

V) LA SPECIALISATION ET L'HERITAGE

V. 1) Principe

- Héritage des attributs et des méthodes
- Surcharge et redéfinition possible des méthodes.
- Pas d'héritage multiple.
- Pas de hiérarchie des tables, ni d'inclusion des objets d'un table dans l'autre.

-- Le super-type

CREATE TYPE *superType* **AS OBJECT**

(*nomAttribut₁* *nomType*,

...

)

Propriétés relatives aux sous-types & à l'instanciation

/

Propriétés

NOT INSTANTIABLE pour un type qui ne permettra pas le stockage d'instances,

NOT FINAL pour un type qui a des sous-types.

La valeur par défaut est : **INSTANTIABLE FINAL**.

Une banque gère des comptes pour ses clients. Certains comptes sont des comptes d'épargne.

OR.27 Créer le type compte_type avec numero, solde & date d'ouverture comme propriétés.

V. 2) Création de sous-types

-- Un sous-type

CREATE TYPE *sousType* **UNDER** *superType*

(*nomAttributs&méthodes supplémentaires*)

/

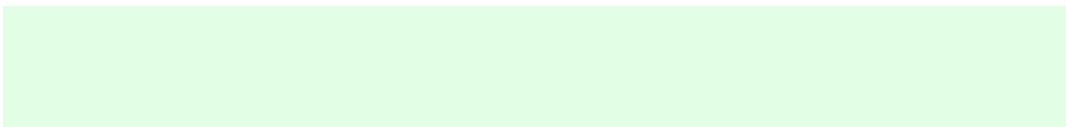
OR.28 Le compte_epargne_type est un type de compte particulier, il comporte des propriétés supplémentaires comme le taux de rémunération et le nombre d'années avant liquidation possible.



V. 3) Pas de hiérarchie des tables

Les tables sont indépendantes et il n'y a pas d'inclusion des objets d'un table dans l'autre. Mais la TOR associée au supertype va pouvoir contenir des objets du supertype et de ses sous-types.

OR.29 Créer les tables nécessaires au stockage des objets comptes & compte_épargne.



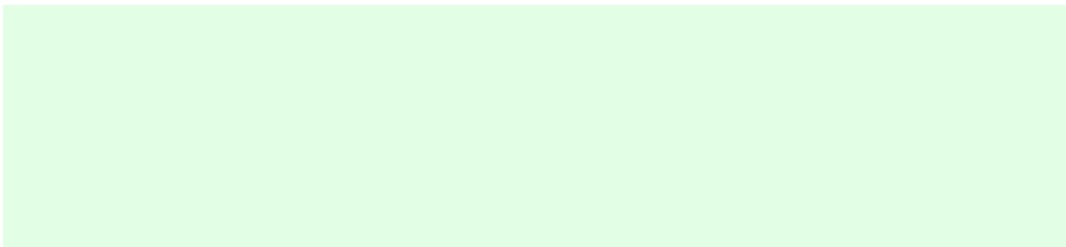
V. 4) Les méthodes

Par défaut, les méthodes sont non finales mais si l'on spécifie FINAL, elles ne peuvent être redéfinies.

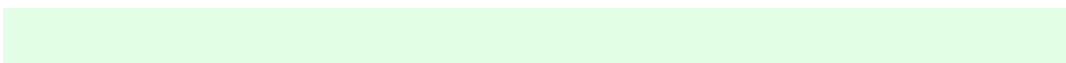
La surcharge offre la possibilité d'avoir plusieurs méthodes de même nom dans le même type avec des paramètres différents (en nombre ou en type). Le code à exécuter sera déterminé lors de l'appel de la méthode. Cf. le polymorphisme.

La redéfinition permet de préciser le comportement d'une méthode dans le sous-type.

OR.30 Créer pour les comptes la méthode valeur_francs qui effectue une conversion.



OR.31 Pour les comptes épargne, la méthode de conversion tient compte des particularités de ces comptes : on va pouvoir connaître la valeur en franc après valorisation du compte mais attention cette somme ne sera effective qu'après la date d'échéance du compte.



V. 5) Insertion d'objets & spécialisation

OR.32 Ajouter des comptes et des comptes d'épargnes dans notre base de données.

V. 6) Interrogation & spécialisation

➤ Affichage des objets et de leurs type (sous/super)

OR.33 Afficher tous les comptes.

VALUE renvoie les valeurs mais aussi le type.

➤ Interrogation avec accès méthode

OR.34 Afficher la valorisation des comptes convertis en francs.

➤ Interrogation sur l'appartenance à un (sous-)type OF

OR.35 Afficher les comptes d'épargne.

OR.36 Afficher les informations des comptes d'épargne.

ONLY ôterait les instances des sous-types de `compte_epargne_type`

La fonction IS OF assure la sélection des objets de table à un certain niveau de la hiérarchie.

➤ **Possible accès aux attributs et aux méthodes du sous-type**

(Si l'objet est effectivement de ce sous-type-là)

TREAT

OR.37 Afficher les données relatives aux comptes d'épargne.

-- Les attributs du supertype sont accessibles par l'alias de la table : c.numero...

TREAT(...). ne marche pas*

OR.38 Calculer la valorisation des comptes d'épargne.