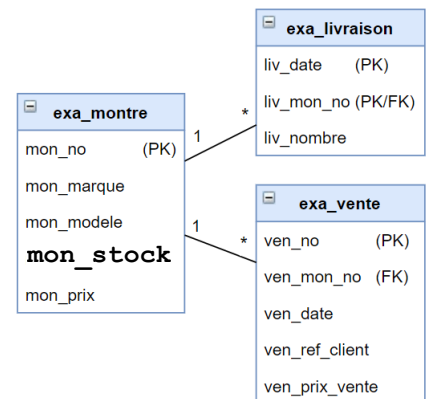


Énoncé :

Une boutique de vente de montres de luxe vous demande de finaliser ses bases de données. Elle possède 3 tables dans Oracle qui gèrent son stock : `exa_montre`, `exa_livraison` et `exa_vente`.

Il faut tenir à jour le stock de montres à chaque nouvelle livraison et vente, et pouvoir afficher des statistiques de ses ventes.

Explication des identifiants : les ventes sont numérotées (`ven_no`) à l'aide d'une séquence (`sq_exa_vente_no`), alors que la clé primaire des livraisons est constituée de la date de livraison et du numéro de la montre.



Fonctionnalités souhaitées, travail demandé :

a) Gestion des livraisons et mise-à-jour du stock

- Différentes applications permettent d'enregistrer les livraisons dans le système. Afin d'éviter toute incohérence, malversation ou fausse manipulation, cette table est protégée, et n'est donc pas accessible directement depuis les applications. Il est obligatoire de passer par la vue `vw_exa_livraison` qui a déjà été créée dans ce but.

vw_exa_livraison				
	Date	Marque	Modele	Nombre
1	03/01/2022	Cartier	Secret	1
2	03/01/2022	Piaget	Limelight	1
3	03/01/2022	Rolex	Daytona	3
4	10/01/2022	cartier	Rotonde	1
5	10/01/2022	Patek Philippe	Supercomplication	1
6	17/01/2022	Cartier	BallonBlanc	1
7	17/01/2022	Cartier	Secret	1
8	17/01/2022	Piaget	Emperador	2
9	17/01/2022	Rolex	Daytona	5

- Les insertions se feront donc toujours à travers cette vue `vw_exa_livraison`. Il faut fournir la date de livraison, la marque et le modèle de montre, ainsi que le nombre livré.

Exemple d'insertion d'une livraison (par exemple dans l'application Java) :

```
INSERT INTO vw_exa_livraison VALUES('19.01.2022', 'Cartier', 'Secret', 2);
```

- Cette instruction `INSERT` va donc insérer cette livraison dans la table, et ajouter le nombre (2) au stock de montre dans le champ `exa_montre.mon_stock`.
- Dans les rares cas où il y aurait déjà eu une autre livraison de cette montre le même jour, on modifiera le champ `liv_nombre` (on rajoutera 2) au lieu d'insérer une nouvelle livraison.
- Dans le cas où la montre livrée n'existe pas encore (on a reçu un nouveau modèle), il faut ajouter cette nouvelle montre dans la table `exa_montre`.
- On ne doit pas pouvoir modifier ou supprimer une livraison !

b) Gestion des ventes : validation de la vente, et mise-à-jour du stock

- Ex : `INSERT INTO exa_vente VALUES (NULL, 3, '19.01.2022', 'ChSt', 750);`
- Lors de l'insertion d'une nouvelle vente, la BDD doit vérifier les informations suivantes :
 - la date de vente doit être inférieure ou égale à la date du jour
 - la montre vendue doit être en stock (`mon_stock > 0`)
 - le numéro de vente (`ven_no`) n'a pas besoin d'être fourni (ou sera ignoré), on utilisera la séquence `sq_exa_vente_no` qui a été créée dans ce but.
- En cas d'erreur, la BDD retournera un message explicatif qui sera affiché dans l'application Java :
 - Soit « `Date de vente incorrecte !` »
 - Soit « `Aucune Piaget Limelight en stock!` » (en indiquant la marque & le modèle)
- Le stock doit être automatiquement décrémenté (diminué) de 1.

c) Afficher les statistiques des ventes

- Créez une procédure qui affiche (dans le log) les statistiques des ventes pour une certaine marque. Cette procédure recevra donc obligatoirement le nom de la marque en paramètre, et peut éventuellement recevoir un attribut booléen indiquant qu'on souhaite afficher les modèles triés par prix décroissant (par défaut, les modèles doivent être triés par ordre alphabétique).
- Données à afficher pour chaque modèle (de la marque spécifiée en paramètre) :
 - le nombre de vente
 - le prix moyen des ventes (les prix sont exprimés en milliers)
 - les dates des ventes (ou « aucune vente » le cas échéant)
 - le message « Aucune montre de marque xxxx » si on a fourni une marque inconnue en paramètre.
- Résultats affichés si on a appelé 3x la procédure avec les paramètres suivants :
 (`'Cartier'`) / (`'Cartier', TRUE`) / (`'Swatch', FALSE`)

```

Statistiques des ventes Cartier
- BallonBlanc : 2 ventes au prix moyen de 32 K Frs aux dates suivantes : 06/01/22 13/01/22
- Panthère : aucune vente
- Rotonde : 3 ventes au prix moyen de 64 K Frs aux dates suivantes : 10/01/22 11/01/22 14/01/22
- Secret : 2 ventes au prix moyen de 113 K Frs aux dates suivantes : 04/01/22 18/01/22

Statistiques des ventes Cartier
- Secret : 2 ventes au prix moyen de 113 K Frs aux dates suivantes : 04/01/22 18/01/22
- Rotonde : 3 ventes au prix moyen de 64 K Frs aux dates suivantes : 10/01/22 11/01/22 14/01/22
- Panthère : aucune vente
- BallonBlanc : 2 ventes au prix moyen de 32 K Frs aux dates suivantes : 06/01/22 13/01/22

Aucune montre de marque Swatch !

```

d) Test des fonctionnalités depuis l'application Java, et mise-à-jour des bdd NoSQL

- Les tests doivent être lancés depuis l'application Java ExaMontres. Cette application se connecte à votre BDD Oracle, puis doit exécuter les `INSERT` d'exemples fournis.
- Dans le cas où une insertion ne s'exécute pas correctement, le programme ne doit pas se planter, mais afficher le message d'erreur retourné par la bdd (cf. point b ci-dessus).
- Si la vente est bien enregistrée dans Oracle, on peut alors effectuer les 2 mises-à-jour suivantes :

Mise-à-jour des bdd NoSQL :

- Remarque : il n'est pas nécessaire d'installer les bases de données MongoDB et Neo4j ! Vous n'avez pas besoin de lancer les tests, mais uniquement écrire les instructions permettant de se connecter et de mettre à jour les données dans l'une ou l'autre de ces bdd NoSQL.
- La bdd Neo4j permet de visualiser tous nos clients ayant acheté une montre : insérez donc un nouveau nœud contenant le client (on ne connaît que sa référence), et créez un lien avec la montre qu'il a achetée ; celle-ci existe déjà forcément dans la BDD, dans un nœud dont le label est Montre.
- La bdd MongoDD contient une collection « Montre » qui répertorie toutes les informations de chaque montre. Ajoutez donc à la montre vendue (dont vous connaissez la marque et le modèle) une information supplémentaire avec la référence de l'acheteur et le prix payé.