

## Partie III

4. Créer une tablespace : « DATA » et un utilisateur « Std\_mgr » identifié par un mot de passe : « std\_pass ».

a- Créer deux répertoires « DATA » et « TEMP » dans C:/ ; The taplespace « DATA » sera stockée dans C:/DATA.

```
CREATE SMALLFILE TABLESPACE "DATA"  
DATAFILE 'C:\DATA\DATA.DBF' SIZE 2G  
AUTOEXTEND ON NEXT 100M MAXSIZE 3G  
LOGGING  
ONLINE  
PERMANENT  
BLOCKSIZE 8192  
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 10M  
SEGMENT SPACE MANAGEMENT AUTO;
```

Sortie de script x | Tâche terminée en 21,547 secondes  
smallfile TABLESPACE créé(e).

b- DEFAULT taplespace pour l'utilisateur « Std\_mgr » est : « DATA » et TEMPORARY TABLESPACE est : "TEMP"

```
CREATE USER Std_mgr IDENTIFIED BY "std_pass"  
DEFAULT TABLESPACE "DATA"  
TEMPORARY TABLESPACE "TEMP"  
PROFILE DEFAULT  
QUOTA UNLIMITED ON "DATA" ;
```

Sortie de script x | Tâche terminée en 6,302 secondes  
user STD\_MGR créé(e).

c- Ensuite, vous attribuez les privilèges : « CONNECT » et « RESOURCE » à l'utilisateur « Std\_mgr ».

```
GRANT "CONNECT" TO Std_mgr;  
GRANT "RESOURCE" TO Std_mgr;
```

Sortie de script x | Tâche terminée en 0,003 sec  
Succès de l'élément GRANT.  
Succès de l'élément GRANT.

d- Lancer SQL DEVELOPER puis créer une nouvelle connexion (Nom connexion : con3, Rôle : sysdba, SID : orcl, User/pwd : Std\_mgr / std\_pass )

Nom de connexion	con3
Nom utilisateur	Std_mgr
Mot de passe	*****
<input checked="" type="checkbox"/> Enregistrer le mot de passe	
<b>Oracle</b> Access	
Type de connexion	De base
Rôle	par défaut
Nom d'hôte	localhost
Port	1521
<input checked="" type="radio"/> SID	ord

**e- Exécuter les script « Gestion\_des\_etudiants.sql » pour créer les tables : ETUDIANT ; MODULE ; INSCRIPTION ; PREREQUIS ; EXAMEN ; RESULTAT.**

```
CREATE TABLE ETUDIANT (numEtud number, nom varchar2 (40), prenom varchar2 (40), datenaiss date, civilite varchar2 (4), Patronymie
varchar2 (40), numsecu varchar2 (15) NOT NULL, moyenne NUMBER DEFAULT NULL);
CREATE TABLE MODULE (codMod number, nomMod varchar2 (15), effecMax number DEFAULT 30, effec number default 0);
CREATE TABLE EXAMEN (codMod number, codExam number, dateExam date);
CREATE TABLE INSCRIPTION (numEtud number, codMod number, dateInsc date default sysdate);
CREATE TABLE PREREQUIS (codMod number, codModPrereq number, noteMin number (4, 2) NOT NULL);
CREATE TABLE RESULTAT (codMod number, codExam number, numEtud number, Note number (4, 2));
```

```

CREATE TABLE ETUDIANT...
CREATE TABLE MODULE...
CREATE TABLE EXAMEN...
CREATE TABLE INSCRIPTION...
CREATE TABLE PREREQUIS...
CREATE TABLE RESULTAT...

ALTER TABLE ETUDIANT ADD CONSTRAINT pk_etudiant PRIMARY KEY (numEtud);
ALTER TABLE MODULE ADD CONSTRAINT pk_module PRIMARY KEY (codMod);
ALTER TABLE EXAMEN ADD CONSTRAINT pk_examen PRIMARY KEY (codMod, codExam);
ALTER TABLE PREREQUIS ADD CONSTRAINT pk_prerequis PRIMARY KEY (codMod, codModPrereq);
ALTER TABLE INSCRIPTION ADD CONSTRAINT pk_inscription PRIMARY KEY (codMod, numEtud);
ALTER TABLE RESULTAT ADD CONSTRAINT pk_resultat PRIMARY KEY (codMod, numEtud, codExam);
ALTER TABLE INSCRIPTION ADD (CONSTRAINT fk_inscription_etudiant FOREIGN KEY (numEtud) REFERENCES ETUDIANT (numEtud) , CONSTRAINT fk_inscription_module FOREIGN KEY (codMod) R
ALTER TABLE PREREQUIS ADD (CONSTRAINT fk_prerequis_codmod FOREIGN KEY (codMod) REFERENCES MODULE (codMod), CONSTRAINT fk_prerequis_codmodprereq FOREIGN KEY (codModPrereq) RE
ALTER TABLE EXAMEN ADD CONSTRAINT fk_examen FOREIGN KEY (codMod) REFERENCES MODULE (codMod);
ALTER TABLE RESULTAT ADD (CONSTRAINT fk_resultat_examen FOREIGN KEY (codMod , codExam) REFERENCES EXAMEN (codMod , codExam), CONSTRAINT fk_resultat_inscription FOREIGN KEY (
ALTER TABLE ETUDIANT ADD (CONSTRAINT ck_civilite CHECK (civilite IN ('Mr', 'Mae', 'Mlle')), CONSTRAINT ck_civilite_numse CHECK (SUBSTR (numsecu, 1, 1) = '2' OR patronyme IS
  
```

Sortie de script x | Tâche terminée en 0,052 secondes

```

table EXAMEN créé(e).
table INSCRIPTION créé(e).
table PREREQUIS créé(e).
table RESULTAT créé(e).
table ETUDIANT modifié(e).
table MODULE modifié(e).
table EXAMEN modifié(e).
table PREREQUIS modifié(e).
table INSCRIPTION modifié(e).
table RESULTAT modifié(e).
table INSCRIPTION modifié(e).
table PREREQUIS modifié(e).
table EXAMEN modifié(e).
table RESULTAT modifié(e).
table ETUDIANT modifié(e).
  
```

**f- Implémenter les contraintes suivantes. Vous répertorierez les numéros d'erreurs que vous affecterez à chaque levée d'exception. Créer et tester vos déclencheurs.**

- ```

*****
- On ne doit pas modifier la note min dans la table PREREQUIS.
*****

*****
- On ne peut pas créer un examen pour un module que s'il y a des élèves inscrits dans ce module.
*****

*****
- Un élève ne peut passer un examen que si sa date d'inscription est antérieure à la date de l'examen.
*****

*****
- Dans un module, il ne doit pas y avoir plus de effecMax élèves inscrits.
*****
  
```