

UNIVERSITÉ DES SCIENCES ET DE LA TECHNOLOGIE HOUARI
BOUMEDIENE



ARCHITECTURES ET SYSTÈMES DES BASE DE DONNÉES

Rapport de Travaux Pratiques N°2
Gestion des droits

Binôme

MOHAMMEDI HAROUNE
HOUACINE NAILA AZIZA

Professeur

Pr. BOUKHALFA KAMEL

13 octobre 2017

1 Création de l'Admin

Lors de la création de l'utilisateur Admin un mot de passe lui est aussi donné. Aussi un quota de la tablespace INTERVENTION_TBS lui est affecté.

1.1 Requête

```
CREATE USER Admin IDENTIFIED BY Admin ;  
Alter user Admin quota unlimited on INTERVENTION_TBS;
```

1.2 Résultat

```
User created.  
User altered.
```

2 Connexion à l'utilisateur Admin.

2.1 Requête

```
connect Admin/Admin;
```

l'utilisateur ne peut pas se connecter car il n'a pas le droit de créer un compte (session).

2.2 Résultat

```
ERROR :  
ORA-01045: user ADMIN lacks CREATE SESSION privilege; logon denied
```

3 Permission de création d'une session accordée à l'utilisateur Admin puis connexion de se dernier a son compte.

Cette permission lui est accordé par le créateur de l'utilisateur Admin, et il s'agit d'un privilège de création donc on utilise la commande GRANT.

3.1 Requête

```
GRANT CREATE SESSION TO Admin;  
connect Admin/Admin;
```

Maintenant l'utilisateur Admin peut créer sa session et donc se connecter à l'aide de son nom : Admin et mot de passe : Admin.

3.2 Résultat

```
Grant succeeded.  
Connected.
```

4 Donner les privilèges de création de tables et d'utilisateurs à l'Admin.

C'est l'utilisateur qui a créer l'Admin qui va lui affecter les privilèges.

4.1 Requête

```
connect DBAINTERVENTION/DBAINTERVENTION;  
GRANT CREATE TABLE TO Admin;  
GRANT CREATE USER TO Admin;
```

4.2 Résultat

```
Connected.  
Grant succeeded.  
Grant succeeded.
```

Vérification des privilèges accordés à l'Admin

4.3 Requête

```
connect Admin/Admin;  
  
CREATE TABLE My_TABLE(  
my_num INTEGER PRIMARY KEY,  
my_name VARCHAR2(30)  
);  
  
CREATE USER USER0 IDENTIFIED BY USER0;
```

4.4 Résultat

```
Connected.  
  
Table created.  
  
User created.
```

5 Exécution de la requête : Select * from DBAINTERVENTION.EMPLOYE.

Tentative de l'Admin d'accéder en lecture à la table EMPLOYE de l'utilisateur DBAINTERVENTION.

5.1 Requête

```
Select * from DBAINTERVENTION.EMPLOYE ;
```

5.2 Résultat

```
ERROR at line 1:  
ORA-00942: table or view does not exist
```

l'utilisateur Admin n'a pas pu effectuer la lecture sur la table EMPLOYE, car il n'a que les droits de création de table et pas de lecture d'ailleurs oracle n'avoue même pas l'existence de la table EMPLOYE

6 Donner les droits de lecture à l'Admin pour la table EMPLOYE.

L'utilisateur affectera le droit de lecture sur sa table DBAINTERVENTION à l'Admin.

6.1 Requête

```
connect DBAINTERVENTION/DBAINTERVENTION;  
GRANT SELECT ON EMPLOYE TO Admin;
```

6.2 Résultat

```
Connected.  
Grant succeeded.
```

Exécution de la requête `SELECT * FROM DBAINTERVENTION.EMPLOYE`

6.3 Requête

```
connect Admin/Admin;  
SELECT * FROM DBAINTERVENTION.EMPLOYE ;
```

6.4 Résultat

```
Connected.  
  
NUMEMPLOYE  PRENOMEMP      NOMEMP      CATEGORIE      SALAIRE  
-----  
53    lachemi      bouzid      MECANICIEN      25000  
54    bouchemla    elias      ASSISTANT        10000  
55      hadj      zouhir      ASSISTANT        12000  
56    oussedik      hakim      MECANICIEN      20000  
57      abad    abdelhamid    ASSISTANT        13000  
58    babaci      tayeb      MECANICIEN      21300  
59    belhamidi    mourad      MECANICIEN      19500  
60    igoudjil    redouane    ASSISTANT        15000  
61      koula      bahim      MECANICIEN      23100  
62    rahali      ahcene      MECANICIEN      24000  
63    chaoui      ismail      ASSISTANT        13000  
64      badi      hatem      ASSISTANT        14000  
65    mohammedi    mustapha    MECANICIEN      24000  
66      fekar    abdelaziz    ASSISTANT        13500  
67    saidouni      wahid      MECANICIEN      25000  
68    boularas      farid      ASSISTANT        14000  
69    chaker      nassim      MECANICIEN      26000  
71      terki      yacine      MECANICIEN      23000
```

72	tebibel	ahmed	ASSISTANT	17000
80	lardjoune	karim		25000

20 rows selected.

7 Le centre de gestion des interventions augmente les salaires des employés dont le nombre total des interventions est supérieur à 5.

il faut d'abord comptabiliser le nombre de d'interventions de chaque employé appartenant à la table INTERVENANT puis comparer la valeur avec 5 dans le cas ou le nombre d'intervention est supérieure à 5 alors : on passe a la recherche de l'employé en question dans la table EMPLOYE puis l'on ajouter la somme à son salaire.

7.1 Requête

```
UPDATE DBAINTERVENTION.EMPLOYE E
SET E.SALAIRE = E.SALAIRE + 1000
WHERE E.NUMEMPLOYE IN
  (SELECT I.NUMEMPLOYE
   FROM DBAINTERVENTION.INTERVENANT I
   WHERE (SELECT COUNT(NUMINTERVENTION)
          FROM DBAINTERVENTION.INTERVENANT II
          WHERE I.NUMEMPLOYE = II.NUMEMPLOYE
          GROUP BY NUMEMPLOYE) > 5);
```

7.2 Résultat

```
ERROR at line 1:
ORA-00942: table or view does not exist
```

l'utilisateur Admin n'a ni les droits de modification (mise a jours) sur la table EMPLOYE ni les droits de lecture sur la table INTERVENANT d'ailleurs oracle n'avoue même pas l'existence des tables

8 Donner les droits de mise à jour à l'utilisateur Admin pour la table EMPLOYE, les droits de lecture sur la table INTERVENANT, Puis exécution de la commande précédente par l'Admin .

L'utilisateur DBAINTERVENTION accord à l'utilisateur Admin les droits UPDATE sur la table EMPLOYE et SELECT sur la table INTERVENANT.

8.1 Requête

```
connect DBAINTERVENTION/DBAINTERVENTION;
GRANT UPDATE ON EMPLOYE TO Admin;
GRANT SELECT ON INTERVENANT TO Admin;
```

8.2 Résultat

```
Connected.
Grant succeeded.
Grant succeeded.
```

Exécution de la requête par l'Admin après avoir reçu les droits nécessaires.

8.3 Requête

```
connect Admin/Admin;

UPDATE DBAINTERVENTION.EMPLOYE E
SET E.SALAIRE = E.SALAIRE + 1000
WHERE E.NUMEMPLOYE IN
  (SELECT I.NUMEMPLOYE
   FROM DBAINTERVENTION.INTERVENANT I
   WHERE (SELECT COUNT(NUMINTERVENTION)
          FROM DBAINTERVENTION.INTERVENANT II
          WHERE I.NUMEMPLOYE = II.NUMEMPLOYE
          GROUP BY NUMEMPLOYE) > 5);
```

8.4 Résultat

```
Connected.
0 rows updated.
```

Il n'y a aucun enregistrement qui correspond aux conditions de la requête.

9 Création d'un index NOMEMP_IX sur l'attribut NOMEMP de la table EMPLOYE.

9.1 Requête

```
CREATE INDEX NOMEMP_IX ON DBAINTERVENTION.EMPLOYE(NOMEMP);
```

9.2 Résultat

```
ERROR at line 1:
ORA-01031: insufficient privileges
```

on remarque que l'utilisateur Admin n'a pas le droit de création d'un index sur la table EMPLOYE, pour cela le privilège de création d'index sur la table en question doit lui être attribué; Aussi vu que l'Admin possède déjà des droits sur la table EMPLOYE, oracle reconnait que cette table existe mais que l'utilisateur n'a pas le droit à la création d'un index la dessus.

10 Donner les droits de création d'index à l'Admin pour la table EMPLOYE, puis réessayer la création de l'index.

L'utilisateur DBAINTERVENTION accorde le droit de création d'un index sur la table EMPLOYE.

10.1 Requête

```
connect DBAINTERVENTION/DBAINTERVENTION;
GRANT INDEX ON EMPLOYE TO Admin;
```

10.2 Résultat

```
Connected.
Grant succeeded.
```

Admin tente à nouveau la création de l'index.

10.3 Requête

```
connect Admin/Admin;  
CREATE INDEX NOMEMP_IX ON DBAINTERVENTION.EMPLOYE(NOMEMP);
```

10.4 Résultat

```
Connected.  
Index created.
```

Une fois le privilège de création d'index sur la table EMPLOYE accordé à l'Admin, la création de déroule normalement.

11 Enlever les privilèges précédemment accordés à l'utilisateur Admin.

L'utilisateur DBAINTERVENTION doit connaître les privilèges accordés et sur quelles tables afin de pouvoir les retirer.

11.1 Requête

```
CONNECT DBAINTERVENTION/DBAINTERVENTION;  
REVOKE SELECT ON EMPLOYE FROM Admin;  
REVOKE UPDATE ON EMPLOYE FROM Admin;  
REVOKE SELECT ON INTERVENANT FROM Admin;  
REVOKE INDEX ON EMPLOYE FROM Admin;
```

11.2 Résultat

```
Connected.  
  
Revoke succeeded.  
Revoke succeeded.  
Revoke succeeded.  
Revoke succeeded.
```

12 Vérifier que les privilèges ont bien été supprimés.

La vérification que les privilèges précédemment supprimés le sont bien, nous prendrons un exemple de test pour chaque privilège supprimé.

12.1 Requête

```
CONNECT Admin/Admin  
SELECT * FROM EMPLOYE  
UPDATE EMPLOYE SET SALAIRE=0 WHERE SALAIRE > 1000;  
SELECT * FROM INTERVENANT  
CREATE INDEX NOMEMP_IX ON DBAINTERVENTION.EMPLOYE(NOMEMP);
```

12.2 Résultat

```
ERROR at line 1:  
ORA-00942: table or view does not exist  
ERROR at line 1:  
ORA-00942: table or view does not exist  
ERROR at line 1:
```

```
ORA-00942: table or view does not exist
ERROR at line 1:
ORA-00942: table or view does not exist
```

Ainsi nous pouvons constater que tous les privilèges ont été retiré à l'Admin.

13 Création du profil « Interv_Profil » qui suit des caractéristiques données en TP.

la création d'un profile permet la restriction des ressources à un utilisateur.

13.1 Requête

```
CREATE PROFILE Interv_Profil
LIMIT SESSIONS_PER_USER 3
CPU_PER_SESSION DEFAULT
CPU_PER_CALL 35 CONNECT_TIME
90 IDLE_TIME 30
LOGICAL_READS_PER_SESSION DEFAULT
LOGICAL_READS_PER_CALL 1200
COMPOSITE_LIMIT DEFAULT
PRIVATE_SGA 25K
FAILED_LOGIN_ATTEMPTS 5
PASSWORD_LIFE_TIME 50
PASSWORD_REUSE_TIME 40
PASSWORD_REUSE_MAX UNLIMITED
PASSWORD_LOCK_TIME 1
PASSWORD_GRACE_TIME 5 ;
```

13.2 Résultat

```
Profile created.
```

14 Affectation de ce profil à l'utilisateur Admin.

14.1 Requête

```
ALTER USER Admin PROFILE Interv_Profil;
```

14.2 Résultat

```
User altered.
```

15 Création de rôle : « GESTIONNAIRE_DES_INTERVENTIONS » qui peut voir les tables EMPLOYE, VEHICULE, CLIENT et peut modifier les tables INTERVENTIONS et INTERVENANTS.

A la création d'un rôle on lui accorde un ensemble de privilèges, puis se rôle peut être affecté a plusieurs utilisateurs.

15.1 Requête

```
CREATE ROLE GESTIONNAIRE_DES_INTERVENTIONS IDENTIFIED;  
GRANT SELECT ON EMPLOYE TO GESTIONNAIRE_DES_INTERVENTIONS;  
GRANT SELECT ON VEHICULE TO GESTIONNAIRE_DES_INTERVENTIONS;  
GRANT SELECT ON CLIENT TO GESTIONNAIRE_DES_INTERVENTIONS;  
GRANT UPDATE ON INTERVENTIONS TO GESTIONNAIRE_DES_INTERVENTIONS;  
GRANT UPDATE ON INTERVENANT TO GESTIONNAIRE_DES_INTERVENTIONS;
```

15.2 Résultat

```
Role created.  
Grant succeeded.  
Grant succeeded.  
Grant succeeded.  
Grant succeeded.  
Grant succeeded.
```

16 Assignation de ce rôle à Admin. Avec vérification que les autorisations assignées au rôle GESTIONNAIRE_DES_INTERVENTIONS, ont été bien transférées à Admin.

Assigner le rôle à l'Admin avec la commande GRANT.

16.1 Requête

```
GRANT GESTIONNAIRE_DES_INTERVENTIONS TO Admin;
```

16.2 Résultat

```
Grant succeeded.
```

Vérifier que le rôle est transmis à l'Admin via une série de test suivant les privilèges accordés au rôle.

16.3 Requête

```
SELECT COUNT(NUMEMPLOYE) FROM DBAINTERVENTION.EMPLOYE;  
SELECT COUNT(NUMVEHICULE) FROM DBAINTERVENTION.VEHICULE;  
SELECT COUNT(NUMCLIENT) FROM DBAINTERVENTION.CLIENT;  
UPDATE DBAINTERVENTION.INTERVENTIONS SET COUTINTERV=COUTINTERV+100;  
UPDATE DBAINTERVENTION.INTERVENANT SET DATEDEBUT=DATEDEBUT+2;
```

16.4 Résultat

```
COUNT(NUMEMPLOYE)  
-----  
20  
COUNT(NUMVEHICULE)  
-----  
26  
COUNT(NUMCLIENT)  
-----  
22  
15 rows updated.  
26 rows updated.
```

les résultats de ces requêtes confirme que les privilèges transmis à l'Admin via le role 'GESTIONNAIRE_INTERVENTION fonctionnent correctement.