

ENSTA IN207 - Examen de Bases de Données 2019

Durée : 2h. Tous documents autorisés.

Il faut répondre sur la copie. En cas d'erreur, ajoutez des feuilles en indiquant votre nom
Les questions sont généralement indépendantes et notées sur 1 point (sauf indication contraire)

Nom :	Prénom :
--------------	-----------------

La base ci-dessous permet de gérer des acheteurs et des vendeurs d'"objets". Les tables ont été réduites à leur strict minimum. Il n'y a pas de valeurs nulles, Il n'y a pas d'homonymes (nulle part). La casse (majuscules/minuscules) n'a pas d'importance. **Utilisez les nom abrégés (en gras : ACH, REC, TYP, PRO, VEN).**

ACH eteur	(NAch, NomAch, Ville,...)	Table des acheteurs...
RE cherche	(NAch, NTyp)	Quel acheteur recherche quel type d'objet ...
TY PeObjet	(NTyp, NomTyp, PrixMoy,...)	Types d'objets recherchés (avec prix moyen)
PRO position	(NPro, NVen, NTyp, Prix, Desc, ...)	Propositions de ventes (avec prix)
VEN deur	(NVen, NomVen, Ville,...)	Vendeurs d'objets

Un **ACH**eteur **RE**cherche un ou plusieurs **TY**Pe d'objets (par exemple, "Lit"). Un prix moyen constaté est associé à un type d'objet (par exemple 100 euros pour un lit). D'un autre côté, un **VEN**deur fait des **PRO**positions de vente d'objets d'un certain **TY**Pe, avec un prix et une description de l'état de l'objet (par exemple, "Lit" à 150 euros dont la description est "Lit 2 places en état moyen, de marque IKEA,". Chaque objet proposé correspond à un seul **TY**Pe d'objet.

Partie 1 : Conception (3 points)

Q1 (2,5 points) : Dessiner le MCD initial duquel on a déduit ce MLD. Justifier en particulier si REC et PRO sont des entités ou des associations et pourquoi. Penser à indiquer les cardinalités.

Q2 (0.5 point) : Dessiner le MLD en indiquant les clé primaires (souligné plein), les clés étrangères (souligné tiret) et en indiquant par une flèche, les liens entre les tables (des clés étrangères vers les clés primaires)

ACH eteur	(N ACH , Nom ACH , Ville,...)	Table des acheteurs...
RE cherche	(N ACH , N Typ)	Quel acheteur recherche quel type d'objet ...
TYPE Objet	(N Typ , Nom Typ , Prix Moy ,...)	Types d'objets recherchés (avec prix moyen)
PRO position	(N Pro , N Ven , N Typ , Prix, Desc, ...)	Propositions de ventes (avec prix)
VEN deur	(N Ven , Nom Ven , Ville,...)	Vendeurs d'objets

Partie 2 : Algèbre (4 points) : Exprimer en algèbre les questions suivantes (1 pt/question)

Q3 : Nom des acheteurs qui recherchent un objet de type « lit » ?

Q4 : Nom des acheteurs qui recherchent un objet de type « lit » et un objet de type « Armoire »?

Q5 : Nom des acheteurs qui ne recherchent que des objets de prix moyen inférieurs à 10 euros ?

Q6 : Ville où il y a des vendeurs mais aucun acheteur ?

Partie 3 : SQL (7 points) : Répondre aux questions suivantes en SQL (1 pt/question)

Q7 : Donner l'instruction SQL pour créer la table **PRO**position (avec les clés primaires et les contraintes d'intégrité référentielles)

Q8 : Indiquer l'impact de la suppression d'un type d'objet, en supposant que les tables ont été créés avec les contraintes d'intégrité référentielles dans le cas (1) avec cascade delete et (2) sans cascade delete.

ACH eteur	(N A ch, Nom A ch, Ville,...)	Table des acheteurs...
RE cherche	(N A ch, N T yp)	Quel acheteur recherche quel type d'objet ...
TYPE Objet	(N T yp, Nom T yp, Prix M oy,...)	Types d'objets recherchés (avec prix moyen)
PRO position	(N P ro, N V en, N T yp, Prix, Desc, ...)	Propositions de ventes (avec prix)
VEN deur	(N V en, Nom V en, Ville,...)	Vendeurs d'objets

Q9 : Nom des vendeurs proposant des objets de prix inférieur à 10 euros

Q10 : Nom des vendeurs ne proposant aucun objet

Q11 : Nom des vendeurs proposant des objets de prix **moyen** inférieur à 10 euros et des objets de prix **moyen** supérieur à 100 euros (Attention le prix moyen se trouve dans la table **TYPE**Objet)

Q12 : Nombre de proposition par ville (de vendeur)

Q13 : Nom du vendeur proposant l'objet le moins cher

Partie 4 : Transactions (3 points)

Trois types de transactions s'exécutent en parallèle sur la base des acheteurs et vendeurs d'objets. Les transactions **de type INS** ne font que des insertions dans la table **PROposition** (une transaction = une seule insertion), celles **de type STAT** calculent des statistiques globales (ex: Nombre de proposition par ville (de vendeur)) et celles **de type QUERY** font des recherches pour déterminer de façon stable (i.e., isolée des effets des autres transactions) la liste des propositions d'objets à la vente dans une certaine ville. Attention, plusieurs transactions de chaque type peuvent être réalisées de manière concurrente. Répondre aux 3 questions suivantes sur la base de ces hypothèses, en justifiant vos réponses.

Q14 : Quel est le degré d'isolation le mieux adapté à chaque type de transaction ?

INS : ☐ Read Uncommitted – ☐ Read Committed – ☐ Repeatable Read - ☐ Serializable

Justification :

STAT : ☐ Read Uncommitted – ☐ Read Committed – ☐ Repeatable Read - ☐ Serializable

Justification :

QUERY : ☐ Read Uncommitted – ☐ Read Committed – ☐ Repeatable Read - ☐ Serializable

Justification :

Q15 : Préciser les situations de blocage qui peuvent se produire entre ces transactions en supposant que le SGBD n'a pas de mécanisme de multiversion et en supposant les degrés de la Q14

INS et INS : ☐ Pas de blocage – ☐ Blocage

Justification :

INS et STAT : ☐ Pas de blocage – ☐ Blocage

Justification :

INS et QUERY : ☐ Pas de blocage – ☐ Blocage

Justification :

STAT et STAT : ☐ Pas de blocage – ☐ Blocage

Justification :

STAT et QUERY : ☐ Pas de blocage – ☐ Blocage

Justification :

QUERY et QUERY : ☐ Pas de blocage – ☐ Blocage

Justification :

Q16 : Idem dans le cas d'un SGBD ayant un protocole multiversion. (ne pas justifier lorsque c'est idem à la Q15)

INS et INS : ☐ Pas de blocage – ☐ Blocage

Justification :

INS et STAT : ☐ Pas de blocage – ☐ Blocage

Justification :

INS et QUERY : ☐ Pas de blocage – ☐ Blocage

Justification :

STAT et STAT : ☐ Pas de blocage – ☐ Blocage

Justification :

STAT et QUERY : ☐ Pas de blocage – ☐ Blocage

Justification :

QUERY et QUERY : ☐ Pas de blocage – ☐ Blocage

Justification :

Nom :

Prénom :

Partie 5 : (3 points) Programmation base de données. Soit la base suivante :

```
VENDEURS (NUMVEN, NOM, SALAIRE, VILLE)
VENTES (NUMVT, NUMVEN, DATE_COM)
LIGNES_VEN (NUMVT, NUMPROD, FOURNISSEUR, QUANTITÉ)
PRODUITS (NUMPROD, NOMPROD, TYPE, COULEUR, PRIX)
LIVRAISON (NUMVT, DATE, STATUT)
```

Q17 : Donner le contenu de la table TEMP après l'exécution suivante : (1 point)

```
CREATE TABLE TEMP (ID NUMBER, SAL NUMBER);
INSERT INTO VENDEURS VALUES (1, 'SCOTT', 1000, 'PARIS');
INSERT INTO VENDEURS VALUES (2, 'BOB', 2000, 'PARIS');
INSERT INTO VENDEURS VALUES (3, 'JOHN', 3000, 'PARIS');
INSERT INTO VENDEURS VALUES (4, 'MAC', 4000, 'PARIS');
INSERT INTO VENDEURS VALUES (5, 'TED', 5000, 'PARIS');
DECLARE
    i NUMBER:=1;
BEGIN
    FOR t IN (SELECT NUMVEN, SALAIRE FROM VENDEURS ORDER BY NOM ASC) LOOP
        EXIT WHEN i>3;
        INSERT INTO TEMP VALUES (t.NUMVEN, t.SALAIRE);
        i:=i+1;
    END LOOP;
END;/
```

Q18 : Indiquer ce que fait le programme ci-dessous. Proposer une requête SQL produisant le même résultat que celui affiché par ce programme (2 points).

```
CREATE TABLE RESULTAT (numven NUMBER, nb_prod_vendus NUMBER);
DECLARE
    i NUMBER;
BEGIN
    FOR vend IN (SELECT * from VENDEURS) LOOP
        i := 0;
        FOR vt IN (SELECT * from VENTES) LOOP
            FOR lig IN (SELECT * from LIGNES_VEN) LOOP
                IF vend.numven = vt.numven AND vt.numvt = lig.numvt THEN
                    i := i+lig.qté;
                END IF;
            END LOOP;
        END LOOP;
        INSERT INTO RESULTAT VALUES (vend.numven, i);
    END LOOP;
END;/
SELECT * FROM RESULTAT ;
```