

TP 1: DECOUVRIR ORACLE ET SQL*PLUS

OBJECTIFS :

- ↪ Familiarisation avec SGBD et son environnement de travail (SQL*Plus d'Oracle)
- ↪ Ecriture des requêtes SQL simples sur une BD préétablie (HR d'Oracle)

Partie I : Présentation d'Oracle

1. Définition:

a) Base de données:

.....

.....

.....

b) SGBD:.....

.....

.....

c) Fonctionnalités d'un SGBD:

-
-
-
-
-
-
-
-

d) Exemples de SGBD:

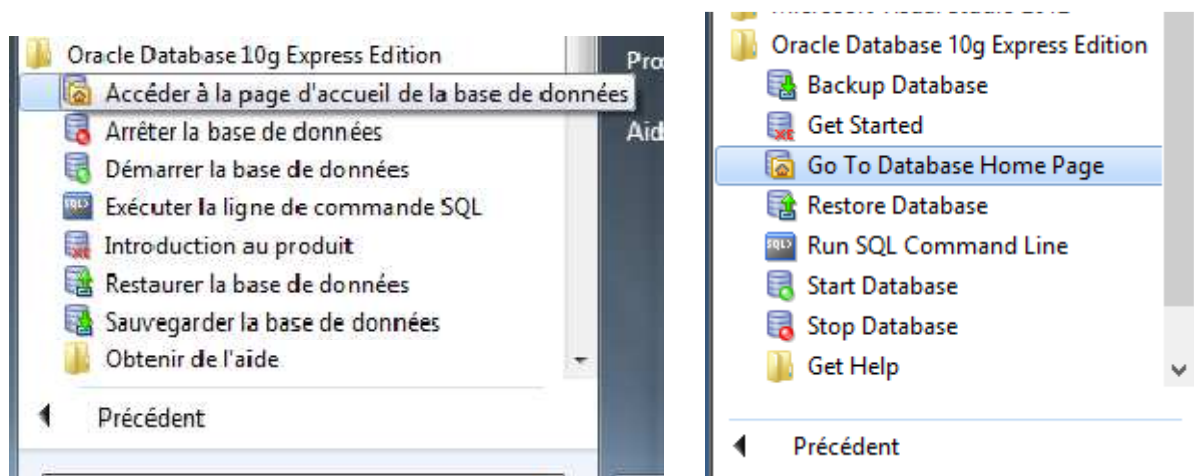
-
-
-
-
-

2. Oracle, présentation et historique:

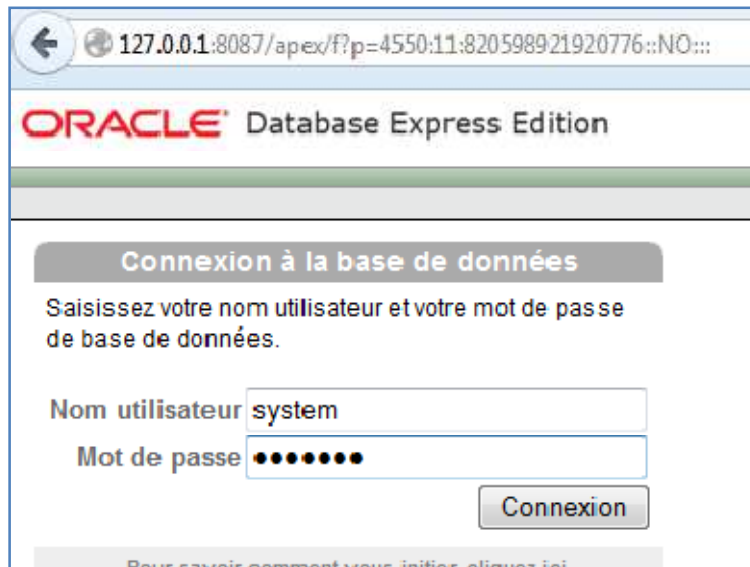
- Oracle Database, est un SGBD relationnel (et Objet à partir de la version 8)
- Dernière version: 12c (2014)
- Disponibilité: Linux, Windows, Unix, MacOSX, Solaris,
- Licence: commerciale (prix élevé), gratuite dans sa version Express.
- Performance: fiable et performant et très riche fonctionnellement.
- Sécurité: Oracle propose une gamme complète de solutions de sécurité garantissant la confidentialité des données, la protection contre les menaces.
- Utilisation: Parmi les SGBD les plus utilisés à l'échèle mondial, il est orienté vers les bases de données volumineuses (>200Go) et un grand nombre d'utilisateurs (>300 utilisateurs).
-
- Historique:
 - 1979 : Oracle 2. Première version commercial. Premier SGBD basé sur le SQL de CODD.
 - 1983 : Oracle 3. Réécrit en C.
 - 1984 : Oracle 4. Gestion des transactions.
 - 1992 : Oracle 7. Contraintes référentielles. Procédures stockées. Triggers.
 - 1997 : Oracle 8. Objet-relationnel.
 - 1998 : Oracle 8i. i pour internet.
 - 2004 : Oracle 10g. g pour grid computing : calcul distribué et gestion de cluster.
 - 2005 : Oracle 10g express édition. Version gratuite de Oracle 10g mais bridée en nombre de processeurs, d'enregistrements (4G0) et de mémoire (1G0). L'objectif est d'entrer sur le marché des PME sensibles aux coûts et aux promesses de l'Open Source.
 - 2007 : Oracle 11g. Amélioration de la version 10g en termes de performance et de facilité d'administration.
 - 2009 : Oracle achète Sun (Java) qui avait acheté MySQL en 2008.
 - juillet 2013, sortie de Oracle Database 12c (c pour le cloud).

Partie II : Découvrir Oracle et SQL*Plus

Après installation d'Oracle client 10g express édition.



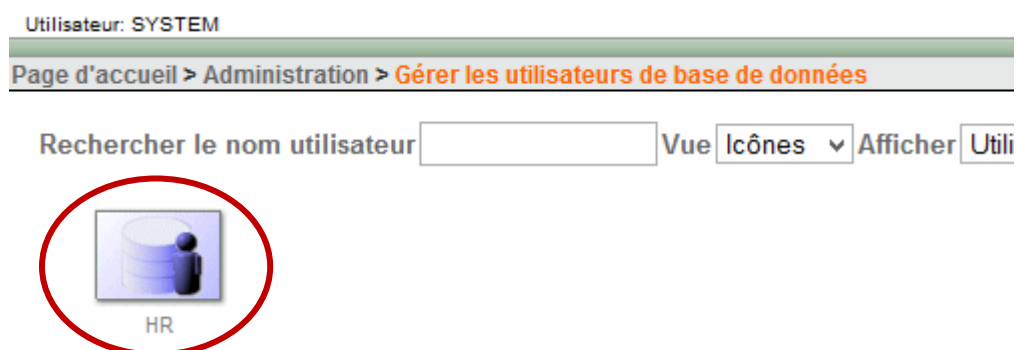
1. Cliquer sur « Accéder à la page d'accueil de la base de données » du menu démarrer puis entrer le nom d'utilisateur système et le mot de passe suivants :
 - Nom utilisateur : **system**
 - Mot de passe : **manager**

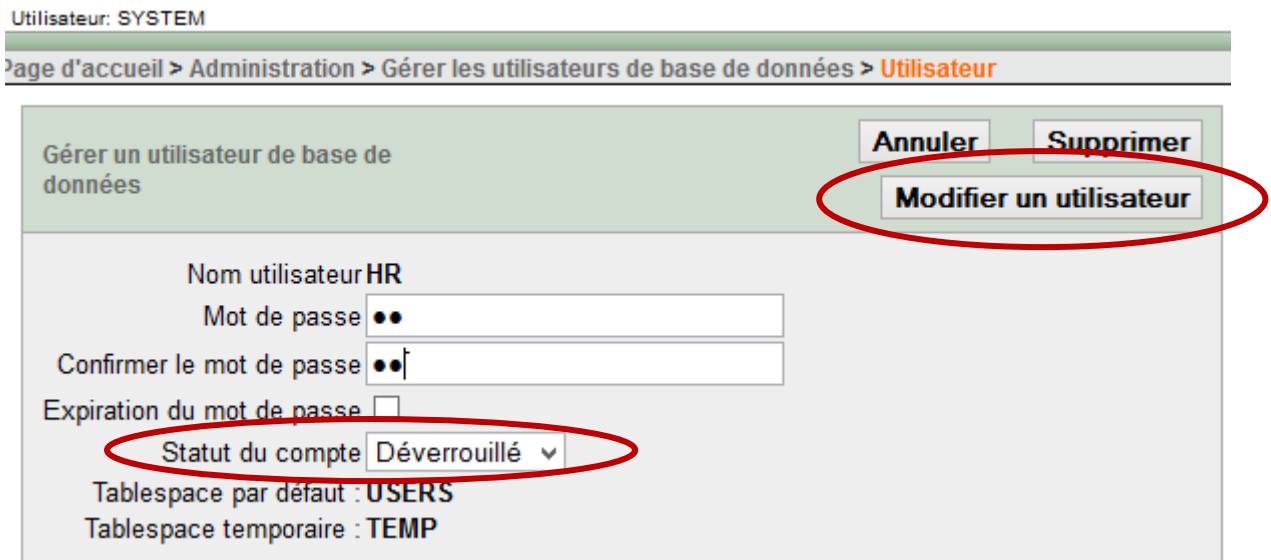


2. Gérer la base HR en cliquant sur Administration / Utilisateurs de base de données / Gérer les utilisateurs:

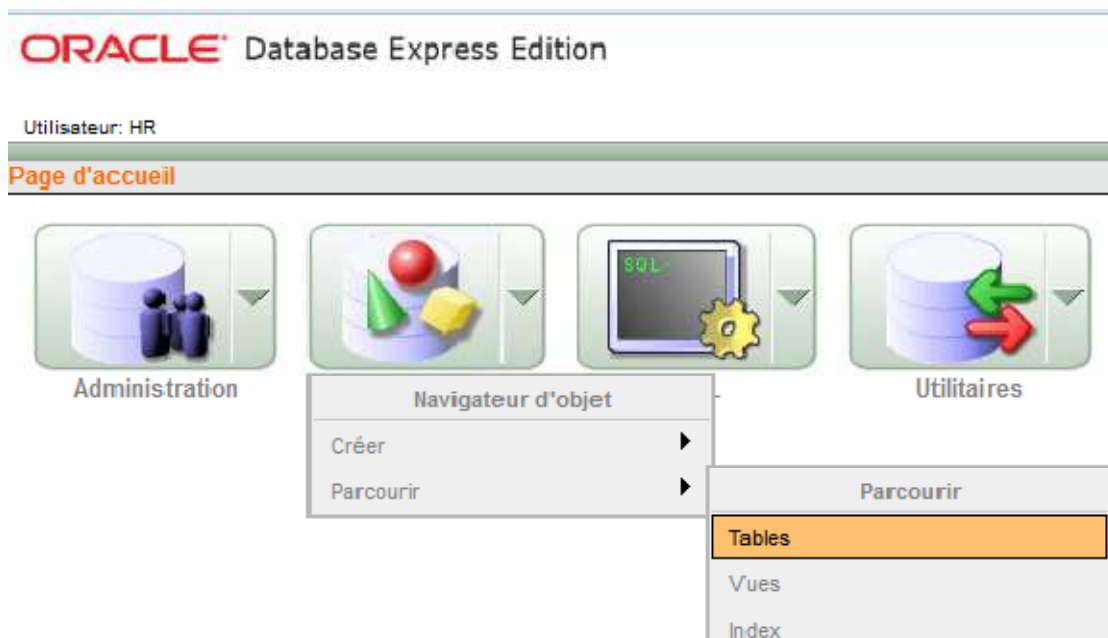


3. Cliquer sur HR ajouter le mot de passe HR puis déverrouiller cette base.

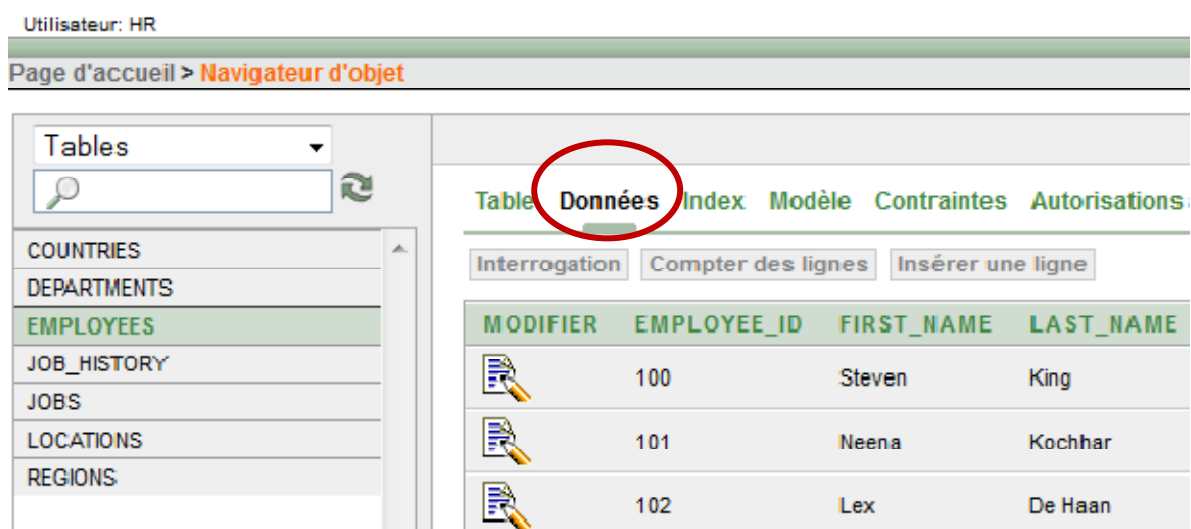




4. Cliquer sur Navigateur d'objet / Parcourir / Table pour visualiser les différentes tables:

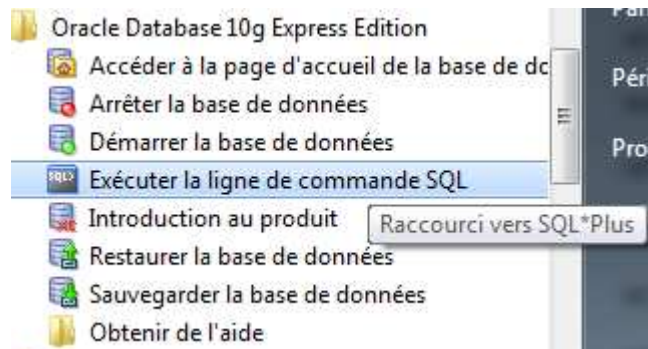


La table EMPLOYEES par exemple pour voir son contenu:



Partie III: Tester quelques commandes SQL:

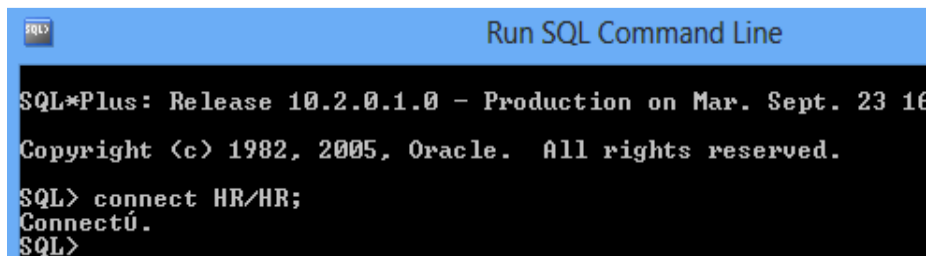
1. Cliquer maintenant sur « Exécuter la ligne de commande SQL » du menu démarrer. A partir de cette ligne établir une connexion à la base HR.



2. Exécuter les commandes suivantes:

- Connexion à la base:

```
SQL>connect HR/HR
```



- Afficher les différentes tables de cette base avec la commande :

```
SQL>Select * from tab ;
```

- Afficher le contenu de chaque table de cette base avec la commande :

```
SQL>Select * from le_nom_de_la_table ;
```

Exemple:

```
SQL>Select * from EMPLOYEES ;
```

- Décrire la structure de chaque table de cette base avec la commande :

```
SQL>Desc le_nom_de_la_table ;
```

3. Répéter les mêmes commandes en utilisant l'outil de "Commandes SQL" de la page d'accueil:

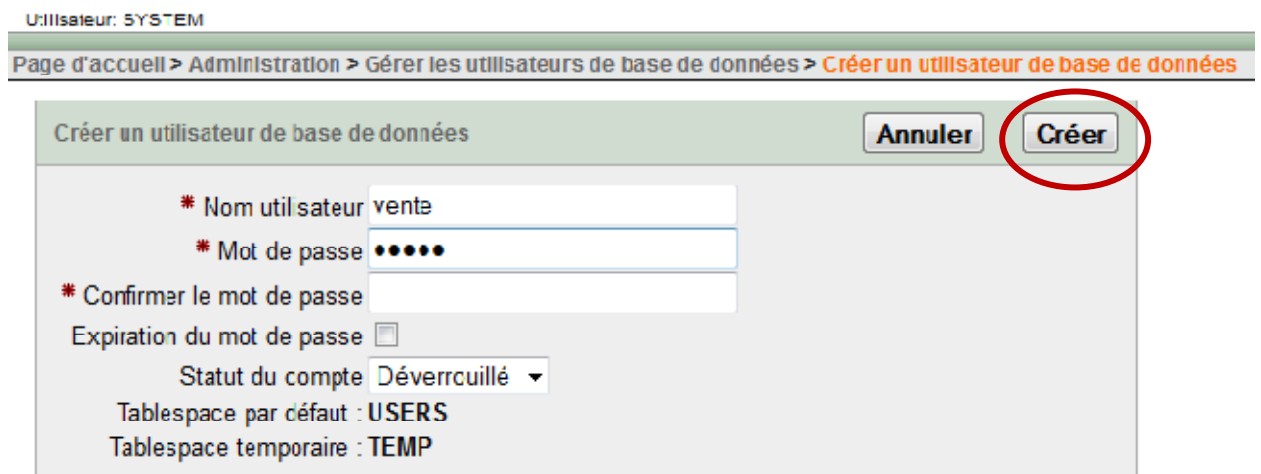


Partie IV : Gestion d'utilisateur

1. Revenant maintenant à l'interface graphique d'Oracle client 10g express édition. Etablir une connexion avec l'utilisateur System / manager. Cliquer sur Administration / Créer un utilisateur pour créer l'utilisateur **vente** avec le mot de passe **vente** :



2. Valider ensuite en cliquant sur le bouton Créer



3. Vous pouvez entrer la ligne suivante à partir de l'invite SQL> pour créer l'utilisateur vente avec le mot de passe vente :

```
SQL>Grant connect, resource to vente identified by vente;
SQL>Commit;
```

4. Connectez-vous maintenant avec le nom d'utilisateur et le mot de passe suivants:

```
Nom utilisateur : vente
Mot de passe : vente
```

En saisissant la ligne suivante à partir de l'invite SQL>

```
SQL>connect vente/vente ;
```

Partie V : Création d'une base de données

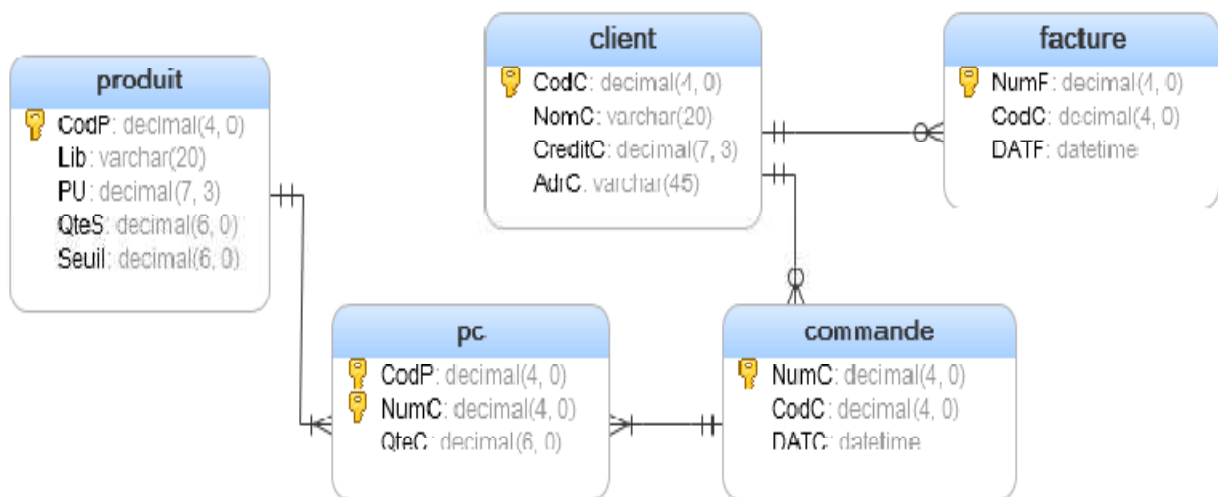
- On désire créer une base de données nommée Vente dont le schéma relationnel est le suivant :

Produit (CodP, Lib, PU, QteS, Seuil)
Client (CodC, NomC, CreditC, AdrC)
Commande (NumC, DatC, #CodC)
Facture (NumF, MontF, DatF, #CodC)
PC (#CodP, #NumC, QteC)

CodC : représente le code client

NumC : représente le numéro de la commande

La représentation des tables est comme suit :



- Exécuter les commandes suivantes à partir de l'invite SQL> afin de créer les tables et les contraintes sur les clés étrangères (Respecter la syntaxe de chaque commande)

```

-- =====
-- Script pour la création de la base
-- Nom de la base : VenteC
-- Nom de SGBD : ORACLE version 10g
-- Date de création : 16/09/2014 11:15
-- =====
-- Table : Prodit
-- =====
create table Produit
(
    CodP NUMBER(4) not null,
    Lib VARCHAR2(20),
    PU NUMBER(8,3),
    QteS NUMBER(6),
    Seuil NUMBER(4),
    constraint cp_CodP primary key (CodP)
);
    
```

```
-- =====
-- Table : Client
-- =====
create table Client
(
  CodC NUMBER(4) not null,
  NomC VARCHAR2(20),
  CreditC NUMBER(8,3),
  AdresseC VARCHAR2(15),
  constraint cp_CodC primary key (CodC)
);
-- =====
-- Table : Commande
-- Avec une clé étrangère CodC
-- =====
create table Commande
(
  NumC NUMBER(4) not null,
  CodC NUMBER(4) not null,
  MontC NUMBER(7,3),
  DATC DATE,
  constraint cp_NumC primary key (NumC)
);
-- =====
-- Table : Facture
-- Avec une clé étrangère CodC
-- =====
create table Facture
(
  NumF NUMBER(4) not null,
  CodC NUMBER(4) not null,
  MontF NUMBER(8,3),
  DATF DATE,
  constraint cp_NumF primary key (NumF)
);
-- =====
-- Table : PC
-- =====
create table PC
(
  CodP NUMBER(4) not null,
  NumC NUMBER(4) not null,
  QteC NUMBER(4),
  constraint cp_PC primary key (CodP, NumC)
);
```

3. Vérifier la création de ces tables avec la commande :

```
Select * from tab ;
```

4. Afficher la description de la table Produit :

```
SQL>Desc Produit;
```