

**TP-2 : Eléments du dictionnaire Oracle****1. Dictionnaire :**

La vue DICT décrit l'ensemble des vues du dictionnaire de données. A partir des informations contenues dans cette vue, retrouver

- Le nombre total de vues du dictionnaire de données
- Le nombre de vues par catégorie (DBA\_, USER\_, ALL\_, V\$, "autres")
- Lister les noms des vues pour les catégories DBA\_ et V\$
- Lister les noms et objets des vues de la catégorie "autres":

**2. Tablespaces**

A l'aide des vues du dictionnaire de données, retrouver la structure logique de la base orcl.

Vous déterminerez :

- Les tablespaces : nom, état du tablespace ainsi que nom, état des fichiers qui les composent (DBA\_TABLESPACES, DBA\_DATA\_FILES)
- La taille totale en Mo de chaque tablespace. L'attribut bytes donne les tailles en octet (DBA\_DATA\_FILES)
- La taille totale de la base de données (DBA\_DATA\_FILES)
- L'espace disponible de chaque tablespace (DBA\_FREE\_SPACE), les rollbacks segments (nom, état, tablespace de mémorisation) (DBA\_ROLLBACK\_SEGS)
- Les fichiers redo-log (groupe, nom, état, taille) (V\$LOGFILE , V\$LOG)
- Les fichiers de contrôle (nom, état) (V\$Controlfile)
- Les caractéristiques de la SGA (System Global Area) (V\$SGA)

**3. Initialisation**

Dans le fichier d'initialisation, on trouve les paramètres d'initialisation. On peut également trouver ces informations dans la vue v\$parameter. Noms des paramètres dans *name* et leurs valeurs dans *value*.

Dans cette vue, on récupère les informations recherchées.

- Nom de base : db\_name
- Nom du service correspondant : service\_name
- Taille du bloc mémoire : db\_block\_size

**4. USER\_TABLESPACE**

- Afficher les noms de vos tablespaces (USER\_TABLESPACES)
- Créer la table T1(a integer, b varchar2(5)) ; dans le tablespace DATA par exemple.
- Vérifier que la table T1 a bien été créée dans ce tablespace (USER\_TABLES) ;

## 5. Caractéristiques du Tablespace

Afficher les caractéristiques physiques du tablespace dans lequel ont été créées les tables (paramètres de la clause storage et espace disponible). Il faut préciser dans la requête `ALLOCATION_TYPE = USER`

Afficher les caractéristiques physiques des tables créées (initial extent, next\_extent, min\_extents, max\_extents, pct\_increase, pct\_free, pct\_used).

### 6.

A partir de la table DEPT, créer pour une table TEST contenant plusieurs milliers (7168 par exemple) d'enregistrements. Afficher les caractéristiques physiques de la table TEST. Utiliser le `CREATE TABLE AS` pour la création. Ensuite créer le bloc PL/SQL pour augmenter le nombre d'enregistrements de la table TEST à partir de la table DEPT;

### 7.

Afficher le nombre d'extents de tous les segments de votre compte. (nom du segment, type, nombre d'extents, taille en bytes). (`dba_segments`).

### 8.

Afficher la taille de chaque extent de la table TEST. (Le segment TEST dans `dba_extents`). Il faut vérifier que les informations affichées concernent bien votre table TEST et pas toutes les tables TEST de la base "orcl".

### 9.

Modifier les paramètres physiques de la table TEST de manière que son prochain extend ait une taille de 200K et un pctincrease de 0.

### 10.

Afficher les valeurs de paramètres fournissant le nombre d'enregistrements (`num_row`), le nombre de blocs occupés (`blocks`), le nombre de blocs vides (`empty_blocks`), `pct_free`, `pct_used` de la table test à partir de la vue `dba_tables`.

Appliquer la composante ANALYZE à la table TEST de manière à calculer les caractéristiques physiques de cette table ;

Utiliser la commande : **`analyze table test compute statistics;`**

Ré-exécuter la requête précédente.

Modifier le pctfree et le pctused (par exemple 1 et 80 respectivement).

Augmenter la taille de la table test en utilisant le bloc PLSQL de la question 6. Revérifier les caractéristiques de stockage de la table test (n'oublier pas de réappliquer la composante ANALYZE).