- Supposons que vous travaillez sur une application Java et que l'une des classes que vous avez dans votre application est la classe Book, qui contient des attributs et des méthodes (des champs de données).
- Supposons donc que vous ayez créé un objet de cette classe Book dans votre application.



# I- La persistance des objets

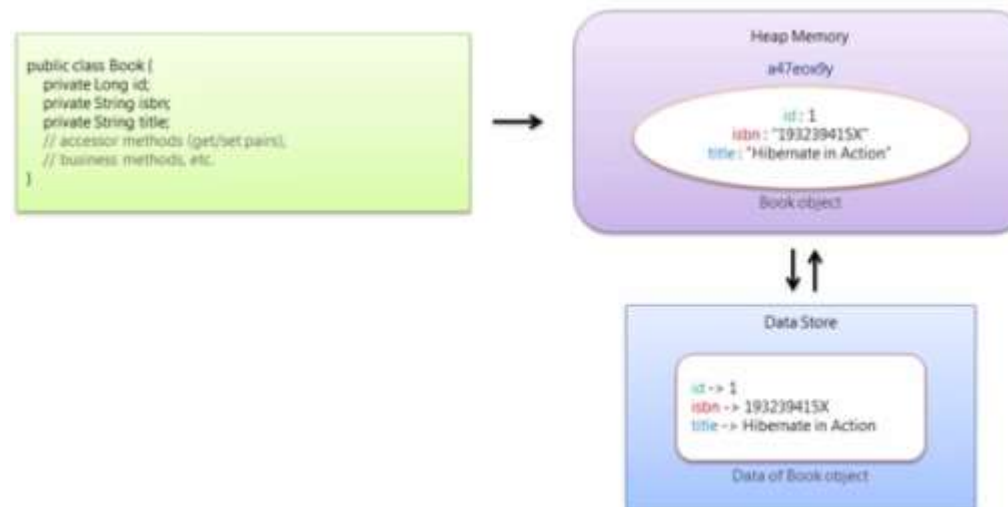
- Voici l'état de cet objet Book nouvellement créé dans la mémoire Java
- Maintenant, votre application s'exécute sur une machine virtuelle Java.



- Qu'est-ce que cela signifie ?
- Cela veut dire que l'état de chaque objet que vous créez dans votre application est créé par une machine virtuelle Java.
- Que pensez-vous qu'il se passera si la même machine virtuelle Java sur laquelle votre application est exécutée soit arrêtée ?

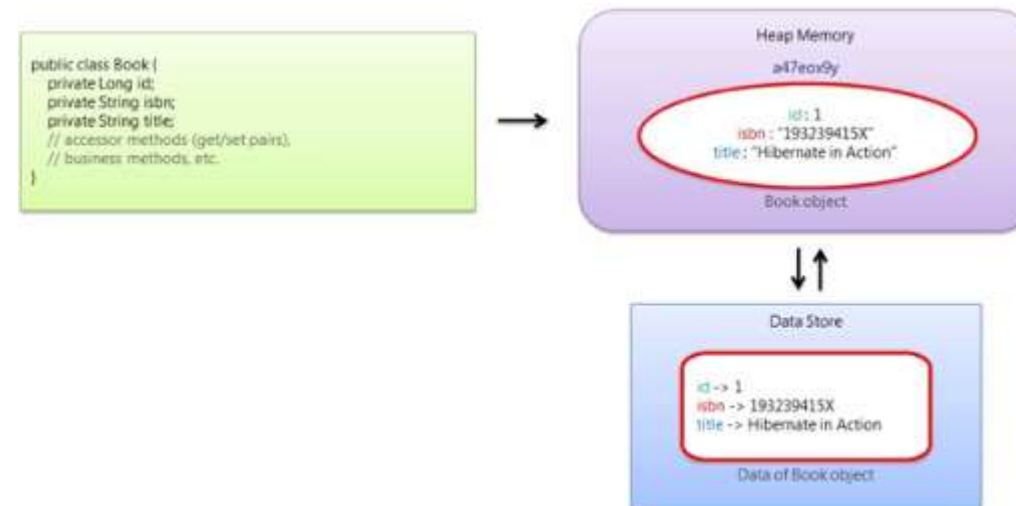
# I- La persistance des objets

- La réponse à cette question est que vous allez perdre l'état de cet objet Book dans votre application, ce qui est la dernière chose que vous souhaitiez dans votre application, à savoir que tous les objets que vous avez créés dans votre application ont perdu leur état.
- Vous perdrez toutes les données de votre application de cette façon.
- Alors, quelle est la solution ?



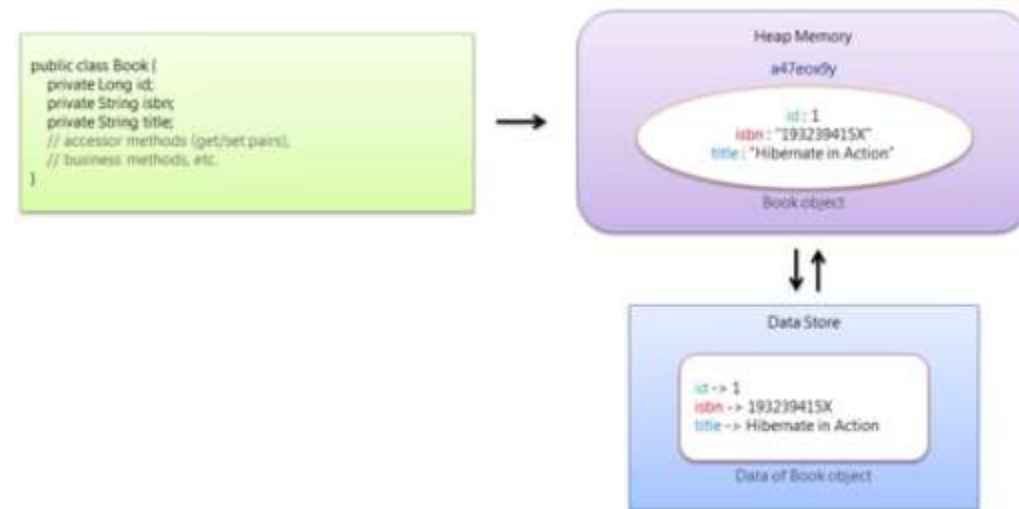
# I- La persistance des objets

- La solution consiste à enregistrer les données de cet objet Book dans un magasin de données.
- Ainsi, chaque fois que vous avez besoin du même état de cet Book dans votre application, vous disposez des données correspondantes dans ce magasin de données.



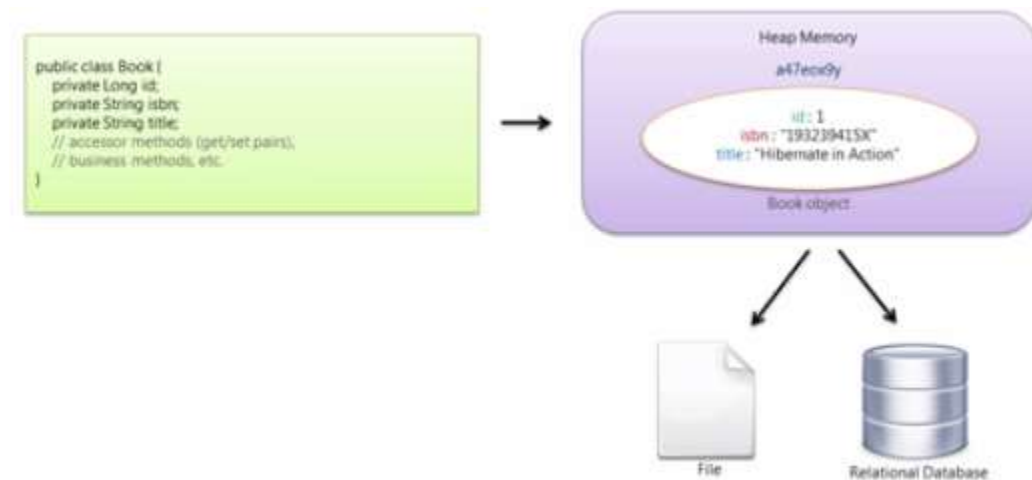
# I- La persistance des objets

- Ainsi, même si la machine virtuelle de Java n'est pas en cours d'exécution, vous disposez des données de cet objet Book dans ce magasin de données.
- Ce que vous voulez fondamentalement, c'est que l'état de tous les objets de votre application se situe au-delà de la portée de Java Virtual Machine.



# I- La persistance des objets

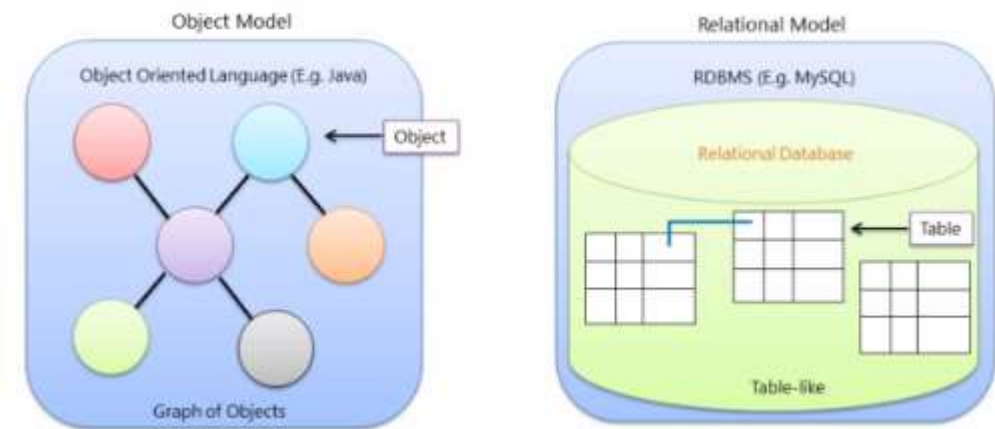
- Ainsi, lorsque nous parlons de persistance d'objet ou d'objet persistant, cela signifie que l'état d'un objet peut être sauvegardé dans un magasin de données et recréé ultérieurement.
- Il existe de nombreuses façons de rendre les données persistantes en Java. Vous pouvez conserver ces données dans un fichier ou une base de données relationnelle, mais il existe également de nombreuses autres manières.
- Mais dans ce cours, les objets Java persistants seront conservés dans une base de données relationnelle.





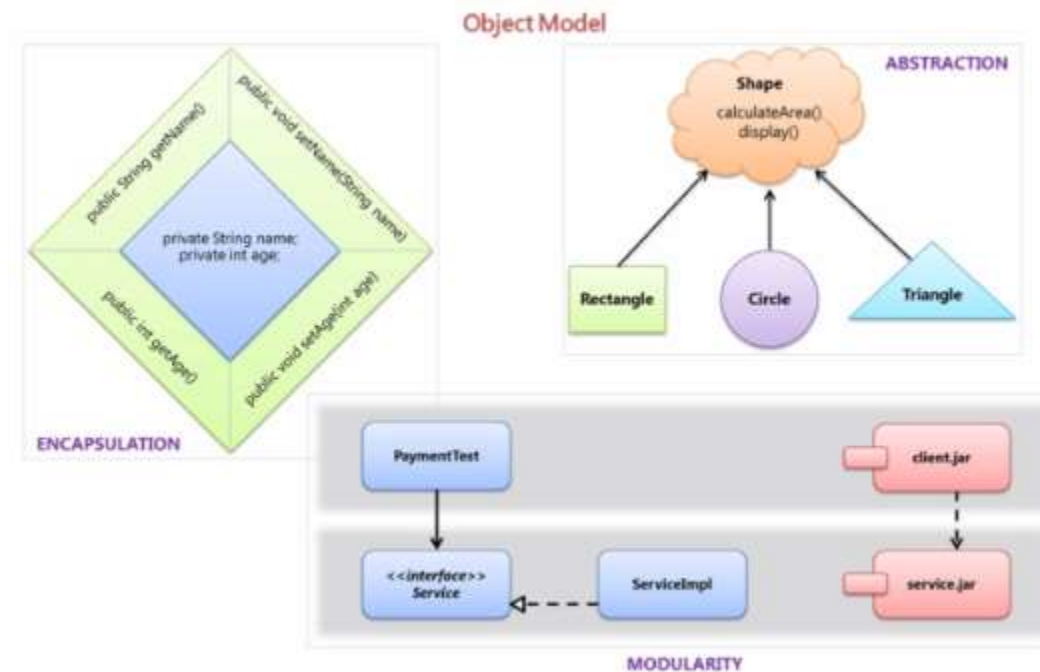
# II- Le modèle objet et le modèle relationnel

- Dans cette partie, nous aurons un aperçu rapide du modèle objet et du modèle relationnel.
- Java est un langage de programmation orienté objet qui représente les données sous forme de graphe d'objets interconnectés, tandis qu'un SGBDR représente les données dans un format de type tableau.
- Dans ce cours, nous allons stocker les objets Java dans une base de données relationnelle, mais Java va implémenter le modèle objet, alors qu'un SGBDR implémente le modèle relationnel.



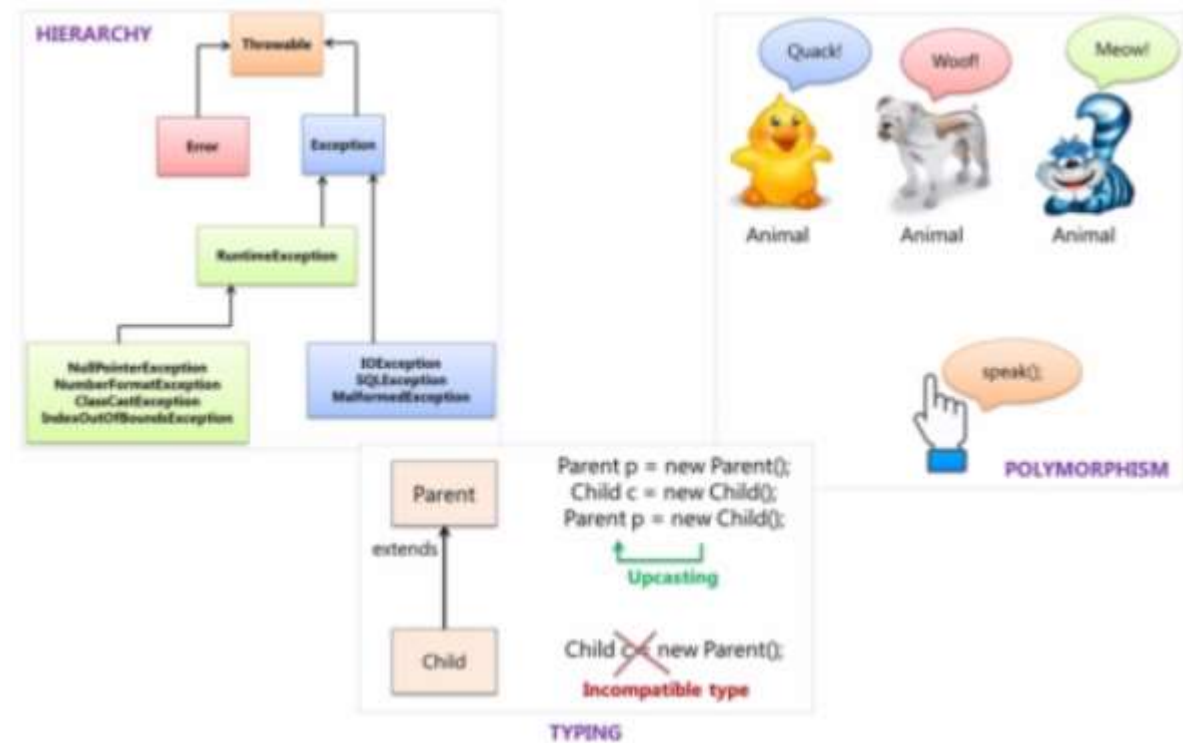
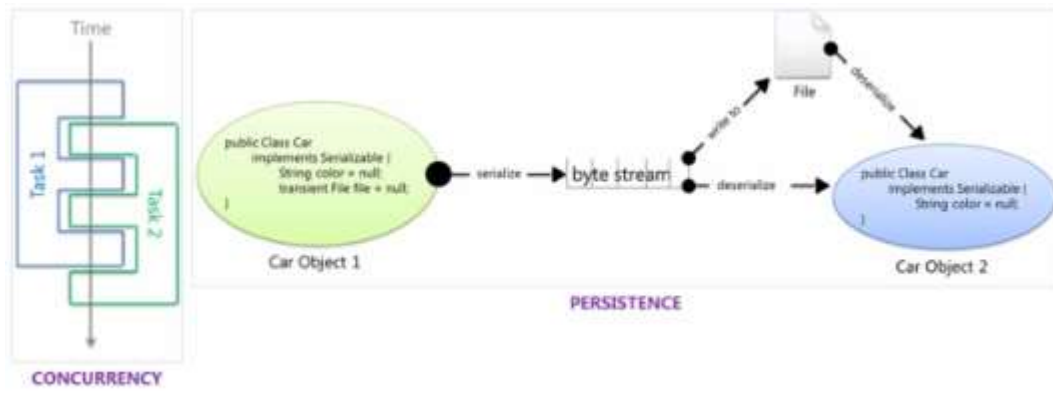
# II- Le modèle objet et le modèle relationnel

- Alors, qu'est-ce qu'un modèle objet et un modèle relationnel ?
- Commençons par le modèle objet.
- Un modèle objet utilise les principes d'abstraction, d'encapsulation, de modularité,



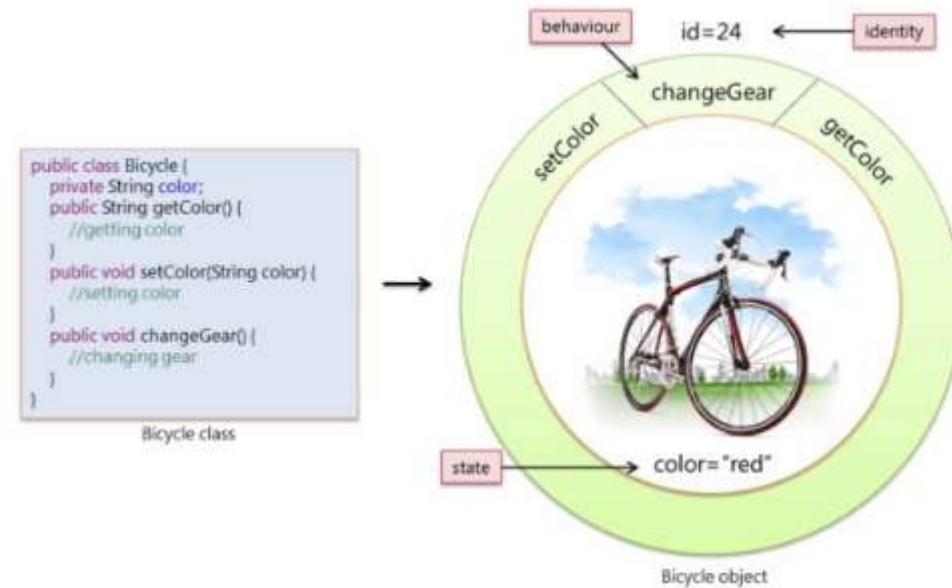
# II- Le modèle objet et le modèle relationnel

- de typage, de polymorphisme, de simultanéité et de persistance.



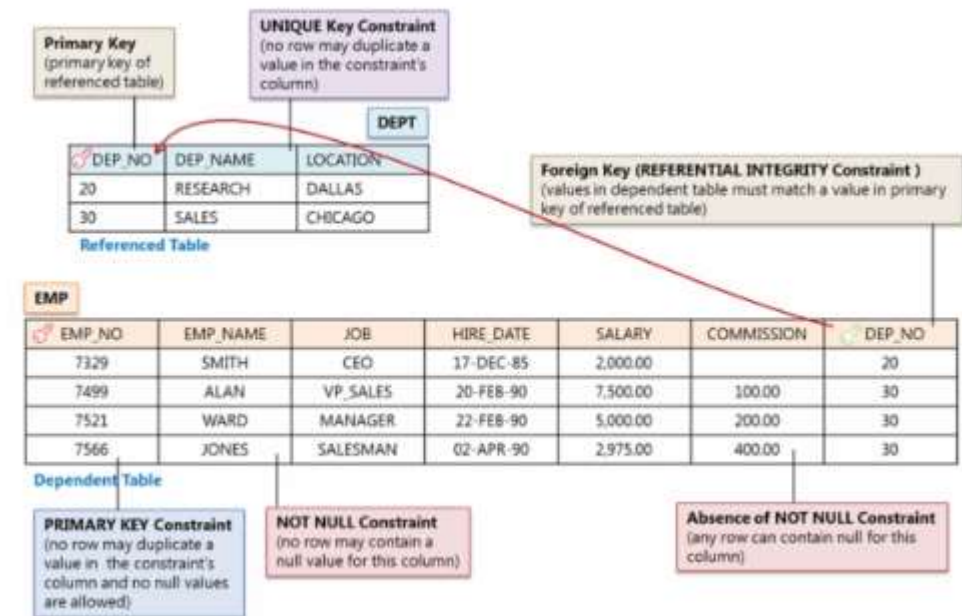
# II- Le modèle objet et le modèle relationnel

- Comme un objet une instance d'une classe, il y a une identité, un état et un comportement.
- Dans un système de modèle objet, nous avons des objets comme composants du système, ce qui nous permet de construire des systèmes complexes et réutilisables bien structurés.



# II- Le modèle objet et le modèle relationnel

- Alors qu'un modèle relationnel définit la structure des données, leur manipulation et leur intégrité.
- Et dans un modèle relationnel, les données sont organisées sous forme de table et différentes tables sont associées les unes aux autres par une intégrité référentielle et une intégrité de contrainte telle qu'une contrainte de clé primaire.
- Toutefois, les contraintes non nulles sont utilisées pour maintenir l'intégrité d'une entité.
- Il s'agissait donc d'un aperçu rapide du modèle objet et du modèle relationnel.



# III-Cas pratique

---

