

ABDD TP

RELARQUE IMPORTANTE

Pour l'ensemble des TP, on demande de réaliser un compte-rendu du travail effectué. Ce compte-rendu comprendra un rapport (manuscrit ou réalisé sur traitement de texte) détaillant les points intéressants abordés à chaque question :

- requêtes SQL correspondant aux questions posées
- résultat de l'exécution de la requête
- réponses aux questions soulevées dans le sujet,

LE rapport sera envoyé dans la boîte email suivante bouhadiba.amine@gmail.com avant le 01/02/2022

Soit le modèle relationnel suivant relatif à la gestion des projets d'un ensemble de laboratoires de recherche.

Laboratoire(CodeLab, NomLab, Chef_de_Lab*, Date-création, SiteWeb).

Chercheur(NumCh, NomCh, CodeLab*, CodeUniversité*)

Université(CodeUniversité, NomUniversité)

Projet(CodePrj, Date-Début, Date-fin, Responsable, TypePrj*, Budget)

TypeProjet(CodeTypePrj, Libellé)

Paraticipe(NumCh*, CodePrj*, Charge_Horaire).

*Remarque : les clés primaires sont soulignées et les clés étrangères sont marquées par **

L'ensemble des instances des différentes tables sont représentées dans les tableaux suivants :

TABLE Laboratoire

CodeLab	NomLab	Chef_de_Lab	Date-création	SiteWeb
LSI	Laboratoire des Systèmes Informatiques	1	06/09/1999	www.lsi-dz.com
LIA	Laboratoire d'intelligence Artificielle	4	12/06/1993	www.lia-daz.com
LMA	Laboratoire de Mathématique appliquée	5	08/02/1982	www.lma.fr
LGP	Laboratoire de Génie des Procédés	7	24/04/2001	http://www.lgp.com

TABLE Chercheur

NumCh	NomCh	Genre	CodeLab	CodeUniversité
1	Salim Ahmed	M	LSI	USTHB
2	Karim Abed	M	LGP	UMBB
3	Salima Herrath	F	LMA	USDB
4	Karima Mahi	F	LSI	UAT
5	Said Khalef	M	LIA	ESI
6	Hichem Labeled	M	LIA	USTHB
7	Souad Khadem	F	LMA	UAT
8	Rafik Bradi	M	LGP	ESI

TABLE Université

CodeUniversité	NomUniversité
USTHB	Université de Bab Ezzouare
UMBB	Université de Boumerdes
USDB	Université de Blida
UAT	Université de Laghouat
ESI	Ecole Supérieure d'Informatique

TABLE TypeProjet

CodeTypePrj	LibelléType
PNR	Projet National
CNEPRU	Recherche Universitaire
AC	Accord Cadre
CI	Coopération Internationale

TABLE Projet

CodePrj	DateDébut	DateFin	Responsable	Budget	TypePrj
P1	5/07/2008	5/08/2011	Salim Ahmed	1 920 000	PNR
P2	6/07/2007	6/09/2010	Karim Abed	2 620 000	CNEPRU
P3	7/07/2008	7/04/2009	Salima Herrath	2 240 000	PNR
P4	8/07/2008	8/09/2010	Karima Mahi	2 230 000	CNEPRU
P5	9/07/2006	9/07/2010	Said Khalef	2 610 000	PNR
P6	10/07/2008	10/08/2009	Hichem Labeled	2 170 000	CI
P7	11/07/2009	11/04/2014	Souad Khadem	1 940 000	CI
P8	12/07/2008	12/02/2010	Rafik Bradi	1 610 000	PNR
P9	13/07/2007	13/08/2008	Salim Ahmed	1 820 000	CNEPRU
P10	14/07/2008	14/07/2013	Karim Abed	2 520 000	AC
P11	16/07/2007	16/09/2009	Salima Herrath	1 920 000	CNEPRU
P12	17/07/2008	17/07/2012	Karima Mahi	4 500 000	PNR
P13	18/07/2009	18/12/2011	Said Khalef	2 035 000	CNEPRU
P14	19/07/2008	19/07/2009	Hichem Labeled	1 480 000	PNR
P15	20/07/2010	20/11/2014	Souad Khadem	2 085 000	AC

TABLE PARTICIPER

NuméroCh	CodePrj	Charge_Horaire
1	P11	200
1	P2	259
2	P12	150
2	P3	320
3	P3	222
3	P14	240
4	P15	125
4	P4	127
5	P9	122
6	P8	322
8	P1	232
6	P5	172
7	P6	198
7	P7	90
8	P10	99
8	P13	187
2	P2	234
3	P5	333
5	P6	434

Syntaxe de création d'une table

```
CREATE TABLE [schéma.]nomTable
```

```
(colonne1 type1 [DEFAULT valeur1] [NOT NULL] [,
colonne2 type2 [DEFAULT valeur2] [NOT NULL] ]
[CONSTRAINT nomContrainte1 typeContrainte1]);
```

Quatre types de contraintes

```
CONSTRAINT nomContrainte
```

- UNIQUE (colonne1 [,colonne2])
- PRIMARY KEY (colonne1 [,colonne2])
- FOREIGN KEY (colonne1 [,colonne2]) REFERENCES [schma.] nomTablePere (colonne1 [,colonne2]) [ON DELETE { CASCADE | SET NULL }]
- CHECK (condition)

Partie I : Langage de définition de données

1. Créer les relations de base avec toutes les contraintes d'intégrité.
2. Ajouter l'attribut DATE-NAISSANCE à la relation Chercheur. Vérifier l'ajout.
3. Supprimer la colonne DateFin dans la table Projet. Vérifier la suppression. Recréez cette colonne.
4. Renommer la colonne « Charge Horaire » dans la table Participer par « ChargeH ». Vérifier.
5. Marquer la colonne Genre comme non utilisée. Vérifier qu'elle a été marquée.
6. Ajouter la contrainte suivante : le budget d'un projet ne doit pas dépasser 5 000 000 DA.

Partie II : Langage de manipulation de données

7. Remplir toutes les tables par les instances représentées ci-dessus.
8. Modifier le laboratoire de « Rafik Bradi » par LSI.
9. Modifier le budget du projet P6 à 2 000 000.
10. Supprimer tous les chercheurs du laboratoire LMA
11. Modifier le budget du projet P12 en 6 000 000. Désactiver la contrainte pour autoriser la modification. Réactiver la contrainte. Voici la commande de réactivation

```
ALTER TABLE          nomTable ENABLE CONSTRAINT  nomContrainte [EXCEPTIONS INTO tableErreurs] ;
```

Créer la table « tableErreurs » pour identifier les tuples qui ne vérifient pas la contrainte.

```
CREATE TABLE TableErreurs (adresse ROWID,      utilisateur VARCHAR2(30),      nomTable VARCHAR2(30),      nomContrainte VARCHAR2(30));
```

Partie III : Langage d'interrogation de données

12. Quels sont les chercheurs (Nom, Prénom, Nom d'université) du laboratoire LSI?
13. Quel est la charge horaire moyenne de chaque projet?
14. Quel est le projet CNEPRU ayant le maximum de budget ?
15. Classer les laboratoires par rapport à la charge horaire globale fournie par ses membres.

Quelques Syntaxes

• Création d'un Tablespace

```
CREATE TABLESPACE SGBD_TBS
DATAFILE 'D:\Oracle_Tbs\tbs_SGBD.dat' SIZE 100M
ONLINE;
```

Le descriptif de la création du tablespace SGBD_TempTBS est :

```
CREATE TABLESPACE SGBD_TBS          Spécifie le nom du tablespace de la base.
DATAFILE 'F:\Oracle_Tbs\tbs_SGBD.dat' SIZE 100M      Spécifie le nom complet du fichier système ainsi que sa taille en méga octets.
ONLINE;                                             Disponible immédiatement suite à sa création
```

• Création d'un Tablespace temporaire

```
CREATE TEMPORARY TABLESPACE SGBD_TempTBS TEMPFILE 'D:\Oracle_Tbs\temp_SGBD.dat'
SIZE 100M ;
```

• Création d'un utilisateur

```
Create User Nom_User
Identified by MotDePasse
Default Tablespace Nom_Table space
Temporary Tablespace Nom_Temp_Tablespace.
```

Partie IV : droit d accès

Le créateur d'un objet peut décider de donner (ou de supprimer) certains droits d'accès à tout autre utilisateur de sa connaissance.

GRANT privilège [ON table/vue] TO utilisateur [WITH GRANT OPTION]

Cet ordre permet de donner le privilège concerné sur la table ou la vue à l'utilisateur.

Un utilisateur ayant accordé un privilège peut le reprendre à tout moment à l'aide de l'ordre RE-VOKE :

REVOKE privilège ON [table/vue] FROM utilisateur

Les privilèges qui peuvent être donnés sont les suivants :

SELECT droit de lecture, **INSERT** droit d'insertion de lignes, **UPDATE** droit de mise-à-jour de lignes, **DELETE** droit de suppression de lignes,

ALTER droit de modification de la définition de la table, **INDEX** droit de création d'index, **ALL** tous les droits ci-dessus.

Un utilisateur ayant reçu un privilège avec la mention facultative **WITH GRANT OPTION** peut les transmettre à son tour à un autre utilisateur (User3).

Questions :

1. Créez un autre utilisateur : utilisateur 2.
2. Connectez-vous l' id cet utilisateur Que remarquez-vous ?
3. Donner les privilèges suivants à utilisateur 2 2 : créer des tables, des utilisateurs. Vérifier.
4. Exécutez la requête Q1 suivante : **Select * from SGBD.Laboratoire.** Que remarquez-vous ?
5. Donner les droits de lecture à cet utilisateur pour la table Laboratoire. Exécutez la requête Q1 maintenant.
6. Enlever les privilèges précédemment accordés.
7. Vérifier que les privilèges ont bien été supprimés.

La syntaxe SQL de création d'un utilisateur

CREATE USER utilisateur

IDENTIFIED {**BY** motdePasse | **EXTERNALLY** | **GLOBALLY AS** 'nomExterne' }

[**DEFAULT TABLESPACE** nomTablespace [**QUOTA** {entier [K | M] | **UNLIMITED**} **ON** nomTablespace]]

[**TEMPORARY TABLESPACE** nomTablespace [**QUOTA** {entier [K | M] | **UNLIMITED**} **ON** nomTablespace].] [**PROFILE** nomProfil]

[**PASSWORD EXPIRE**] [**ACCOUNT** {**LOCK** | **UNLOCK** }];

- **IDENTIFIED BY motdePasse** permet d'affecter un mot de passe à un utilisateur local (cas le plus courant et le plus simple).
- **IDENTIFIED BY EXTERNALLY** permet de se servir de l'authenticité du système d'exploitation pour s'identifier Oracle (cas des compte OPS\$ pour Unix).
- **IDENTIFIED BY GLOBALLY** permet de se servir de l'authenticité d'un système d'annuaire.
- **DEFAULT TABLESPACE** nomTablespace associe un espace disque de travail (appelé tablespace) à l'utilisateur.
- **TEMPORARY TABLESPACE** nomTablespace associe un espace disque temporaire (dans lequel certaines opérations se dérouleront) à l'utilisateur.
- **QUOTA** permet de limiter ou pas chaque espace alloué.
- **PROFILE** nomProfil affecte un profil (caractéristiques système relatives au CPU et aux connexions) à l'utilisateur.
- **PASSWORD EXPIRE** pour obliger l'utilisateur à changer son mot de passe à la première connexion (par défaut il est libre). Le DBA peut aussi changer ce mot de passe.
- **ACCOUNT** pour verrouiller ou libérer l'accès à la base (par défaut UNLOCK).

La syntaxe SQL de Création d'un Profil

CREATE PROFILE nomProfil

LIMIT { ParamètreRessource | ParamètreMotdePasse } [ParamètreRessource | ParamètreMotdePasse];

ParamètreRessource : {{SESSIONS_PER_USER|CPU_PER_SESSION|CPU_PER_CALL|CONNECT_TIME|

IDLE_TIME|LOGICAL_READS_PER_SESSION|LOGICAL_READS_PER_CALL | COMPOSITE_LIMIT}}{entier | UNLIMITED | DEFAULT } |

PRIVATE_SGA {entier[K|M] | UNLIMITED | DEFAULT}}

ParamètreMotdePasse : { FAILED_LOGIN_ATTEMPTS | PASSWORD_LIFE_TIME | PASSWORD_REUSE_TIME | PASSWORD_REUSE_MAX |

PASSWORD_LOCK_TIME | PASSWORD_GRACE_TIME } { expression | UNLIMITED | DEFAULT } }

Les options principales sont les suivantes :

- **SESSIONS_PER_USER** : nombre de sessions concurrentes autorises.
- **CPU_PER_SESSION** : temps CPU maximal pour une session en centièmes de secondes.
- **CPU_PER_CALL** : temps CPU autorisé pour un appel noyau en centièmes de secondes.
- **CONNECT_TIME** : temps total autorisé pour une session en minutes (pratique pour les examens de TP minutés).
- **IDLE_TIME** : temps d'inactivité autorisé, en minutes, au sein d'une même session (pour les étudiants qui ne clôturent jamais leurs sessions).
- **PRIVATE_SGA** : espace mémoire privé alloué dans la SGA (System Global Area).
- **FAILED_LOGIN_ATTEMPTS** : nombre de tentatives de connexion avant de bloquer l'utilisateur.
- **PASSWORD_LIFE_TIME** : nombre de jours de validité du mot de passe (il expire sil n'est pas changé au cours de cette période).
- **PASSWORD_REUSE_TIME** : nombre de jours avant que le mot de passe puisse être utilisé à nouveau. Si ce paramètre est initialisé à un entier, le paramètre **PASSWORD_REUSE_MAX** doit être passé à **UNLIMITED**.
- **PASSWORD_REUSE_MAX** : nombre de modifications de mot de passe avant de pouvoir réutiliser le mot de passe courant. Si ce paramètre est initialisé un entier, le paramètre **PASSWORD_REUSE_TIME** doit être passé à **UNLIMITED**.
- **PASSWORD_LOCK_TIME** : nombre de jours d'interdiction d'accès à un compte après que le nombre de tentatives de connexions a été atteint.
- **PASSWORD_GRACE_TIME** : nombre de jours d'une période de grâce qui prolonge l'utilisation du mot de passe avant son changement (un message d'avertissement s'affiche lors des connexions). Après cette période le mot de passe expire.

Syntaxe de Création d'un rôle

CREATE ROLE nomRle [**NOT IDENTIFIED** | **IDENTIFIED**

{**BY** motdePasse | **USING** [schma.]paquetage | **EXTERNALLY** | **GLOBALLY** }]