

Compte Rendu Tp2

Admin. Bdd

Réalisé par Nadhir Halbouni

L2CS01

Création des utilisateurs :

1. Créer les utilisateurs suivants :

```
create user c##etud1 identified by etud1;  
create user c##etud2 identified by etud2;  
create user c##prof1 identified by prof1;  
create user c##prof2 identified by prof4;  
create user c##prof3 identified by prof3;
```

User C##ETUD1 créé(e).

User C##ETUD2 créé(e).

User C##PROF1 créé(e).

User C##PROF2 créé(e).

User C##PROF3 créé(e).

2. Modifier le mot de passe de l'utilisateur prof2 à prof2 :

```
alter user c##prof2 identified by prof2;
```

User C##PROF2 modifié(e).

3. Supprimer l'utilisateur prof3 :

```
drop user c##prof3;
```

User C##PROF3 supprimé(e).

4. Attribuer les privilèges suivants à l'utilisateur etud1 :

```
grant create session, create view, create any table, select any table, update any table, drop any table to c##etud1 with admin option;
```

Succès de l'élément Grant.

5. Connecter l'utilisateur etud1 sur sql* plus :

```
SQL> connect c##etud1/etud1;  
Error accessing PRODUCT_USER_PROFILE  
Warning: Product user profile information not loaded!  
You may need to run PUPBLD.SQL as SYSTEM  
Connected.
```

6. Créer le table « cours » :

```
SQL> create table cours(matiere varchar(20) primary key);  
  
Table created.
```

7. Révoquer le privilège « création des tables » :

```
revoke create any table from c##etud1;
```

8. Créer la table « tp » :

```
SQL> create table tp(matiere varchar(20) primary key);
create table tp(matiere varchar(20) primary key)
*
ERROR at line 1:
ORA-01031: insufficient privileges
```

9. Révoquer tous les privilèges de l'utilisateur etud1 :

```
revoke create session, create view, select any table, update any table, drop any table from c##etud1;
```

Succès de l'élément Revoke.

Création des rôles :

1. Créer les rôles suivants et attribuer à chaque rôle les privilèges adéquats :

```
create role c##etudiant identified by etudiant;
grant create view, create session, create any table, select any table, update any table, drop any table to c##etudiant with admin option;

create role c##professeur identified by professeur;
grant all privileges to c##professeur with admin option;
```

Role C##ETUDIANT créé(e).

Succès de l'élément Grant.

Role C##PROFESSEUR créé(e).

Succès de l'élément Grant.

2. Attribuer à chaque utilisateur le rôle qui lui correspond :

```
grant c##etudiant to c##etud1;
grant c##etudiant to c##etud2;
grant c##professeur to c##prof1;
grant c##professeur to c##prof2;
```

Succès de l'élément Grant.

Succès de l'élément Grant.

Succès de l'élément Grant.

Succès de l'élément Grant.

3. Afficher les rôles de la base de données :

```
select * from dba_roles;
```

	ROLE	PASSWORD_REQUIRED	AUTHENTICATION_TYPE
1	CONNECT	NO	NONE
2	RESOURCE	NO	NONE
3	DBA	NO	NONE
4	SELECT_CATALOG_ROLE	NO	NONE
5	EXECUTE_CATALOG_ROLE	NO	NONE
6	DELETE_CATALOG_ROLE	NO	NONE
7	EXP_FULL_DATABASE	NO	NONE
8	IMP_FULL_DATABASE	NO	NONE
9	LOGSTDBY_ADMINISTRATOR	NO	NONE
10	DBFS_ROLE	NO	NONE
11	AQ_ADMINISTRATOR_ROLE	NO	NONE
12	AQ_USER_ROLE	NO	NONE
13	DATAPUMP_EXP_FULL_DATABASE	NO	NONE
14	DATAPUMP_IMP_FULL_DATABASE	NO	NONE
15	ADM_PARALLEL_EXECUTE_TASK	NO	NONE
16	GATHER_SYSTEM_STATISTICS	NO	NONE
17	XDB_WEBSERVICES_OVER_HTTP	NO	NONE
18	RECOVERY_CATALOG_OWNER	NO	NONE
19	SCHEDULER_ADMIN	NO	NONE
20	HS_ADMIN_SELECT_ROLE	NO	NONE
21	HS_ADMIN_EXECUTE_ROLE	NO	NONE
22	HS ADMIN ROLE	NO	NONE

4. Afficher les rôles attribués à tous les utilisateurs et les rôles de la base de données :

```
select * from dba_role_privs;
```

	GRANTEE	GRANTED_ROLE	ADMIN_OPTION	DEFAULT_ROLE
1	SYS	XDB_SET_INVOKER	YES	YES
2	SYS	XDBADMIN	YES	YES
3	SYS	IMP_FULL_DATABASE	YES	YES
4	DBA	SCHEDULER_ADMIN	YES	YES
5	DBA	DATAPUMP_IMP_FULL_DATABASE	NO	YES
6	SYSTEM	AQ_ADMINISTRATOR_ROLE	YES	YES
7	EXECUTE_CATALOG_ROLE	HS_ADMIN_EXECUTE_ROLE	NO	YES
8	HS_ADMIN_ROLE	HS_ADMIN_EXECUTE_ROLE	NO	YES
9	OEM_MONITOR	SELECT_CATALOG_ROLE	NO	YES
10	APEX_040000	RESOURCE	YES	YES
11	SYS	APEX_ADMINISTRATOR_ROLE	YES	YES
12	SYS	RECOVERY_CATALOG_OWNER	YES	YES
13	SYS	DELETE_CATALOG_ROLE	YES	YES
14	DBA	DELETE_CATALOG_ROLE	YES	YES
15	DBA	EXECUTE_CATALOG_ROLE	YES	YES
16	HR	RESOURCE	NO	YES
17	NADHIR	DBA	NO	YES
18	SYS	DBA	YES	YES
19	DBA	PLUSTRACE	YES	YES
20	IMP_FULL_DATABASE	EXECUTE_CATALOG_ROLE	NO	YES
21	LOGSTDBY_ADMINISTRATOR	RESOURCE	NO	YES
22	DATAPUMP_EXP_FULL_DATABASE	EXP_FULL_DATABASE	NO	YES

5. Connecter l'utilisateur etud1 sur sql* plus :

```
SQL> connect c##etud1/etud1;
ERROR:
ORA-01045: user C##ETUD1 lacks CREATE SESSION privilege; logon denied
```

6. Attribuer connect, etudiant to etud1 :

```
grant connect, c##etudiant to c##etud1;
```

Succès de l'élément Grant.

7. Connecter etud1 sur sql* plus :

```
SQL> connect c##etud1/etud1;
Error accessing PRODUCT_USER_PROFILE
Warning: Product user profile information not loaded!
You may need to run PUPBLD.SQL as SYSTEM
Connected.
```

8. Afficher les rôles actifs :

```
SQL> select * from session_roles;

ROLE
-----
CONNECT
```

9. Activer le rôle etudiant pour l'utilisateur etud1 :

```
SQL> set role c##etudiant identified by etudiant;

Role set.
```

10. Afficher les rôles actifs :

```
SQL> select * from session_roles;

ROLE
-----
C##ETUDIANT
```

11. Créer la table « Tp » :

```
SQL> create table tp(matiere varchar(20) primary key);

Table created.
```

12. Sur sql-dev, révoquer le privilège de la création de table au rôle étudiant :

```
revoke create any table from c##etudiant;
```

```
Succès de l'élément Revoke.
```

13. Créer la table « Td » :

```
SQL> create table td(matiere varchar(20) primary key);
create table td(matiere varchar(20) primary key)
*
ERROR at line 1:
ORA-01031: insufficient privileges
```

14. Supprimer le rôle « professeur » :

```
drop role c##professeur;
```

```
Role C##PROFESSEUR supprimé(e).
```

Gestion des profils :

1. Créer un profile de ressources dédié aux professeurs avec les limitations suivantes :

```
create profile c##professeur  
limit sessions_per_user 1  
cpu_per_session unlimited  
cpu_per_call 3000  
logical_reads_per_session default  
logical_reads_per_call 5000  
private_sga 80k  
password_life_time 60  
password_reuse_time 10;
```

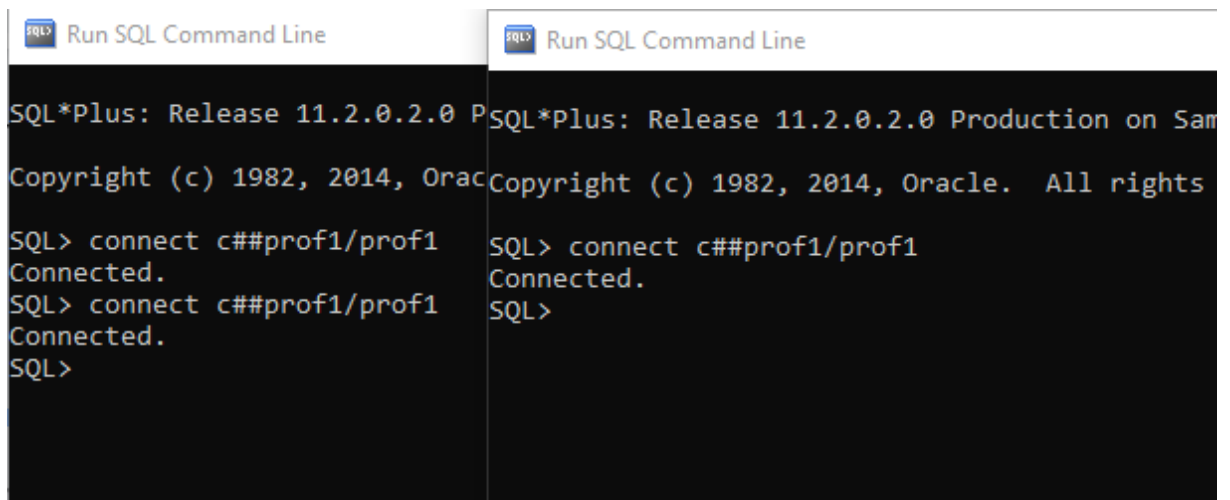
Profile C##PROFESSEUR créé(e).

2. Attribuer le profile « professeur » à l'utilisateur « prof1 » :

```
alter user c##prof1 profile c##professeur;
```

User C##PROF1 modifié(e).

3. Connecter l'utilisateur « prof1 » sur deux sessions :



Il ya un problème, en principe dans la 2^{ème} fois que j'essaye de se connecter il doit afficher le message d'erreur suivant :

ORA-02391: exceeded simultaneous SESSIONS_PER_USER limit