

Les données de l'entreprise de vente de produits informatiques

1. Présentation générale du laboratoire

Ce quatrième laboratoire vise à mettre en pratique vos apprentissages en développement Java (ou C). Vous disposez des scripts pour la création de la base de données. Il s'agit de la création des tables, des déclencheurs et finalement les insertions des données.

Vous avez également un projet Eclipse qui contient le code Java qui est la base à utiliser pour réaliser votre travail. Ce projet contient une classe nommée **Laboratoire4Menu.java** qu'il faut compléter. Concrètement, il y a une méthode à compléter pour chaque question de l'énoncé.

Cette classe contient également, la méthode **main()** pour le l'exécution de votre code. Elle affiche un menu pour le choix des opérations.

En plus de cet énoncé, voici la liste des fichiers qui vous sont fournis :

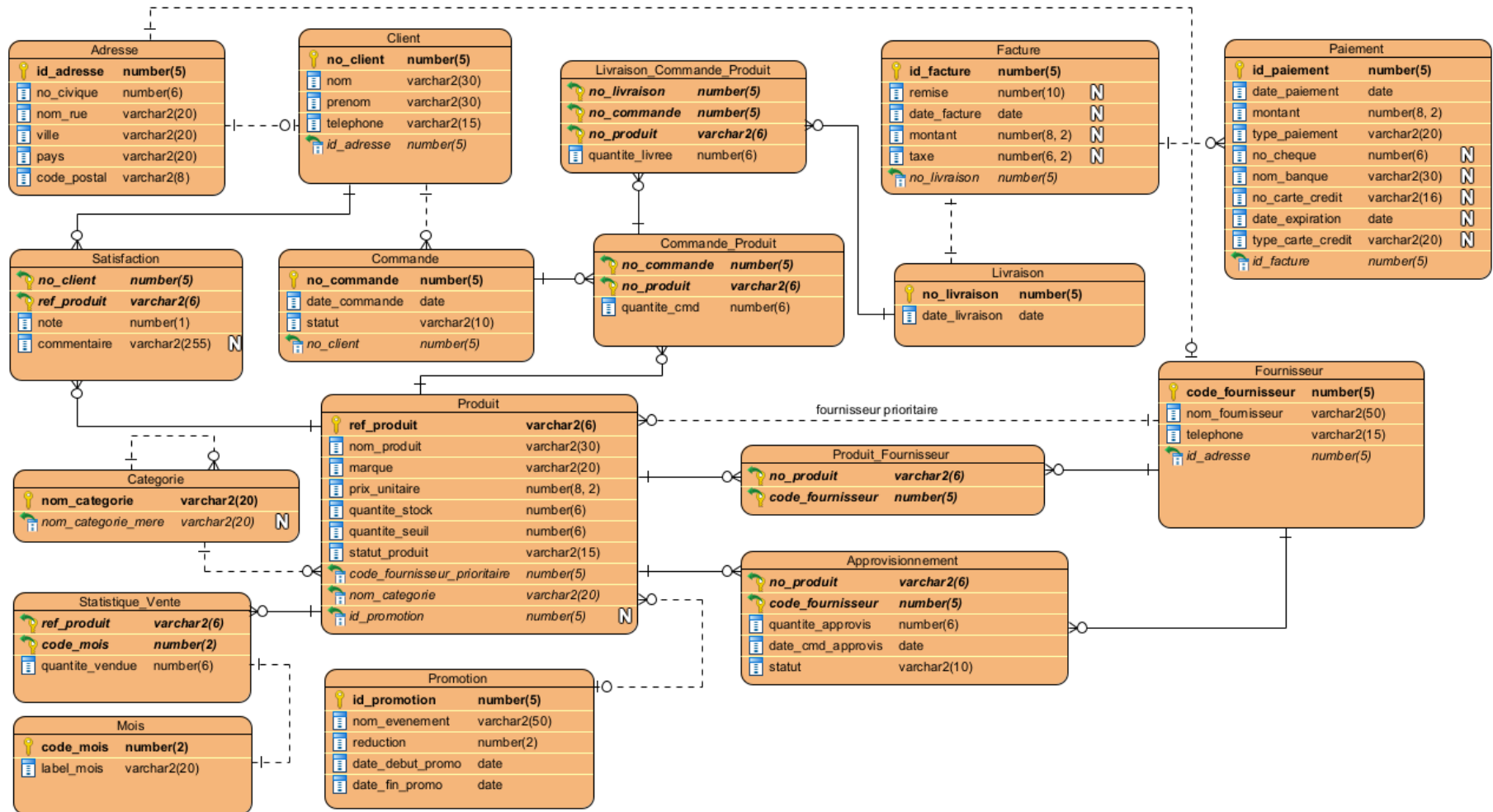
- 1) Script de création des tables : TCH055-H2025-Lab4-CreationBD_VF1.sql
- 2) Script de création de déclencheurs : TCH055-H2025-Lab4-Triggers_VF1.sql
- 3) Script d'initialisation des tables : TCH055-H2025-Lab4-INSERT_VF1.sql
- 4) La classe Laboratoire4Menu.java qu'il faut compléter
- 5) Le projet Eclipse qui contient la classe Laboratoire4Menu.java. Vous pouvez l'importer directement dans Eclipse. C'est l'option recommandée si vous allez utiliser Eclipse comme IDE.

Vous allez donc soit prendre la classe Laboratoire4Menu.java et l'inclure dans votre projet soit utiliser le projet Eclipse.

Note : Il faut exécuter les scripts 1), 2) et 3) dans cet ordre.

Ce laboratoire doit être réalisé en groupes, en préservant les mêmes groupes que les précédents laboratoires (à moins qu'un changement ait été autorisé par votre enseignant-e).

2. Modèle relationnel des données



3. Travail demandé

Indication

Dans la classe Laboratoire4Menu.java qui vous est fournie, la méthode main() fait appel à la méthode afficheMenu(), traite le choix de l'utilisateur pour faire appel aux méthodes que vous allez implémenter. Vous n'avez pas besoin de modifier la méthode main().

La méthode main() affiche le menu suivant :

```
0. Quitter le programme
1. Lister les produits
2. Ajouter un produit
3. Afficher une commande
4. Afficher le montant payé d'une facture
5. Enregistrer un paiement
6. Enregistrer les évaluations des clients

Votre choix...
```

La suite du document décrira les méthodes qu'il faut implémenter. À la fin de l'exécution, les méthodes doivent afficher le message « Appuyer sur ENTER pour continuer... » pour que l'utilisateur appuie sur ENTER avant le retour à la méthode main() pour le réaffichage du menu. Voir le cas particulier de la méthode qui calcule le (total) montant payé pour une facture.

Note : pour certaines questions, l'utilisateur aura besoin d'introduire des données en utilisant la console.

IMPORTANT :

- Au lancement de la méthode main(), cette dernière va établir la connexion en appelant la méthode **connexionBDD()** avant d'afficher le menu. Il faut donc commencer par implémenter cette méthode (Question A) pour voir le menu s'afficher.
- Penser également à mettre les informations de votre compte (nom utilisateur et mot de passe Oracle) dans le code, au début de la méthode main().

Question A : Connexion / déconnexion

Implémenter les méthodes **connexionBDD()** et **fermetureConnexion()** pour la connexion et la déconnexion du serveur de base de données. Elles affichent un message en cas d'échec.

Question B : Option 1 du menu

Implémenter la méthode **listerProduits()** affichant la liste des produits par ordre alphabétique de la référence du produit, comme le montre l'affichage suivant :

Référence	NOM	MARQUE	Prix Unitaire	Quantité	Seuil	Statut	Code fournisseur
DD2001	IO-IDE	IOMEGA	399.0	23	12	ENVENTE	52
DD2002	IO-SSD	IOMEGA	890.0	23	10	ENVENTE	53
DD2003	IO-SSD-S2	IOMEGA	1100.0	5	5	ENVENTE	54
LT2010	Ace Next	ACER	999.0	16	6	ENVENTE	51
LT2011	Prolite	HP	2100.0	20	8	ENVENTE	53
PC2000	Inspiron-5	DELL	1320.0	34	20	ENVENTE	50
PC2001	Elite-X32	HP	1430.0	23	5	ENVENTE	51
SC2001	VS-5433	viewsonic	475.0	35	15	ENVENTE	50
SC2002	Wide 390	DELL	675.0	8	8	ENVENTE	52
SF3001	Windows 11	Microsoft	110.0	200	20	ENVENTE	52
SF3002	Windows 10	Microsoft	110.0	200	20	ENVENTE	52

Appuyer sur ENTER pour continuer...

Question C : Option 2 du menu

Implémenter la méthode **ajouterProduit()** permettant d'ajouter un nouveau produit dans la BD. Cette méthode doit demander à l'utilisateur d'introduire les informations du produit à la console, puis afficher un message de succès si l'insertion a réussi, ou un message d'échec dans le cas contraire.

Indications : vous pouvez utiliser l'objet Scanner pour introduire une donnée à la console :

```
Scanner sc = new Scanner(System.in);
String data = sc.nextLine();
```

Le code suivant permet de transformer data en un nombre, si besoin :

```
// Transformation en un entier
int q stock = Integer.parseInt(data) ;
```

Ou

```
// Transformation en un réel
```

```
float prix = Float.parseFloat(data) ;
```

L'exécution de l'option 2, peut ressembler à l'affichage suivant :

```
Votre choix...
2
Veuillez saisir le numéro de référence : PC3001
Veuillez saisir le nom du produit : Ordinateur
Veuillez saisir la marque : HP
Veuillez saisir le prix unitaire : 1450.99
Veuillez saisir la quantité en stock : 30
Veuillez saisir la quantité seuil : 10
Veuillez saisir le code fournisseur prioritaire : 50
Produit ajouté avec succès
Appuyer sur ENTER pour continuer...
```

Question D : Option 3 du menu

Implémenter la méthode permettant d'afficher une commande dont le numéro est donné comme paramètre : la méthode **afficherCommande(int numCommande)**

Cette méthode affiche le nom, le prénom et le numéro de téléphone du client, le numéro, la date et le statut de la commande, puis la liste des items commandés et finalement le montant total de cette commande. Chaque élément de la liste (des items commandés) comporte le numéro du produit, son nom la marque, le prix unitaire et la quantité commandée, la quantité en stock et le total partiel (total de l'item).

Note : La méthode doit afficher un message si la commande n'existe pas.

Exemple d'affichage :

```
Client      : michel tremblay
Téléphone   : 514 123 4578
No Commande : 30
Date        : 2023-01-26
Statut      : ENCOURS
```

Ref	Produit	Nom	Marque	Prix	Q.Commandée	Q.Stock	T.Partial
DD2002	IO-SSD		IOMEGA	890,00	8	23	7120,00
PC2000	Inspiron-5		DELL	1320,00	4	34	5280,00
SC2001	VS-5433		viewsonic	475,00	3	35	1425,00

```
Total commande : 13825,00 $
Appuyer sur ENTER pour continuer...
```

Question E : Option 4 du menu

Implémenter la méthode qui affiche le montant total des paiements effectués pour une facture. Cette méthode prend un numéro de facture comme paramètre et **retourne** le montant calculé des paiements : `calculerPaiements(int numFacture , boolean affichage)`

Le paramètre `affichage` indique si la méthode peut afficher des messages (`affichage=true`) ou non (`affichage=false`). En conséquence, cette méthode peut être appelée par une autre méthode sans qu'elle affiche un message ou qu'elle soit en attente d'appuyer sur Enter. Elle retourne tout simplement un résultat.

Note : la méthode affiche un message (à condition que le paramètre `affichage` soit `true`) si la facture n'existe pas et retourne -1.

Question F : Option 5 du menu

Implémenter la méthode `enregistrerPaiement(int numFacture)` pour ajouter un nouveau paiement pour une facture. Cette méthode prend le numéro de facture comme paramètre.

Notez que le paiement peut être effectué CASH, par chèque ou par carte de crédit. L'utilisateur doit introduire à la console toutes les données nécessaires.

Attention : le paiement doit être rejeté si le total des paiements (les paiements déjà réalisés et celui que vous allez ajouter) dépasse le montant de la facture incluant les taxes. Utiliser la méthode de la question E pour déterminer les paiements déjà réalisés.

Note : la méthode affiche un message si la facture n'existe pas.

Question G : Option 6 du menu

Implémenter la méthode :

`enregistreEvaluation(SatisfactionData[] listEvaluation)`

pour enregistrer dans la BD, les évaluations des produits données comme paramètre. En effet, cette méthode prend comme paramètre un tableau d'objets de la classe `SatisfactionData`. Il faut utiliser les données de ce tableau pour les insertions dans la BD.

Indication : la méthode `listSatisfactionData()` crée et retourne le tableau d'évaluation qui sera passé comme paramètre à l'appel de la méthode `enregistreEvaluation`.

Attention : pour cette méthode, il est impératif de faire les insertions par lot.

La méthode doit afficher le nombre d'insertion réalisées avec succès, sinon un message d'échec.

Livrable et remise finale

Vous devez remettre un fichier Zip qui contient la classe **Laboratoire4Menu.java** complétée. Il est impératif que le nom du package de votre classe soit :

`ca.ets.tch055_H25.laboratoire4`

Attention : Vous devez porter une attention à la qualité de votre code (indentation, commentaires, etc.).

Date de remise : au plus tard le **jeudi 17 avril 2025 à 23h59**. En respect des règles du Service des Enseignements Généraux, toute remise tardive se verra attribuer la note de 0 (à moins d'entente justifiée et préalable avec l'enseignant-e).

Bon travail ☺