TP_3 BIO-SGBD

Djennaoui raouf 201400002338

1)
lister le catalogue DICT : select * from dict;
le nombre d'instrance : select count(*) from dict;
le resultat : 4568
la structure : discribe dict;
le resultat : la colonne **TABLE_NAME** et la colonne **COMMENTS**
2)
TABLE : ALL_TAB_COLUMNS
STRUCTURE: describe ALL_TAB_COLUMNS ; **figure(1)**

Nom	NULL ?	Type
OWNER	NOT NULL	VARCHAR2(128)
TABLE_NAME	NOT NULL	VARCHAR2(128)
COLUMN_NAME	NOT NULL	VARCHAR2(128)
DATA_TYPE		VARCHAR2(128)
DATA_TYPE_MOD		VARCHAR2(3)
DATA_TYPE_OWNER		VARCHAR2(128)
DATA_LENGTH	NOT NULL	NUMBER
DATA_PRECISION		NUMBER
DATA_SCALE		NUMBER
NULLABLE		VARCHAR2(1)
COLUMN_ID		NUMBER
DEFAULT_LENGTH		NUMBER
DATA_DEFAULT		LONG
NUM_DISTINCT		NUMBER
LOW_VALUE		RAW(2000)
HIGH_VALUE		RAW(2000)
DENSITY		NUMBER
NUM_NULLS		NUMBER
NUM_BUCKETS		NUMBER
LAST_ANALYZED		DATE
SAMPLE_SIZE		NUMBER
CHARACTER_SET_NAME		VARCHAR2(44)
CHAR_COL_DECL_LENGTH		NUMBER
GLOBAL_STATS		VARCHAR2(3)
USER_STATS		VARCHAR2(3)
AVG_COL_LEN		NUMBER
CHAR_LENGTH		NUMBER
CHAR_USED		VARCHAR2(1)
V80_FMT_IMAGE		VARCHAR2(3)
DATA_UPGRADED		VARCHAR2(3)
HISTOGRAM		VARCHAR2(15)
DEFAULT_ON_NULL		VARCHAR2(3)
IDENTITY_COLUMN		VARCHAR2(3)
EVALUATION_EDITION		VARCHAR2(128)
UNUSABLE_BEFORE		VARCHAR2(128)
UNUSABLE_BEGINNING		VARCHAR2(128)
COLLATION		VARCHAR2(100)

figure(1)

ROLE:

décrit les colonnes des tables, des vues et des clusters accessibles à l'utilisateur actuel

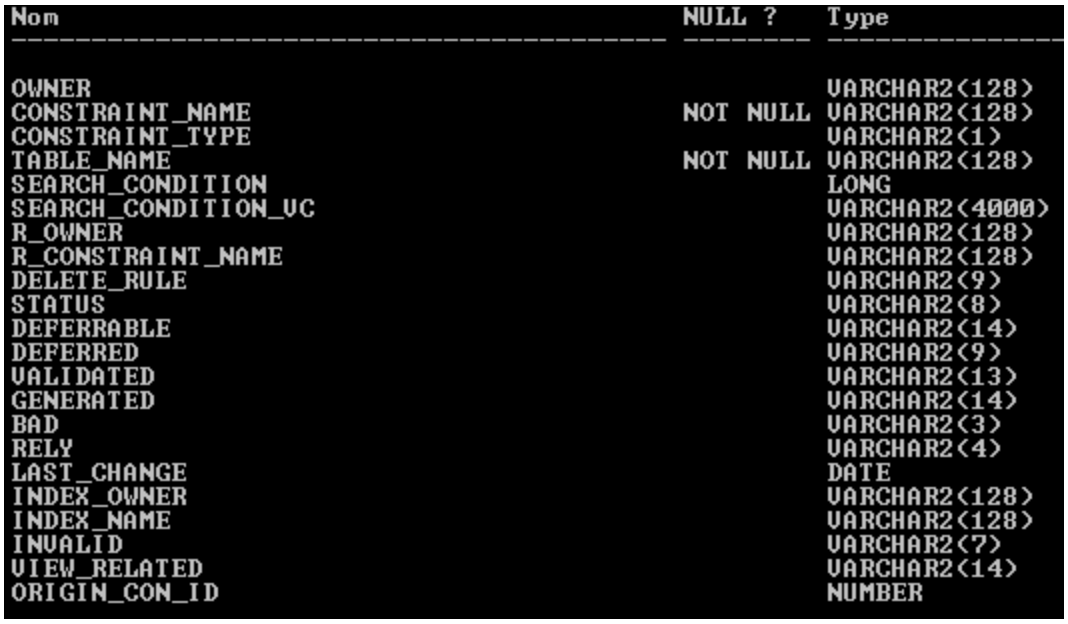
TABLE : USER_USERS
STRUCTURE: describe user_users ; **figure(2)**

Nom	NULL ?	Type
USERNAME	NOT NULL	VARCHAR2(128)
USER_ID	NOT NULL	NUMBER
ACCOUNT_STATUS	NOT NULL	VARCHAR2(32)
LOCK_DATE		DATE
EXPIRY_DATE		DATE
DEFAULT_TABLESPACE	NOT NULL	VARCHAR2(30)
TEMPORARY_TABLESPACE	NOT NULL	VARCHAR2(30)
LOCAL_TEMP_TABLESPACE		VARCHAR2(30)
CREATED	NOT NULL	DATE
INITIAL_RSRC_CONSUMER_GROUP		VARCHAR2(128)
EXTERNAL_NAME		VARCHAR2(4000)
PROXY_ONLY_CONNECT		VARCHAR2(1)
COMMON		VARCHAR2(3)
ORACLE_MAINTAINED		VARCHAR2(1)
INHERITED		VARCHAR2(3)
DEFAULT_COLLATION		VARCHAR2(100)
IMPLICIT		VARCHAR2(3)
ALL_SHARD		VARCHAR2(3)

figure(2)

ROLE:
décrit l'utilisateur actuel

TABLE : ALL_CONSTRAINTS
STRUCTURE: describe ALL_CONSTRAINTS ; **figure(3)**

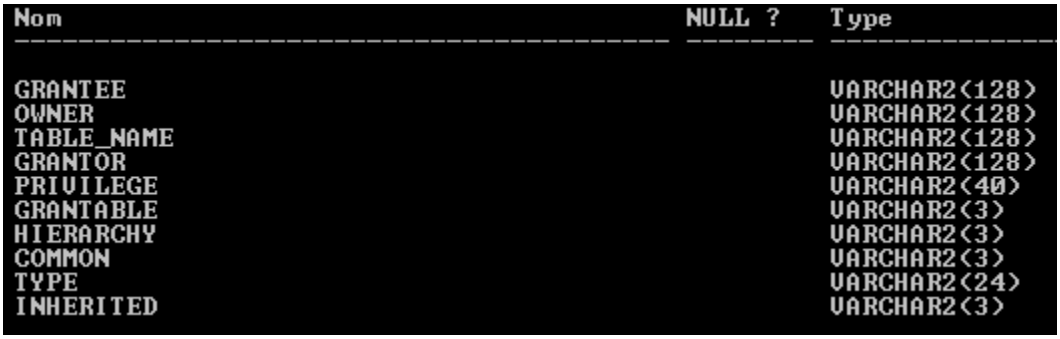


Nom	NULL ?	Type
OWNER		VARCHAR2(128)
CONSTRAINT_NAME	NOT NULL	VARCHAR2(128)
CONSTRAINT_TYPE		VARCHAR2(1)
TABLE_NAME	NOT NULL	VARCHAR2(128)
SEARCH_CONDITION		LONG
SEARCH_CONDITION_VC		VARCHAR2(4000)
R_OWNER		VARCHAR2(128)
R_CONSTRAINT_NAME		VARCHAR2(128)
DELETE_RULE		VARCHAR2(9)
STATUS		VARCHAR2(8)
DEFERRABLE		VARCHAR2(14)
DEFERRED		VARCHAR2(9)
VALIDATED		VARCHAR2(13)
GENERATED		VARCHAR2(14)
BAD		VARCHAR2(3)
RELY		VARCHAR2(4)
LAST_CHANGE		DATE
INDEX_OWNER		VARCHAR2(128)
INDEX_NAME		VARCHAR2(128)
INVALID		VARCHAR2(7)
VIEW_RELATED		VARCHAR2(14)
ORIGIN_CON_ID		NUMBER

figure(3)

ROLE:
décrit les définitions de contraintes sur les tables accessibles à l'utilisateur actuel

TABLE : USER_TAB_PRIVS
STRUCTURE: describe USER_TAB_PRIVS ; **figure(4)**



Nom	NULL ?	Type
GRANTEE		VARCHAR2(128)
OWNER		VARCHAR2(128)
TABLE_NAME		VARCHAR2(128)
GRANTOR		VARCHAR2(128)
PRIVILEGE		VARCHAR2(40)
GRANTABLE		VARCHAR2(3)
HIERARCHY		VARCHAR2(3)
COMMON		VARCHAR2(3)
TYPE		VARCHAR2(24)
INHERITED		VARCHAR2(3)

figure(4)

ROLE:
décrit les autorisations d'objet dont l'utilisateur actuel est le propriétaire

3)
select username from user_users;
4)
STRUCTURE: USER_ TAB_COLUMNS

describe USER_TAB_COLUMNS; **figure(5)**

```
Nom                                       NULL ?   Type
----------------------------------------- -------- ----------------

TABLE_NAME                                NOT NULL VARCHAR2(128)
COLUMN_NAME                               NOT NULL VARCHAR2(128)
DATA_TYPE                                          VARCHAR2(128)
DATA_TYPE_MOD                                      VARCHAR2(3)
DATA_TYPE_OWNER                                    VARCHAR2(128)
DATA_LENGTH                               NOT NULL NUMBER
DATA_PRECISION                                     NUMBER
DATA_SCALE                                         NUMBER
NULLABLE                                           VARCHAR2(1)
COLUMN_ID                                          NUMBER
DEFAULT_LENGTH                                     NUMBER
DATA_DEFAULT                                       LONG
NUM_DISTINCT                                       NUMBER
LOW_VALUE                                          RAW(2000)
HIGH_VALUE                                         RAW(2000)
DENSITY                                            NUMBER
NUM_NULLS                                          NUMBER
NUM_BUCKETS                                        NUMBER
LAST_ANALYZED                                      DATE
SAMPLE_SIZE                                        NUMBER
CHARACTER_SET_NAME                                 VARCHAR2(44)
CHAR_COL_DECL_LENGTH                               NUMBER
GLOBAL_STATS                                       VARCHAR2(3)
USER_STATS                                         VARCHAR2(3)
AVG_COL_LEN                                        NUMBER
CHAR_LENGTH                                        NUMBER
CHAR_USED                                          VARCHAR2(1)
V80_FMT_IMAGE                                      VARCHAR2(3)
DATA_UPGRADED                                      VARCHAR2(3)
HISTOGRAM                                          VARCHAR2(15)
DEFAULT_ON_NULL                                    VARCHAR2(3)
IDENTITY_COLUMN                                    VARCHAR2(3)
EVALUATION_EDITION                                 VARCHAR2(128)
UNUSABLE_BEFORE                                    VARCHAR2(128)
UNUSABLE_BEGINNING                                 VARCHAR2(128)
COLLATION                                          VARCHAR2(100)
```

figure(5)

<u>COMPARAISON:</u>

la colonne **OWNER** n’existe pas dans USER_TAB_COLUMNS t contrairement a ALL_TAB_COLUMNS

5)

select * from ALL_TABLES where OWNER=’SYSTEM’ and table_name IN (‘CLIENT’,’VEHICULE’,’MARQUE’,’MODELE’,’INTERVENANTS’,’INTERVENTIONS’,’EMPLOYE’);

```
SQL> select table_name from ALL_TABLES where OWNER='SYSTEM' and table_name IN ('
CLIENT','VEHICULE','MARQUE','MODELE','INTERVENANTS','INTERVENTIONS','EMPLOYE');

TABLE_NAME
--------------------------------------------------------------------------------

CLIENT
EMPLOYE
INTERVENANTS
INTERVENTIONS
MARQUE
MODELE
VEHICULE

7 lignes sélectionnées.
```

remarque:
(‘CLIENT’,’VEHICULE’,’MARQUE’,’MODELE’,’INTERVENANTS’,’INTERVENTIONS’,’EMPLOYE’) c’est la liste des tables du TP1 (si on precise on affiche toutes les tables dont le owner est **system**)

on a selectionner l attribut table_name a cause de l’affichage dans le CMD

6)
select table_name, owner from ALL_TABLES where owner in(‘SYSTEM’,’GESTIONINTERV’);

```
SYSTEM

LOGMNR_GT_USER_INCLUDE$
SYSTEM

LOGMNR_GT_XID_INCLUDE$
SYSTEM


TABLE_NAME
--------------------------------------------------------------------------------

OWNER
--------------------------------------------------------------------------------

LOGMNRT_MDDL$
SYSTEM

OL$
SYSTEM

OL$HINTS
SYSTEM
```

7)
select column_name , nullable , data_type , char_col_decl_length from USER_TAB_COLUMNS where table_name in(’CLIENT’ ,’VEHICULE’);

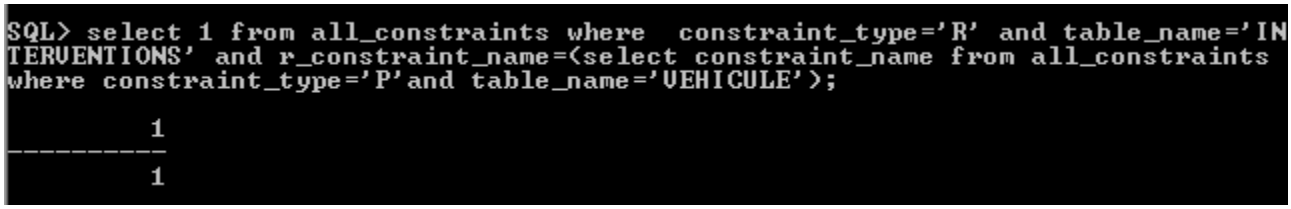
```
COLUMN_NAME
--------------------------------------------------------------------------------

N
-
DATA_TYPE
--------------------------------------------------------------------------------

CHAR_COL_DECL_LENGTH
--------------------
CIV
Y
VARCHAR2
                    6
```

remarque:
- on pouvait faire 2 requete description de **client** et **vehicule** séparément

- column_name = nom de la colonne
- nullable = null ou bien not null
- data_type = type de l'attribut (int ,varchar2, …)
- char_col_decl_length = length (exemple: val varchar2(**100**))

8)

select 1 from all_constraints where constraint_type='R' and table_name='INTERVENTIONS' and r_constraint_name=(select constraint_name from all_constraints where constraint_type='P'and table_name='VEHICULE');



remarque:

a)la partie en rouge : on va extraire le nom de la contrainte PRIMARY KEY
(on a pas preciser le nom de la contrainte donc il va etre générer : SYS_C007363)

b)on va extraire les clé etrangéres de la table all_constraints (type=R) de la table 'INTERVENTIONS'

c) jointure du resultat de (b) avec (a) (r_constraint_name = au resultat de a)

le nombre de 1 en sortie = nombre de resultats trouvés

9)
select table_name , constraint_type from USER_CONSTRAINTS where table_name
IN('CLIENT','VEHICULE','MARQUE','MODELE','INTERVENANTS','INTERVENTIONS','EMPLOYE');

remarque :

c (vérifier la contrainte sur une table)

P (clé primaire)

U (clé unique)

R (intégrité référentielle clé etrangére)

V (avec option de vérification, sur une vue)

O (en lecture seule, sur une vue)

```
SQL> select table_name , constraint_type from USER_CONSTRAINTS where table_name
IN('CLIENT','VEHICULE','MARQUE','MODELE','INTERVENANTS','INTERVENTIONS','EMPLOYE
');

TABLE_NAME
--------------------------------------------------------------------------------

C
-
CLIENT
C

CLIENT
C

CLIENT
P
```

10)

a) select column_name , nullable , data_type , char_col_decl_length from USER_TAB_COLUMNS where table_name=’INTERVENTIONS;

b) select constraint_name , constraint_type , search_condition from USER_CONSTRAINTS where table_name=’INTERVENTIONS;

remarque:
la partie **(a)** nous permet d’avoir : nom de l’attribut , type, si c’est null ou non null, taille
la partie **(b)** nous permet d’avoir les contraintes

11)
select privilege from user_tab_privs where grantee=’GESTIONINTERV’;

```
SQL> select table_name,privilege from user_tab_privs where grantee='GESTIONINTER
V';

TABLE_NAME
------------------------------------------------------------------------------

PRIVILEGE
----------------------------------------
CLIENT
INDEX

CLIENT
SELECT

INTERVENTIONS
SELECT


TABLE_NAME
------------------------------------------------------------------------------

PRIVILEGE
----------------------------------------
CLIENT
UPDATE

INTERVENANTS
UPDATE
```

12)
select granted_role from dba_role_privs where grantee=’GESTIONINTERV’;

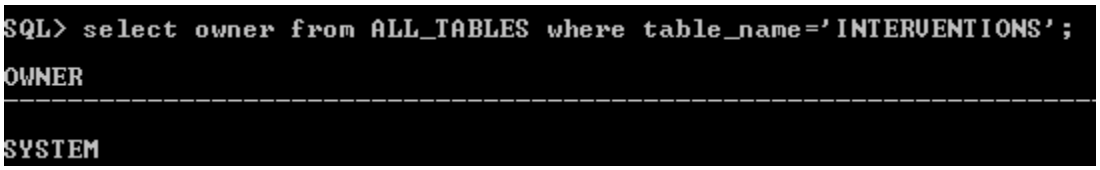
```
SQL> select granted_role from dba_role_privs where grantee='GESTIONINTERV';

GRANTED_ROLE
------------------------------------------------------------------------------

GESTIONAIREINTERV
```

13)
select object_name from ALL_OBJECTS where owner=’GESTIONINTERV’;

```
SQL> select object_name from ALL_OBJECTS where owner='GESTIONINTERV';

aucune ligne súlectionnúe
```

14)

select owner from ALL_TABLES where table_name=’INTERVENTIONS’;



15)

select ROUND(sum(bytes)/1024) from user_segments where segment_type='TABLE' and segment_name = 'INTERVENTIONS';