

TP-1 – Création et manipulation de bases de données Cas Emploi

Objectif :

*** Révisions et approfondissements du langage SQL sous Oracle**

1. Conception de Bases de Données (MCD – MLD)
2. Création de Bases de Données : Créer les tables, les vues, les index
Commandes SQL : CREATE TABLE ...; CREATE OR REPLACE VIEW ...
3. Mises à jour des données dans la base
Commandes SQL : INSERT INTO ... ; UPDATE ... ; DELETE FROM
4. Interrogations d'une Base de Données
Commandes SQL :
SELECT ... FROM ... WHERE ... GROUP BY ... HAVING ... ORDER BY ...
5. Notion de transaction
Commandes SQL : COMMIT ... ; ROLLBACK ...
6. Fonctions SQL très utiles telles que :
ROUND, TRUNC, RPAD, LPAD, SUBSTR, TO_CHAR ...
7. Quelques commandes très utiles sous SQL-PLUS telles que :
ALTER SESSION ; SET PAGES ; COLUMN ...

I – Le Modèle Conceptuel de Données (MCD) :

A partir du modèle logique de données relationnel (MLD) suivant, proposer le modèle conceptuel de données (MCD) équivalent :

EMP(NUMEMP, NOM, FONCTION, RESPONSABLE, DATE_ENTREE, SALAIRE, COMMISSION, NUMDEPT)
DEPT(NUMDEPT, NOM, VILLE)

Remarques :

- dans la relation EMP, NumEmp est l'identifiant de l'employé, Fonction est la fonction qu'il exerce actuellement, Responsable est le numéro de son responsable hiérarchique (qui est lui-même un employé), NumDept est le numéro du département dans lequel il travaille,
- dans DEPT, NumDept est le numéro de département, Nom son nom et Ville la ville où il est situé.
- Les clés étrangères ne sont pas signalées, à vous de les déduire

II – Le Modèle Logique de Données relationnel (MLD) :

Le script SQL de création de la base de données emploi est ajouté en annexe de ce document.

L'insertion des données se fera comme suit :

```
INSERT INTO matable (ListeAttributs) VALUES (ListeValeurs) ;
```

Pour valider les travaux en cours : `COMMIT ;`

Eventuellement pour changer le format d'affichage de la date par exemple :

```
ALTER SESSION SET NLS_DATE_FORMAT = 'DD-MM-YYYY' ;
```

2.1. Lancez l'interface SQL d'oracle.

Sous Linux exécuter sous Terminal d'abord la commande : `. /opt/oracle/oracle.env`

Puis : `sqlplus login/login@sgbd` (En remplaçant login par votre identifiant).

Le nom de la base de données est **sgbd**

Votre compte Oracle a pour identifiant votre login Linux/Windows et pour mot de passe ce même login.

On vous conseille de changer votre mot de passe une fois connecté à Oracle par la commande :

```
ALTER USER monLogin IDENTIFIED BY monNouveauMotDePasse ;
```

2.2. Lancer le script SQL qui permet de créer les tables et les données de la base de données « Emploi »

- créer le fichier texte qui contiendra votre script sql à l'aide d'un éditeur de texte quelconque à partir de l'annexe.
- Le fichier doit avoir obligatoirement l'extension sql.
- Rajouter dans le script les commandes de création des tables EMP et DEPT sans les contraintes de clés primaires et de clés étrangères qui seront traitées dans la partie III
- exécuter le script par la commande SQLPLUS **start** monscript.sql ou bien @monscript.sql

2.3. Testez les commandes ci-dessous :

⇒⇒ A la découverte du dictionnaire d'Oracle

```
show user ;
select * from all_users;
pause
desc user_tables ;
select table_name from user_tables;
pause
column table_name heading Nom_de_l_utilisateur
select table_name from user_tables;
pause
column object_name format a30
set pages 30
select object_name, object_type from user_objects order by object_type;
pause
desc user_constraints;
column constraint_name format a30
select constraint_name from user_constraints where table_name='&matable';
```

⇒⇒ A la redécouverte de quelques fonctions SQL/Oracle

```
SELECT RPAD('Soleil',17,'bla') "RPAD exemple" FROM DUAL;
SELECT LPAD('DESS EID',15,'*.') "LPAD exemple" FROM DUAL;
SELECT SUBSTR('DESS EID',6,3) "SUBSTR exemple" FROM DUAL;
SELECT SUBSTR('ABCDEFGHIJ',-5,4) "SUBSTR exemple" FROM DUAL;
SELECT TO_CHAR (SYSDATE, 'MM-DD-YYYY HH24:MI:SS') "Now" FROM DUAL;
SELECT LENGTH('WEB WAREHOUSE') "Longueur en caractères" FROM DUAL;
SELECT DECODE(ANNEE, 1, 'Première', 2, 'Seconde', 'Valeur différente de 1 et de 2 !!') AS ANETUDE FROM ELEVES;
SELECT NVL(SPECIALITE, 'Valeur NON renseignée') FROM PROFESSEURS;
SELECT NVL(SPECIALITE, 'Valeur NON renseignée') AS SPEC_PROF FROM PROFESSEURS;
SELECT ROUND(17.0958,1) "ROUND exemple" FROM DUAL;
SELECT ROUND(17.58,2) "ROUND exemple" FROM DUAL;
SELECT TRUNC(1958.0917,1) "TRUNC exemple" FROM DUAL;
SELECT TRUNC(1958.0917,2) "TRUNC exemple" FROM DUAL;
SELECT ROUND(TO_DATE('17-SEP-2002'), 'YEAR') "New Year" FROM DUAL;
SELECT EXTRACT(YEAR FROM SYSDATE) FROM DUAL;
SELECT ADD_MONTHS(SYSDATE,7) FROM DUAL;
SELECT TRUNC(MONTHS_BETWEEN(SYSDATE, TO_DATE('19-JUN-2001'))) AS AGEB FROM DUAL;
SELECT TO_NUMBER(TO_CHAR(SYSDATE, 'YYYY'))FROM DUAL;
```

III – Les Contraintes d'intégrité :

1. Vérifier la création et le remplissage des tables EMP et DEPT.
2. Visualiser la liste des contraintes en affichant le contenu de **USER_CONSTRAINTS** du dictionnaire de données. Quelles contraintes peut-on remarquer ?
3. Tester l'insertion d'un employé dont on ne connaît que le nom (DUPONT), le salaire (600 €) et le département (30).
4. Modifier le schéma de EMP de manière à ce que l'attribut NUMEMP comporte 5 chiffres au maximum. Peut-on remettre NUMEMP à 4 chiffres ?
5. Modifier la table EMP de manière à ce qu'il ne puisse pas y avoir de salaire inférieur à 800€. Tester la contrainte.
6. Modifier la table EMP de manière à ce que la date d'entrée soit par défaut la date courante. Tester la contrainte.
7. Vérifier, en affichant le contenu de DEPT, si NUMDEPT satisfait la contrainte de clé primaire dans la table DEPT.
8. Insérer un département ayant la même valeur de NUMDEPT qu'un département existant.
9. Tenter d'ajouter une contrainte sur la table DEPT définissant NUMDEPT comme clé primaire.
10. Supprimez le tuple que vous venez d'insérer. Retentez la modification définissant NUMDEPT comme clé primaire de DEPT. Testez cette contrainte en réinsérant à nouveau le tuple.
11. Modifier la table EMP de manière à ce que le cumul du salaire et de la commission soit obligatoirement compris entre 800 et 8000 €. Tester la contrainte.
12. Modifier la table EMP de manière à ce que le numéro de département NUMDEPT définisse une clé étrangère correspondant à NUMDEPT de DEPT. Tester la contrainte.
13. Modifier la table EMP de manière à ce que le numéro d'employé soit clé primaire. Tester la contrainte.
14. Tenter de définir la contrainte que COMMISSION ne puisse pas avoir de valeur nulle.
15. Tenter de supprimer la contrainte de clé primaire sur DEPT (Utiliser la clause **DROP**? Avec et sans la clause **CASCADE**).
16. Y'a-t-il un moyen pour exprimer la contrainte que les départements ont tous des employés ?
17. Pour récapituler, visualisez la liste des contraintes en affichant la vue du dictionnaire de données **USER_CONSTRAINTS**.

IV – Interrogation :

1. Donner la liste de tous les départements
2. Donner la liste des départements sans employés
3. Qui est le responsable de l'entreprise (il s'agit de l'employé sans responsable)
4. Donner la liste des employés du plus grand salaire au plus petit
5. Quel est le département qui emploie le plus d'employés ?
6. Donner la masse salariale par département (total des salaires par département)
7. Quel est le total des commissions par département ?

V – Les vues :

Une vue (appelée souvent table virtuelle) est structurée en lignes et en colonnes comme une relation mais n'occupe pas de place disque. Une vue peut être interrogée exactement comme une table, par contre il existe des restrictions sur les mises à jour autorisées.

1. Créer une table PROJET(NUMP, NOMP) en définissant NUMP comme clé primaire, puis insérer 5 projets numérotés entre 101 et 105.
2. Ajouter une colonne NUMP à la table EMP et une contrainte de clé étrangère sur cette colonne pour référencer la table PROJET. Affecter chaque employé à un projet.
3. Création de vue à partir d'une relation :
 - (a) Créer une vue EMP10 qui permet de donner pour le département numéro 10, les numéros des employés, leurs noms et leurs fonctions.
 - (b) Donner le contenu de la vue.
4. Impact d'une modification d'une relation sur une vue et vice versa :
 - (a) En utilisant la relation EMP, changer la fonction de l'employée HELENE en ANALYSTE.
Vérifier le contenu de EMP10.
 - (b) En utilisant la relation EMP, insérer un nouvel employé (7999, 'BROWN', 'SECRETAIRE', 7902, to_date('25-JAN-80','dd-mon-yy'), 800, NULL, 10, 102).
Vérifier le contenu de EMP10.
 - (c) En utilisant la vue EMP10, donner les noms et les fonctions des employés du département numéro 10.
 - (d) En utilisant la vue EMP10, remettre la fonction de l'employée HELENE à SECRETAIRE.
 - (e) Donner le contenu de la vue EMP10. Que peut-on déduire ?
5. Création des vues par jointure sur les relations :
 - (a) Créer une vue PROJSTAFF contenant les noms des employés, les noms et les numéros des projets sur lesquels ils sont affectés. Afficher le contenu de la vue.
 - (b) Créer une vue PROJECTS contenant les noms des projets, les noms et les numéros des employés, ainsi que la ville de leurs départements. Afficher le contenu de la vue.
 - (c) Que peut-on dire sur l'insertion du tuple ('DURAND','SATELLITE', 110) dans PROJSTAFF.
6. Interrogation des vues :
 - (a) Donner les noms des employés qui sont affectés au projet numéro 101.
 - (b) Donner les noms des projets et les noms des employés travaillant à Paris.
7. Interrogation par jointure de relation et de vue
 - (a) Donner les postes et les projets des employés travaillant à Paris. (Utiliser la vue PROJECTS et la relation EMP)
8. Création de vue à partir d'une autre vue :
 - (a) En utilisant la vue PROJECTS, créer une autre vue PARIS donnant les noms des projets et les noms des employés travaillant à Paris. Lister ensuite le contenu de cette vue.

9. Création de vue avec utilisation de champs calculés :

- (a) Créer une vue PAY contenant les noms des employés, leurs salaires mensuels et leurs salaires annuels ainsi que leur numéro de département.
- (b) Lister ensuite ceux du département 30.
- (c) Peut-on modifier cette vue ?

10. Création de vue avec utilisation des agrégats :

- (a) Créer une vue DEPT_SAL contenant pour chaque département, le minimum, le maximum, la moyenne et le cumul des salaires.

VI – Les index :

Créer un index sur la table EMP sur la colonne nom afin d'améliorer les performances de recherche sur le nom d'employé. Tester le temps d'exécution d'une recherche sur le nom d'employé avant et après la mise en place de l'index, avec les commandes **SET TIMING ON** et **SET TIMING OFF**.

```
CREATE INDEX NDXEMP ON EMP (NOM ASC) ;
```

Annexe : Script de création de la base de données Emploi

```
-- =====
-- Nom de la base      : EMPLOI
-- Nom de SGBD         : ORACLE Version 9i
-- Date de création   : Janvier 2010
-- Auteur              : Abdelkrim LAHLOU - EFREI
-- =====

DROP TABLE EMP;
DROP TABLE DEPT;

-- =====
ALTER SESSION SET NLS_DATE_FORMAT = 'DD-MM-YYYY' ;
-- =====

-- =====
-- Table : EMP
-- =====

-- A vous d'ecrire la commande de creation de la table EMP
-- avec les types conseilles suivants :
--   NumEmp NUMBER(4)
--   Nom VARCHAR2(20) ce champs doit toujours etre renseigne
--   Fonction VARCHAR2(15)
--   Responsable NUMBER(4)
--   DATE_ENTREE DATE
--   Salaire NUMBER(7, 2)
--   Commission NUMBER(7, 2)
--   NumDept NUMBER(2)

-- =====
-- Table : DEPT
-- =====

-- A vous d'ecrire la commande de creation de la table DEPT
-- avec les types conseilles suivants :
--   NumDept NUMBER(2)
--   Nom VARCHAR2(15)
--   Ville VARCHAR2(25)

REM *****
REM Creation des donnees
REM *****

SELECT 'Creation des donnees' FROM DUAL;
SELECT 'Tapez sur RETURN pour continuer' FROM DUAL;
PAUSE

INSERT INTO EMP VALUES (7369, 'SARAH', 'SECRETAIRE', 7902, '17-12-1980', 800,
NULL, 20);
INSERT INTO EMP VALUES (7499, 'PIERRE', 'VENDEUR', 7698, '20-02-1981', 1600,
300, 30);
INSERT INTO EMP VALUES (7521, 'MARIE', 'VENDEUR', 7698, '22-02-1981', 1250,
500, 30);
INSERT INTO EMP VALUES (7566, 'MARC', 'MANAGER', 7839, '02-04-1981', 2975,
NULL, 20);
INSERT INTO EMP VALUES (7654, 'MARTIN', 'VENDEUR', 7698, '28-09-1981', 1250,
1400, 30);
INSERT INTO EMP VALUES (7698, 'JULIE', 'MANAGER', 7839, '01-05-1981', 2850,
NULL, 30);
INSERT INTO EMP VALUES (7782, 'CLARK', 'MANAGER', 7839, '09-06-1981', 2450,
NULL, 10);
```

```

INSERT INTO EMP VALUES (7788, 'SCOTT', 'ANALYSTE', 7566, '09-12-1982', 3000,
NULL, 20);
INSERT INTO EMP VALUES (7839, 'KING', 'PRESIDENT', NULL, '17-11-1981', 5000,
NULL, 10);
INSERT INTO EMP VALUES (7844, 'TURNER', 'VENDEUR', 7698, '08-09-1981', 1500,
0, 30);
INSERT INTO EMP VALUES (7876, 'SAM', 'SECRETAIRE', 7788, '12-01-1983', 1100,
NULL, 20);
INSERT INTO EMP VALUES (7900, 'JAMES', 'SECRETAIRE', 7698, '03-02-1981', 950,
NULL, 30);
INSERT INTO EMP VALUES (7902, 'FORD', 'ANALYSTE', 7566, '03-12-1981', 3000,
NULL, 20);
INSERT INTO EMP VALUES (7934, 'HELENE', 'SECRETAIRE', 7782, '23-01-1982', 1300,
NULL, 10);

INSERT INTO DEPT (NUMDEPT, NOM, VILLE) VALUES (10, 'FINANCES', 'PARIS');
INSERT INTO DEPT (NUMDEPT, NOM, VILLE) VALUES (20, 'DSI', 'NICE');
INSERT INTO DEPT (NUMDEPT, NOM, VILLE) VALUES (30, 'VENTES', 'PARIS');
INSERT INTO DEPT (NUMDEPT, NOM, VILLE) VALUES (40, 'RH', 'LYON');

REM *****
COMMIT;
REM *****

REM *****
REM ***** AFFICHAGE DU CONTENU DE LA BD *****
REM *****

REM ***** LA TABLE EMP *****

DESC EMP;
SELECT 'Tapez sur RETURN pour continuer'FROM DUAL;
PAUSE

COLUMN NOM FORMAT A10
TTITLE 'La liste des employés'

SELECT * FROM EMP ;

SELECT * FROM DEPT ;

```