

Contrôle SQL

SQL

Soient les tables Candidat, Matiere, Note qui contiennent les données du Bacc

Candidat (numero number(5), nom varchar2(30), prenom varchar2(30))

Matiere (code_matiere number, designation varchar2(30))

Note (#numero number(5), #code_matiere number, note number)

numero clé primaire de la table Candidat

code_matiere clé primaire de la table Matiere

(*numero*, *code_matiere*) clé primaire de la table Note

numero, *code_matiere* clés étrangères de la table Note

1. Ecrire la commande SQL qui affiche la note du candidat Rakoto en Histoire

<u>Nom</u>	<u>Matiere</u>	<u>Note</u>
Rakoto	Histoire	10,5

```
SELECT nom, designation, note FROM candidat JOIN note USING (numero) JOIN matiere  
USING (code_matiere) WHERE nom='Rakoto' AND designation+= 'Histoire';
```

2. Ecrire la commande SQL qui donne le nombre de matières

```
SELECT COUNT(*) FROM matiere ;
```

3. Créer un vue nommée *vue_resultat* qui donne la moyenne générale de tous les candidats.

Select * from *vue_resultat* devra afficher :

<u>Nom</u>	<u>Moyenne</u>
Rabe	8,5
Rakoto	11,5
Rasoa	12

```
CREATE VIEW vue_resultat AS SELECT nom, AVG(note) Moyenne FROM candidat JOIN note  
USING (numero) JOIN matiere USING (code_matiere) GROUP BY nom
```

4. Ecrire la commande SQL qui copie les données de la table Candidat dans la table nommée *Candidat_Copie*. Cette dernière a même structure que la table Candidat

```
INSERT INTO candidat_copie SELECT * FROM candidat ;
```

5. En utilisant la vue *vue_resultat*, écrire la commande SQL qui affiche « Non admis » si la moyenne est inférieure à 10 et « Admis » si la moyenne est supérieure ou égale à 10

<u>Nom</u>	<u>Moyenne</u>	<u>Resultat</u>
Rabe	8,5	Non admis
Rakoto	11,5	Admis
Rasoa	12	Admis

```
SELECT nom, moyenne, CASE WHEN moyenne >= 10 THEN 'Admis' ELSE 'Non admis' END as  
Resultat FROM vue_resultat ;
```

6. Ecrire la commande SQL qui remplace la note en Malagasy du candidat Rakoto par 14 dans la table Note

```
UPDATE note SET note=14 WHERE numero =(SELECT numero FROM candidat WHERE  
nom='Rakoto ') AND code_matiere=(SELECT code_matiere FROM matiere WHERE  
designation='Malagasy ');
```

QCM

1. La table EMP contient les colonnes :

LAST_NAME VARCHAR2(25)

SALARY NUMBER(6,2)

DEPARTMENT_ID NUMBER(6)

Vous voudriez sélectionner les employées qui n'ont pas été affectés à un département

Que peut-on dire de l'instruction

```
SELECT LAST_NAME, SALARY, DEPARTMENT_ID
```

```
FROM EMP
```

```
WHERE DEPARTMENT_ID = NULL
```

- A. L'instruction affiche le résultat souhaité
- B. La colonne dans la clause WHERE devrait être modifiée pour afficher le résultat souhaité
- C. L'opérateur dans la clause WHERE devrait être modifiée pour afficher le résultat souhaité
- D. La clause WHERE devrait être modifiée et on devrait une jointure externe (un outer join) pour produire le résultat souhaité

C

2. Quelle est l'instruction qui définit la colonne DEPTNO comme étant la clé étrangère

- A. CREATE TABLE emp (empno NUMBER(4), ename VARCHAR2(35), deptno NUMBER(7,2) NOT NULL CONSTRAINT emp_deptno_fk FOREIGN KEY deptno REFERENCES dept(deptno))
- B. CREATE TABLE emp (empno NUMBER(4), ename VARCHAR2(35), deptno NUMBER(7,2) NOT NULL CONSTRAINT emp_deptno_fk FOREIGN KEY deptno REFERENCES dept(deptno))
- C. CREATE TABLE emp (empno NUMBER(4), ename VARCHAR2(35), deptno NUMBER(7,2) NOT NULL, CONSTRAINT emp_deptno_fk REFERENCES dept(deptno) FOREIGN KEY (deptno))
- D. CREATE TABLE emp (empno NUMBER(4), ename VARCHAR2(35), deptno NUMBER(7,2) FOREIGN KEY CONSTRAINT emp_deptno_fk deptno REFERENCES dept(deptno))
- E. CREATE TABLE emp (empno NUMBER(4), ename VARCHAR2(35), deptno NUMBER(7,2) NOT NULL, CONSTRAINT emp_deptno_fk FOREIGN KEY (deptno) REFERENCES dept(deptno))

E