

Université de Valenciennes et du Hainaut-Cambrésis

ISTV

Année universitaire 2011-2012



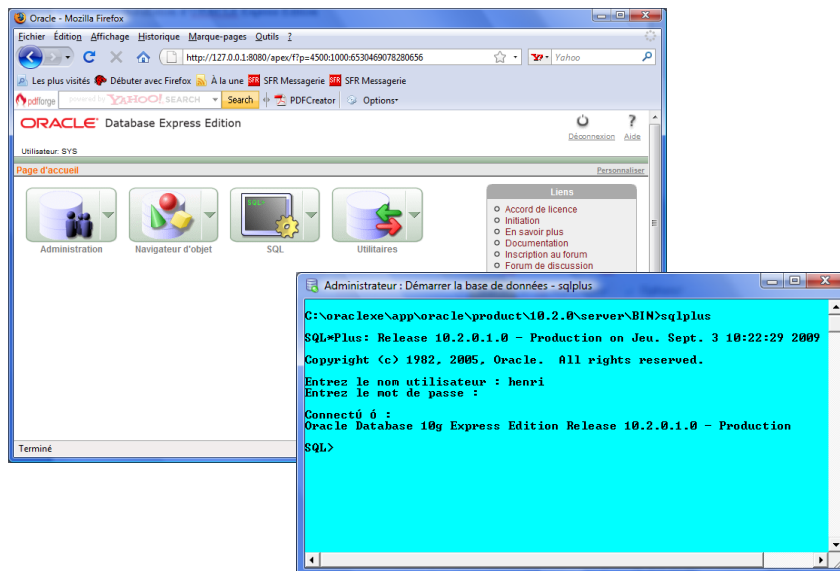
TRAVAUX PRATIQUES

Conception de bases de données

ORACLE Database Express Edition

Henri TSOUNGUI

Installation sous windows et Linux
Connexion au serveur
Manipulation d'objets de la BDD
Gestion des rôles et privilèges
Travaux pratiques



INSTALLATION SOUS LINUX ubuntu 10.10

Oracle XE 10g R2 sous Linux Ubuntu

Oracle Express Edition 10g R2 est la version allégée gratuite d'Oracle. Vous pouvez la télécharger ici (il vous faudra créer un compte) :

<http://www.oracle.com/technetwork/database/express-edition/downloads/index.html>

ou ici :

<http://www.oracle.com/technology/software/products/database/xe/htdocs/102xelinsoft.html>.

Pour cette version Express, vous disposerez d'une seule instance, une seule base de données dont l'identificateur **SID** (*Oracle System ID*) est par défaut « XE ».

Ressources nécessaires

Oracle a besoin de beaucoup de ressources : au moins 512 Mo de RAM et une SWAP de 1024 Mo, 1,5 Gio d'espace disque libre.

Installation

Paquets nécessaires (bibliothèques)

- **libaio1** - kernel aio access library
- **bc** - The GNU bc arbitrary precision calculator language

Utilitaires d'installation

- dpkg

Bon bah là, c'est le régal d'installer un serveur, une ligne de commande et voilà !

```
dpkg -i oracle-xe-universal_10.2.0.1-0_i386.deb
```

Si vous avez une architecture 64 bits, vous pouvez forcer l'installation avec

```
sudo dpkg --force-architecture -i oracle-xe-universal_10.2.0.1-0_i386.deb
```

- apt-get

Ajouter le dépôt de paquet suivant (ligne à ajouter dans le fichier **/etc/apt/sources.list**) :

```
deb http://oss.oracle.com/debian unstable main non-free
```

Si vous avez un problème de clé lors de "apt-get update", il faut télécharger et importer la clé du dépôt Oracle la commande en ligne :

```
wget http://oss.oracle.com/el4/RPM-GPG-KEY-oracle -O- | sudo apt-key add -
```

Mettez à jour la liste des paquets des dépôts et [installez les trois paquets](#) suivants :

- **oracle-xe** - Oracle Database 10g Express Western European Edition
- **oracle-xe-client** - Oracle Client 10g Express Edition

- **oracle-xe-universal** - Oracle Database 10g Express Universal Edition

Mise à jour des sources d'installation :

```
sudo apt-get update
```

Installation des paquets précédemment présentés :

```
sudo apt-get install oracle-xe
OU
sudo apt-get install oracle-xe-universal
ET
sudo apt-get install oracle-xe-client
```

Swap insuffisant

Si lors de l'installation du paquet, un message du style :

This system does not meet the minimum requirements for swap space

Lors de l'installation via Synaptic ou Adept un message d'erreur générique apparaît, et non pas le message ci-avant.

Et qu'il vous est impossible de re-partitionner le swap, il est possible de créer un fichier de swap comme suit :

```
sudo dd if=/dev/zero of=/swapfile bs=1M count=100
```

Où « count » est le nombre de Mo à rajouter à votre swap existant pour arriver au 1006 Mo requis par Oracle XE.

Puis mise au format swap :

```
sudo mkswap /swapfile
```

Enfin activation :

```
sudo swapon /swapfile
```

Source : Oracle.com

insuffisant diskpace

Si lors de l'installation du paquet, un message du style :

You have insufficient diskpace in the destination directory (/usr/lib) to install

Oracle Database 10g Express Edition. The install requires at lease 1.5 GB free on this disk

créer un lien symbolique dans le dossier d'installation /usr/lib dans un endroit ou il y'a de l'espace ex: /home/dossier_personnel

```
sudo ln -s /home/dossier_personnel/oracle oracle
```

Configuration

Paramétrage de l'installation

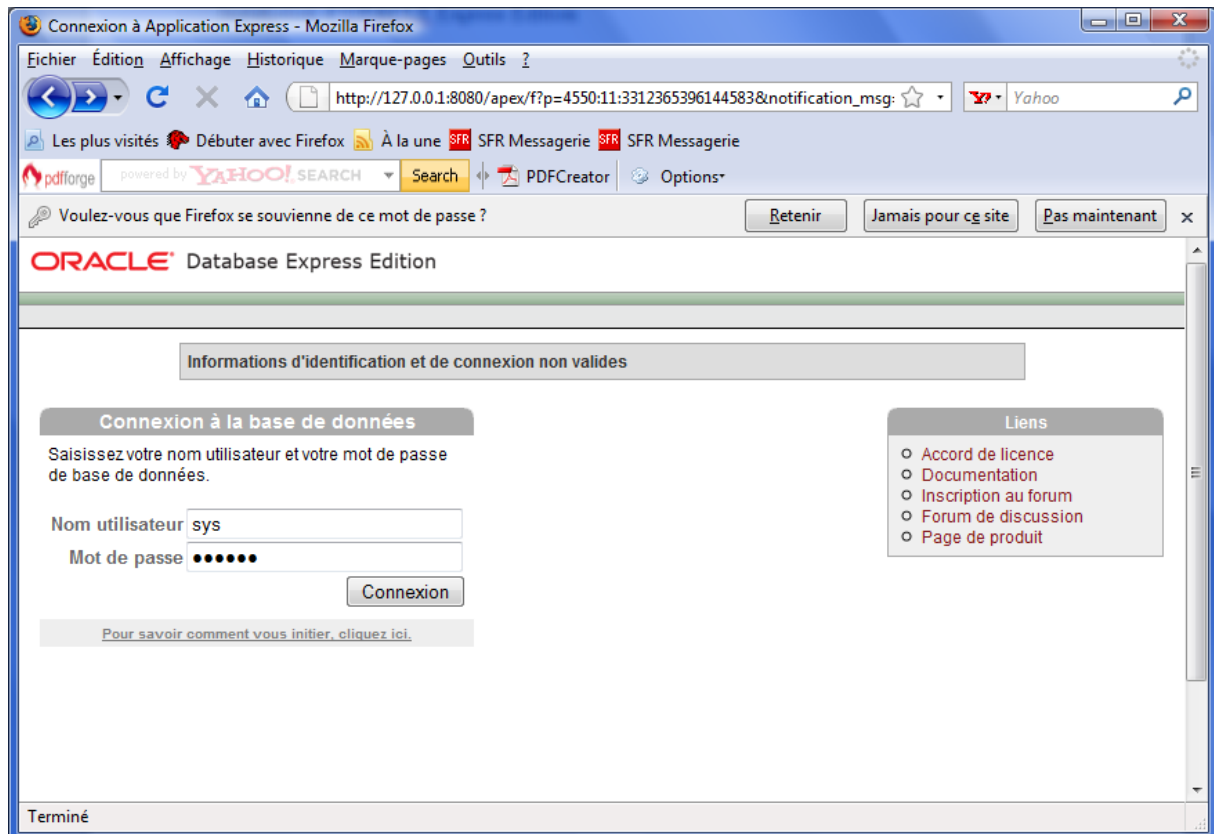
Pour la configuration, lancer ce script :

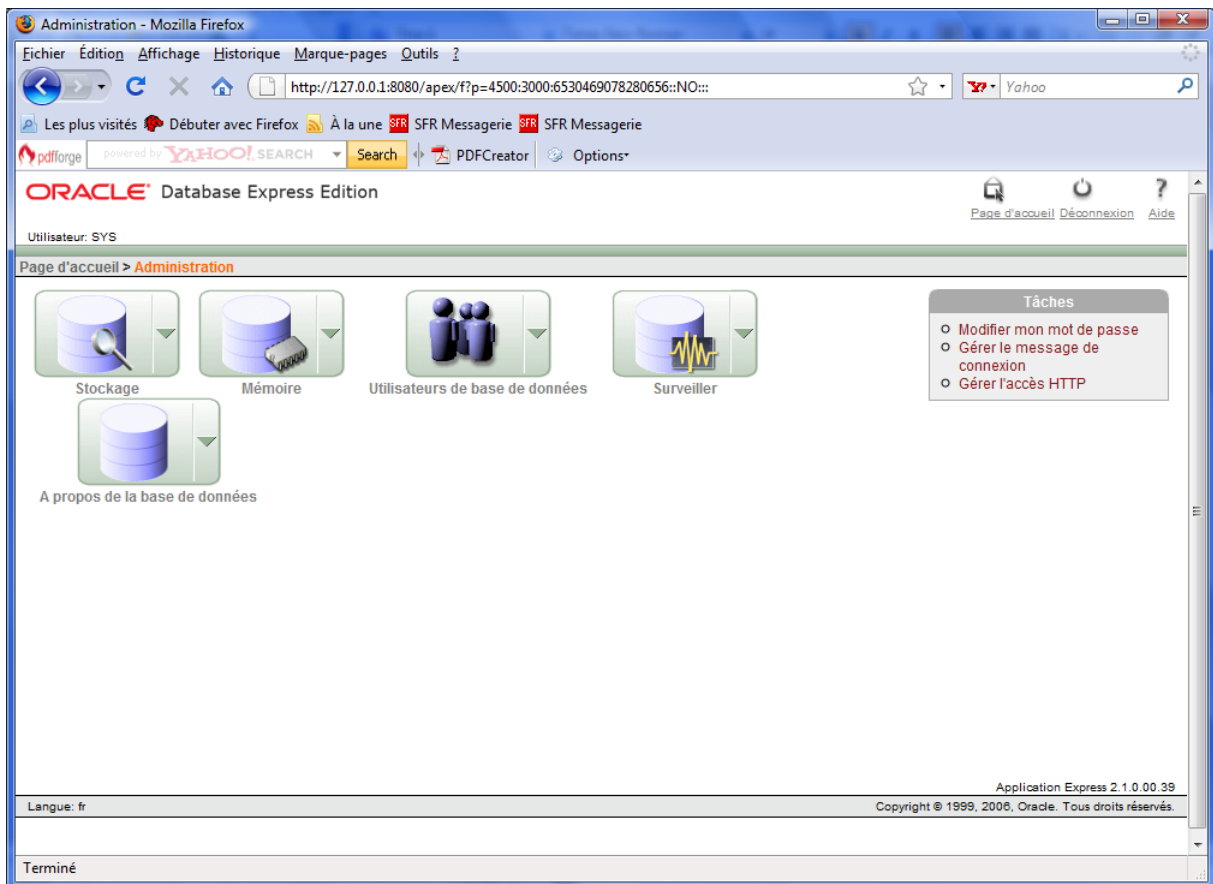
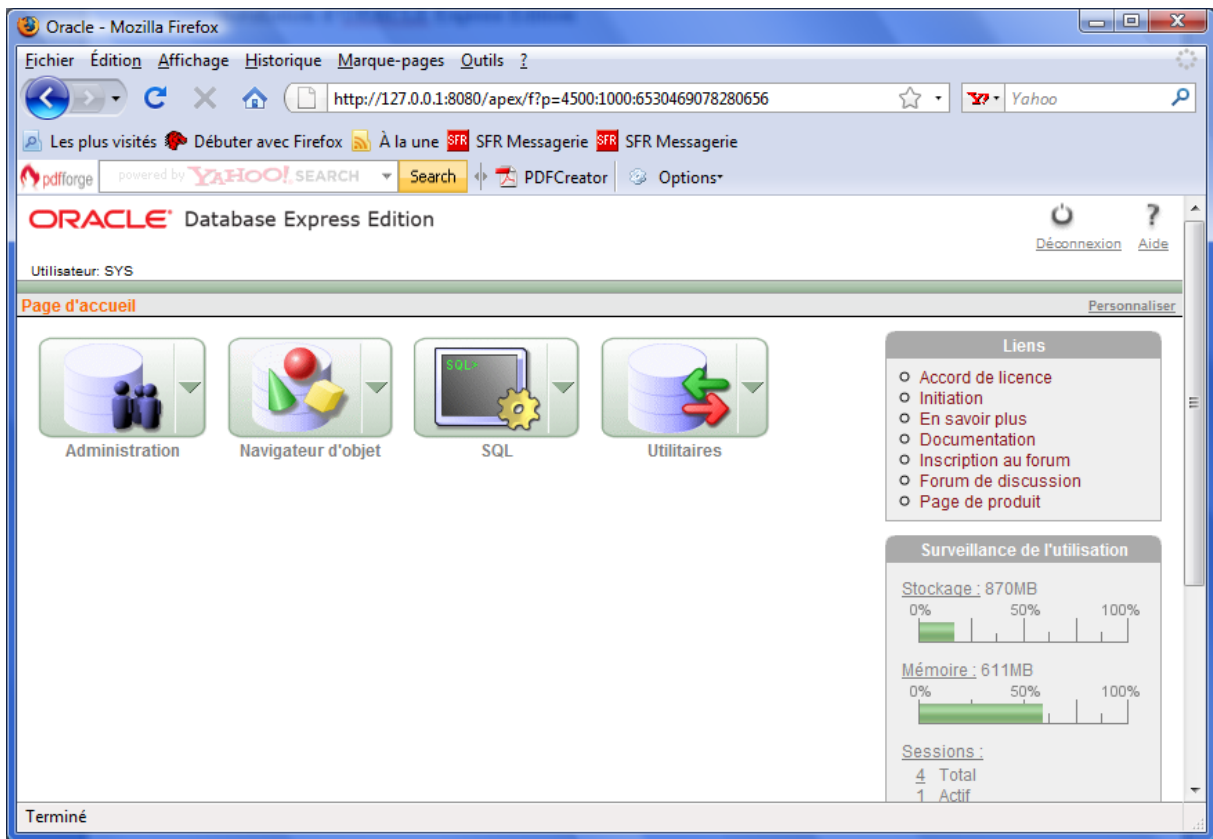
```
/etc/init.d/oracle-xe configure
```

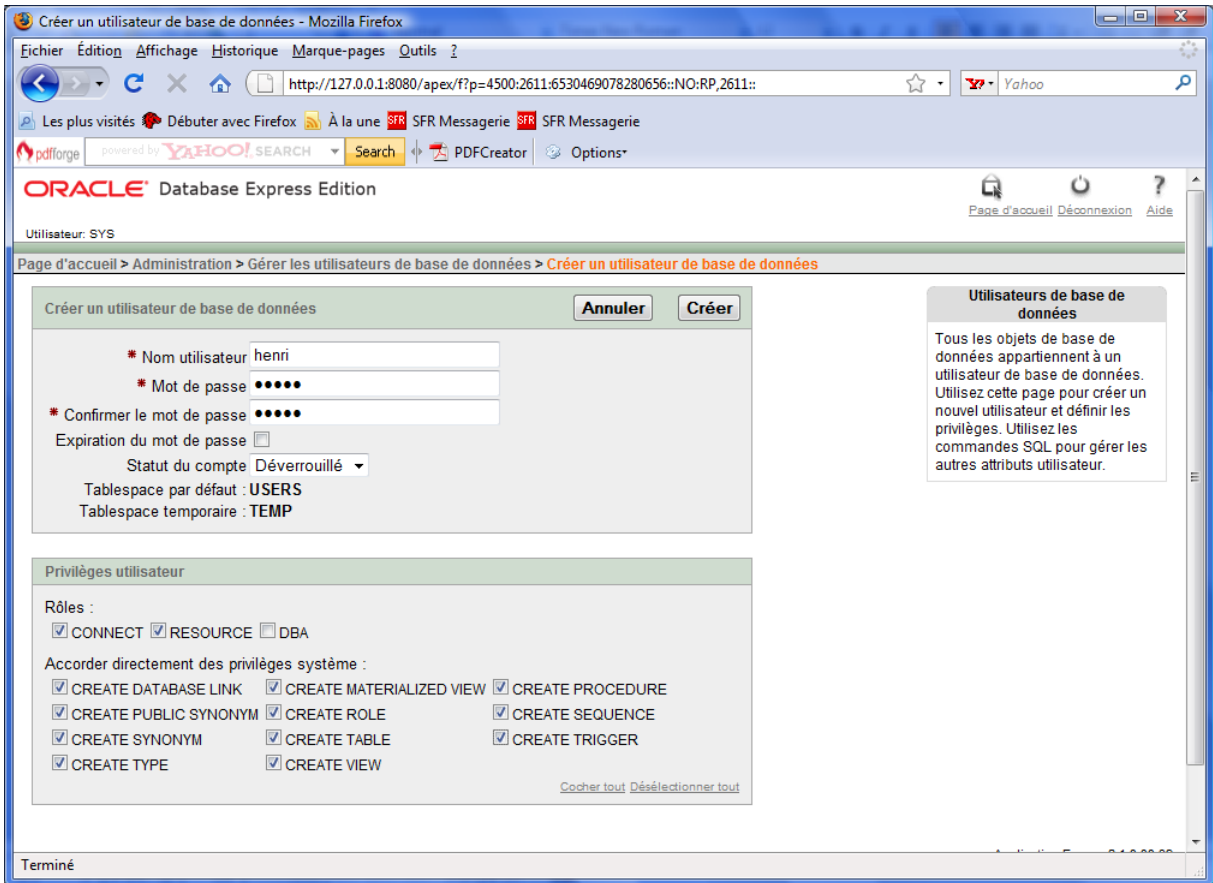
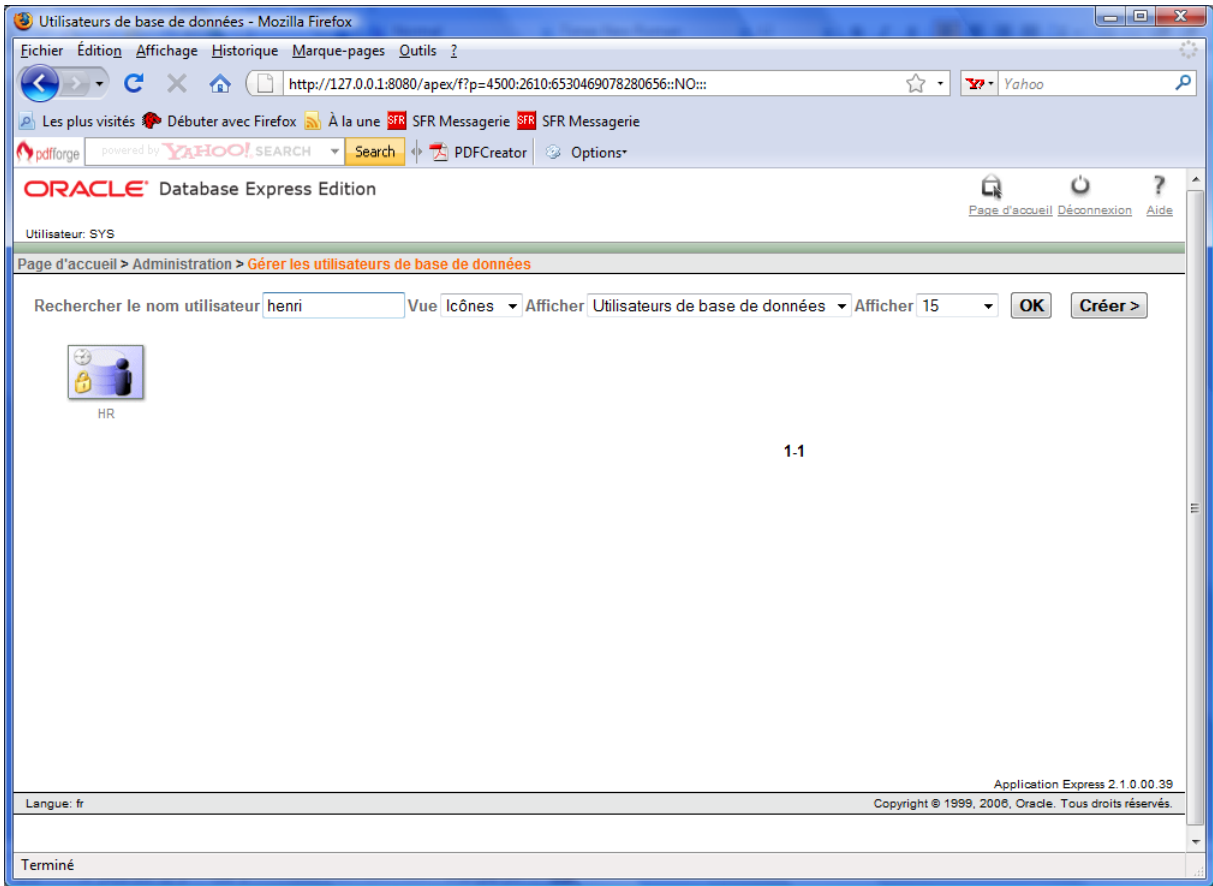
Il vous demandera les mots de passes SYS et SYSTEM, les ports TCP de Oracle listener et de l'application serveur d'Oracle. Dans la plupart des cas, vous devez laisser libre les ports standards.

Maintenant pour créer users / schema / ... Vous avez une jolie interface http pour cela, dans un navigateur, tapez :

<http://localhost:8080/apex/>







Utilisateurs de base de données - Mozilla Firefox

Echier Édition Affichage Historique Marque-pages Outils ?

http://127.0.0.1:8080/apex/f?p=4500:2610:6530469078280656::NO:RP,2610::&success_msg=Utilis

Les plus visités Débuter avec Firefox À la une SFR SFR Messagerie SFR SFR Messagerie

pdfforge powered by YAHOO! SEARCH Search PDFCreator Options

Page d'accueil > Administration > Gérer les utilisateurs de base de données

Utilisateur créé.

Rechercher le nom utilisateur Vue Icônes Afficher Utilisateurs de base de données Afficher 15 OK Créer >

HENRI HR

1-2

Application Express 2.1.0.00.39
Langue: fr Copyright © 1999, 2006, Oracle. Tous droits réservés.

Terminé

Première connexion de l'utilisateur Henri

Connexion à Application Express - Mozilla Firefox

Echier Édition Affichage Historique Marque-pages Outils ?

http://127.0.0.1:8080/apex/f?p=4550:11:1599531855598715::NO::

Les plus visités Débuter avec Firefox À la une SFR SFR Messagerie SFR SFR Messagerie

pdfforge powered by YAHOO! SEARCH Search PDFCreator Options

ORACLE Database Express Edition

Connexion à la base de données

Saisissez votre nom utilisateur et votre mot de passe de base de données.

Nom utilisateur henri

Mot de passe ●●●●

Connexion

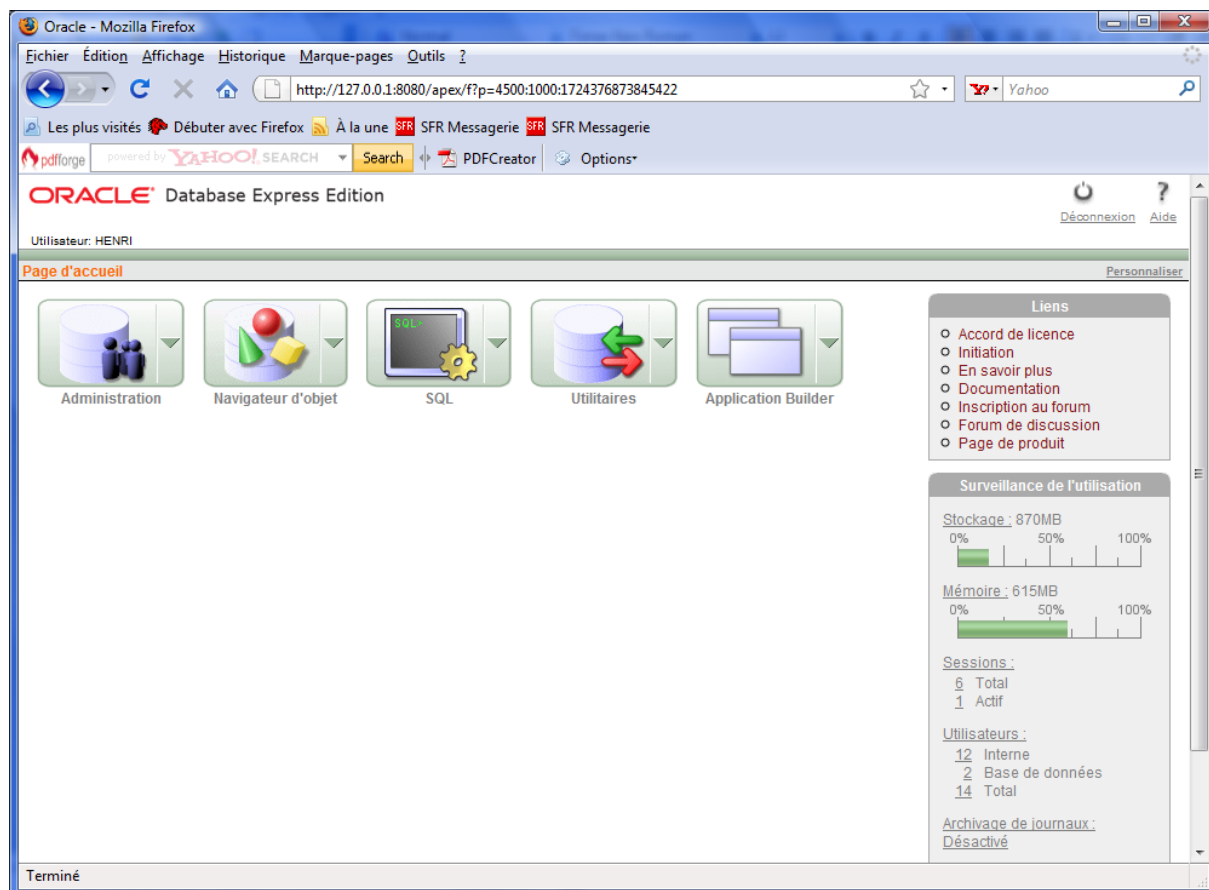
Pour savoir comment vous initier, cliquez ici.

Liens

- Accord de licence
- Documentation
- Inscription au forum
- Forum de discussion
- Page de produit

Application Express
Langue: fr Copyright © 1999, 2006, Oracle. Tous droits réservés.

Terminé



Vous avez réussi à vous connecter, c'est très bien. Nous allons quitter cette belle interface graphique et revenir aux fondamentaux, à savoir le langage SQL proprement dit. La suite va vous obliger à travailler en ligne de commande exclusivement dans SQLPLUS. Nous reviendrons plus tard dans ce client graphique.

ORACLE SQLPLUS

SQLPLUS est l'interface ligne de commande pour Oracle. Son utilisation est vivement conseillée pour maîtriser le langage de requêtes SQL.

Après l'installation d'Oracle-xe sous windows, SQLPLUS est disponible sous le shell DOS.

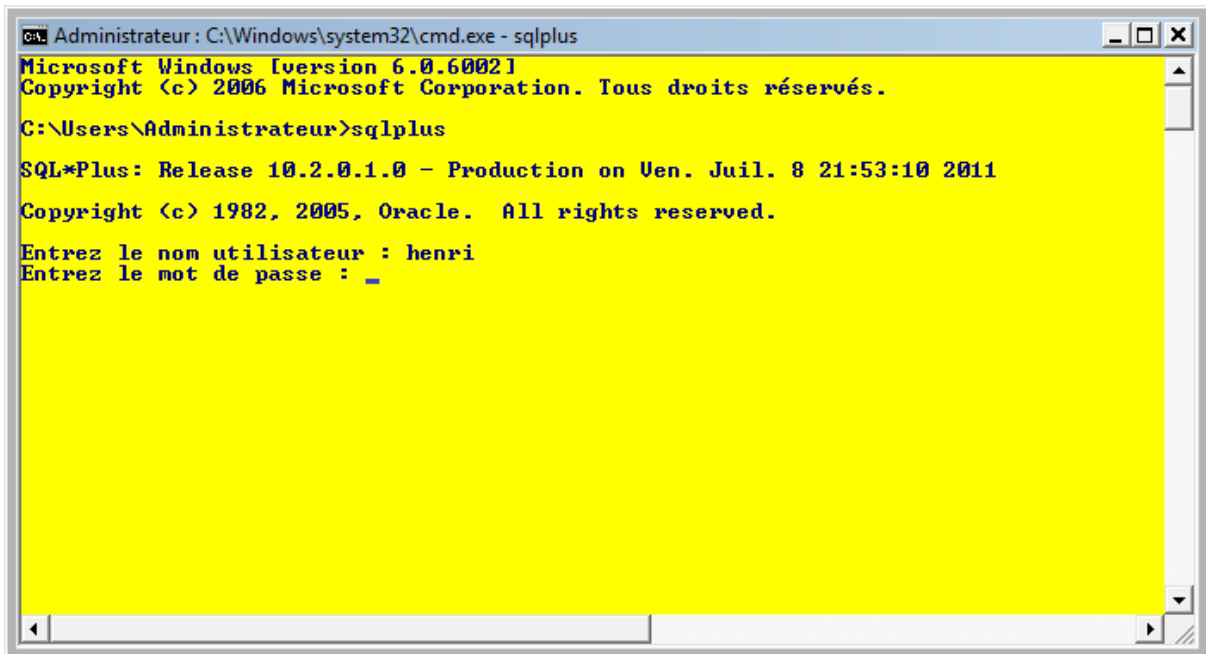
Pour la mise en œuvre de SQLPLUS sous Linux, il faut une reconfiguration particulière que je ne détaillerai pas dans ce document d'introduction à Oracle.

1.Première connexion au serveur

Après l'installation d'Oracle-XE, vous avez créé au moins un compte. Pour vous connecter

-Dans le menu Démarrer de Windows puis Exécuter, tapez SQLPLUS .

Donnez ensuite votre login et le mot de passe de l'utilisateur.



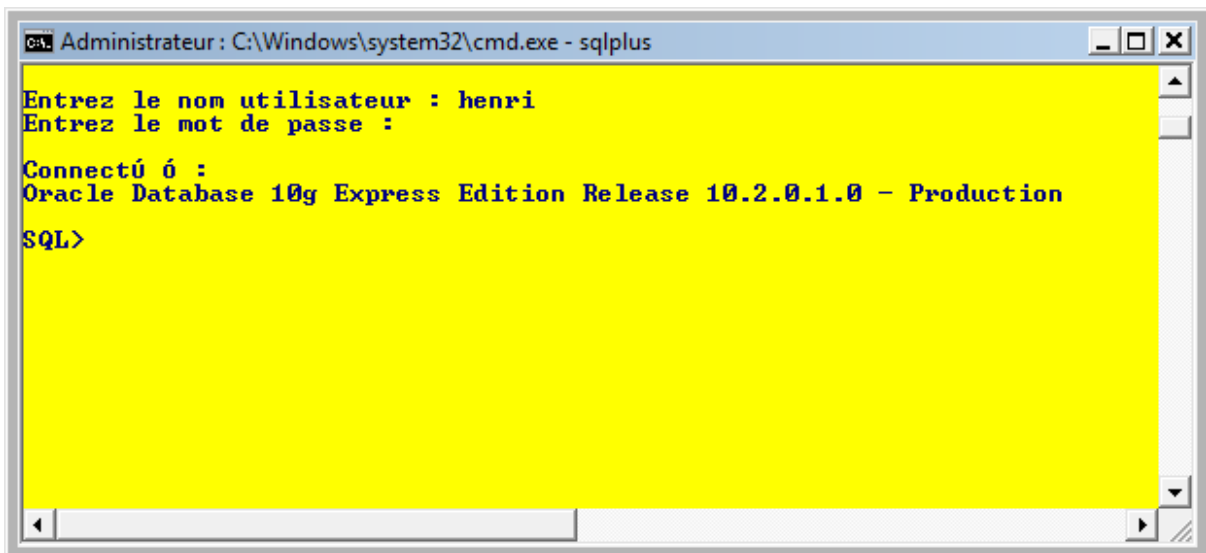
```
Administrateur : C:\Windows\system32\cmd.exe - sqlplus
Microsoft Windows [version 6.0.6002]
Copyright (c) 2006 Microsoft Corporation. Tous droits réservés.

C:\Users\Administrateur>sqlplus

SQL*Plus: Release 10.2.0.1.0 - Production on Ven. Juil. 8 21:53:10 2011
Copyright (c) 1982, 2005, Oracle. All rights reserved.

Entrez le nom utilisateur : henri
Entrez le mot de passe : _
```

Résultat de la procédure de connexion :



```
Administrateur : C:\Windows\system32\cmd.exe - sqlplus

Entrez le nom utilisateur : henri
Entrez le mot de passe :

Connecté à :
Oracle Database 10g Express Edition Release 10.2.0.1.0 - Production
SQL>
```

Vous avez obtenu le prompt d'Oracle, vous êtes désormais connecté et pouvez travailler en SQL.

Nous allons commencer par visualiser la liste de toutes les tables présentes sur le serveur. Pour cela, il suffit de taper la commande SQL donnant le nom des tables(table_name) de l'instance.

```

C:\Administrateur : C:\Windows\system32\cmd.exe - sqlplus

SQL> select table_name from tabs;

TABLE_NAME
-----
DEPT
TYPELOG
COMMODITE
USAGER
LOGEMENT
DISPOSER

6 ligne(s) sélectionné(s).

SQL>

```

2. Manipulation des objets

Création d'une table CLIENT (cliNum A(3), clé primaire, A : alphanumérique
cliNom A(15),
cliPrenom A(15),
cliAdresse A(25),
cliCpostal A(5),
cliVille A(20)),
cliChiffre N(8,2)); N : Numérique/Décimal(longueur, décimales).

Commandes de création de la table et visualisation de la structure (DESCription).

```

C:\Administrateur : C:\Windows\system32\cmd.exe - sqlplus

SQL> create table CLIENT (cliNum char(3) not null,cliNom varchar2(15),cli
har2(20),cliChiffre numeric(8,2));

Table créée.

SQL> DESC CLIENT;

Nom                                NULL ?    Type
-----
CLINUM                             NOT NULL  CHAR(3)
CLINOM                             VARCHAR2(15)
CLIPRENOM                           VARCHAR2(15)
CLIADRESSE                           VARCHAR2(25)
CLICPOSTAL                           CHAR(5)
CLIVILLE                            VARCHAR2(20)
CLICHIFFRE                           NUMBER(8,2)

SQL> _

```

-Insertion des enregistrements (records).

```

SQL> insert into CLIENT values('E32','DARIUS','Pierre','4 rue Monge','59000','Lille',5432.89);
1 ligne créée.

```

```

SQL> insert into CLIENT values('H67','MOLO','Honoré','63 place d Italie','59300','Valenciennes',1843.45);
1 ligne créée.

```

Création d'un fichier de commandes d'insertion appelé *insert.sql*

```
SQL>edit insert.sql
```

```
insert into CLIENT values('V56','NGUEMA','Elias','87 rue de la Paix','80020','Amiens',654.94);
```

```
insert into CLIENT values('B90','TROUILLET','Blaise','5 av. Victor Hugo','59233','Maing',3987.78);
insert into CLIENT values('K79','LEROY','Pascal','3 rue de la Digue','13002','Marseille',1856.37);
```

-Exécution du fichier de commandes
 SQL>R insert.sql ou simplement
 SQL>R insert (RUN insert)

-Visualisation des données

```
Administrateur : C:\Windows\system32\cmd.exe - sqlplus
K79 LEROY                Marseille
SQL> select clinum,clinom,clicpostal,cliville from client;
CLI CLINOM                CLICP CLIVILLE
-----
E32 DARIUS                59000 Lille
H67 MOLO                  59300 Valenciennes
U56 NGUEMA                80020 Amiens
B90 TROUILLET            59233 Maing
K79 LEROY                13002 Marseille
```

```
Administrateur : C:\Windows\system32\cmd.exe - sqlplus
SQL> select cliCpostal CODE, count(*) NOMBRE from client group by clicpostal order by count(*);
CODE      NOMBRE
-----
59000     1
59233     1
80020     1
13002     1
59300     2

SQL> select cliCpostal CODE, count(*) NOMBRE from client group by clicpostal order by count(*) desc;
CODE      NOMBRE
-----
59300     2
59233     1
13002     1
```

On va maintenant effectuer quelques modifications de structure, ajouter des contraintes à la table CLIENT. Nous allons VOLONTAIREMENT commettre quelques erreurs communes et les corriger dans les requêtes. Ceci vous permettra d'avoir un exemple plus réaliste des traitements. L'inter-activité est nécessaire en ligne de commande et il faut savoir interpréter les messages de SQLPLUS.

```
Administrateur : C:\Windows\system32\cmd.exe - sqlplus
SQL> desc client;
Nom                                NULL ?   Type
-----
CLINUM                             NOT NULL CHAR(3)
CLINOM                             UARCHAR2(15)
CLIPRENOM                           UARCHAR2(15)
CLIADRESSE                           UARCHAR2(25)
CLICPOSTAL                           CHAR(5)
CLIUILLE                             UARCHAR2(20)
CLICHIFFRE                           NUMBER(8,2)

SQL>
SQL> alter table client add column cliRegion char(3)not null;
alter table client add column cliRegion char(3)not null
*
ERREUR ó la ligne 1 :
ORA-00904: : identificateur non valide

SQL> alter table client column add cliRegion char(3)not null;
alter table client column add cliRegion char(3)not null
*
ERREUR ó la ligne 1 :
ORA-00933: la commande SQL ne se termine pas correctement

SQL> alter table client add cliRegion char(3)not null;
Table modifiúe.

SQL> desc client;
Nom                                NULL ?   Type
-----
CLINUM                             NOT NULL CHAR(3)
CLINOM                             UARCHAR2(15)
CLIPRENOM                           UARCHAR2(15)
CLIADRESSE                           UARCHAR2(25)
CLICPOSTAL                           CHAR(5)
CLIUILLE                             UARCHAR2(20)
CLICHIFFRE                           NUMBER(8,2)
CLIREGION                             NOT NULL CHAR(3)

SQL> alter table client modify cliRegion char(2) not null;
alter table client modify cliRegion char(2) not null
*
ERREUR ó la ligne 1 :
ORA-01442: colonne ó modifier en NOT NULL est dújé NOT NULL

SQL> alter table client modify cliRegion char(2);
Table modifiúe.
```

Nous souhaitons créer une autre table REGION liée à la précédente sur la clé étrangère *cliRegion* qui permettra la vérification de l'intégrité référentielle sur la table des régions.

```

Administrateur: C:\Windows\system32\cmd.exe - sqlplus
SQL> desc client;
Nom
-----
CLINUM          NOT NULL  CHAR(3)
CLINOM          VARCHAR2(15)
CLIPRENOM       VARCHAR2(15)
CLIADRESSE      VARCHAR2(25)
CLICPOSTAL      CHAR(5)
CLIVILLE       VARCHAR2(20)
CLICHIFFRE      NUMBER(8,2)
CLIREGION       NOT NULL  CHAR(2)

SQL> create table REGION(regCode char(2) not null, regNom varchar(15));
Table cr  e.

SQL> alter table client constraint PK_client primary key (cliNum);
alter table client constraint PK_client primary key (cliNum)
*
ERREUR   la ligne 1 :
ORA-01735: option ALTER TABLE non valide

SQL> alter table client add constraint PK_client primary key (cliNum);
alter table client add constraint PK_client primary key (cliNum)
*
ERREUR   la ligne 1 :
ORA-02260: la table ne peut avoir qu'une cl  primaire

SQL> alter table region constraint PK_region primary key (regCode);
alter table region constraint PK_region primary key (regCode)
*
ERREUR   la ligne 1 :
ORA-01735: option ALTER TABLE non valide

SQL> alter table region add constraint PK_region primary key (regCode);
Table modifi e.

SQL> alter table client add constraint FK_client foreign key (cliReg) references region(regCode);
alter table client add constraint FK_client foreign key (cliReg) references region(regCode)
*
ERREUR   la ligne 1 :
ORA-00904: "CLIREG" : identificateur non valide

SQL> alter table client add constraint FK_client foreign key (cliRegion) references region(regCode);
Table modifi e.

SQL>

```

Nous disposons d sormais de deux tables li es CLIENT et REGION.

Insertion des donn es et v rification de l'int grit  r f rentielle

A faire !

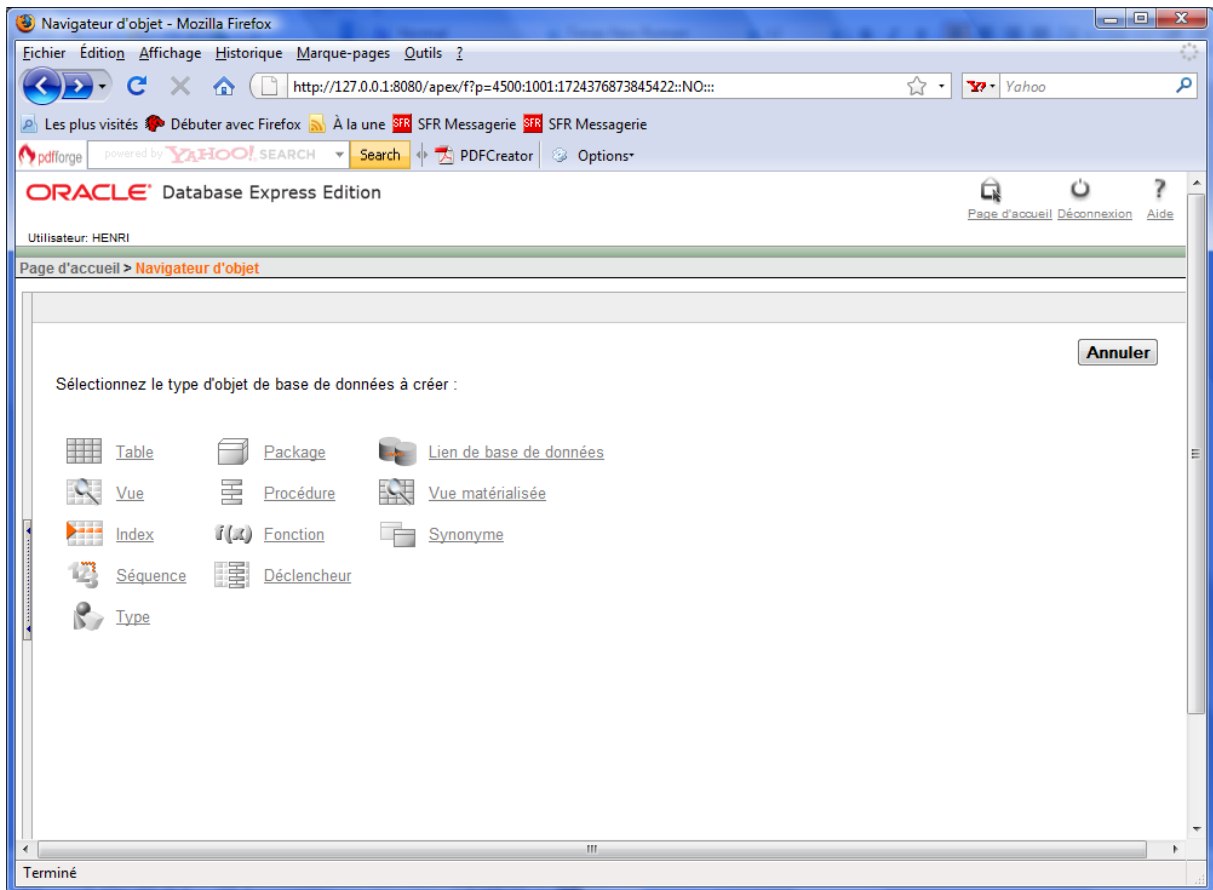
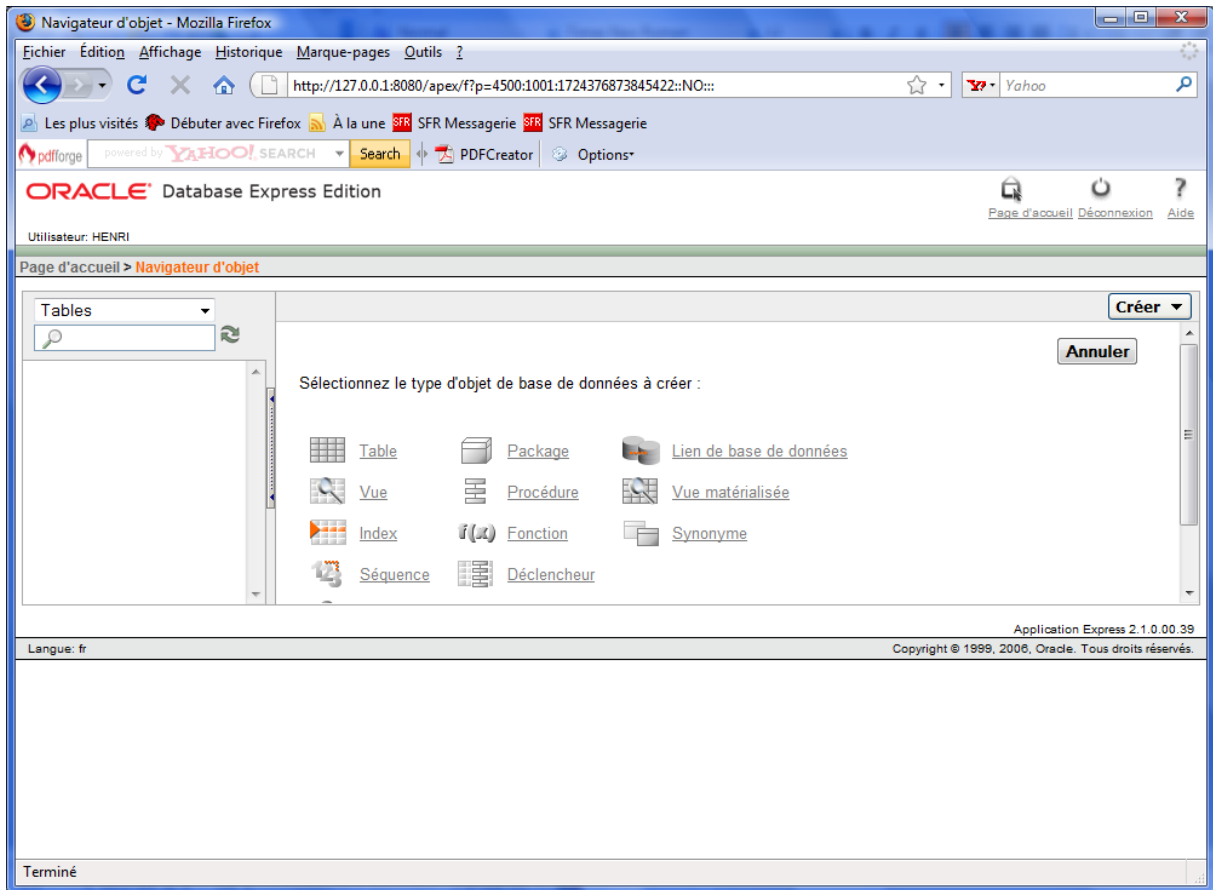
Cr ation d'une table **OPERATION**

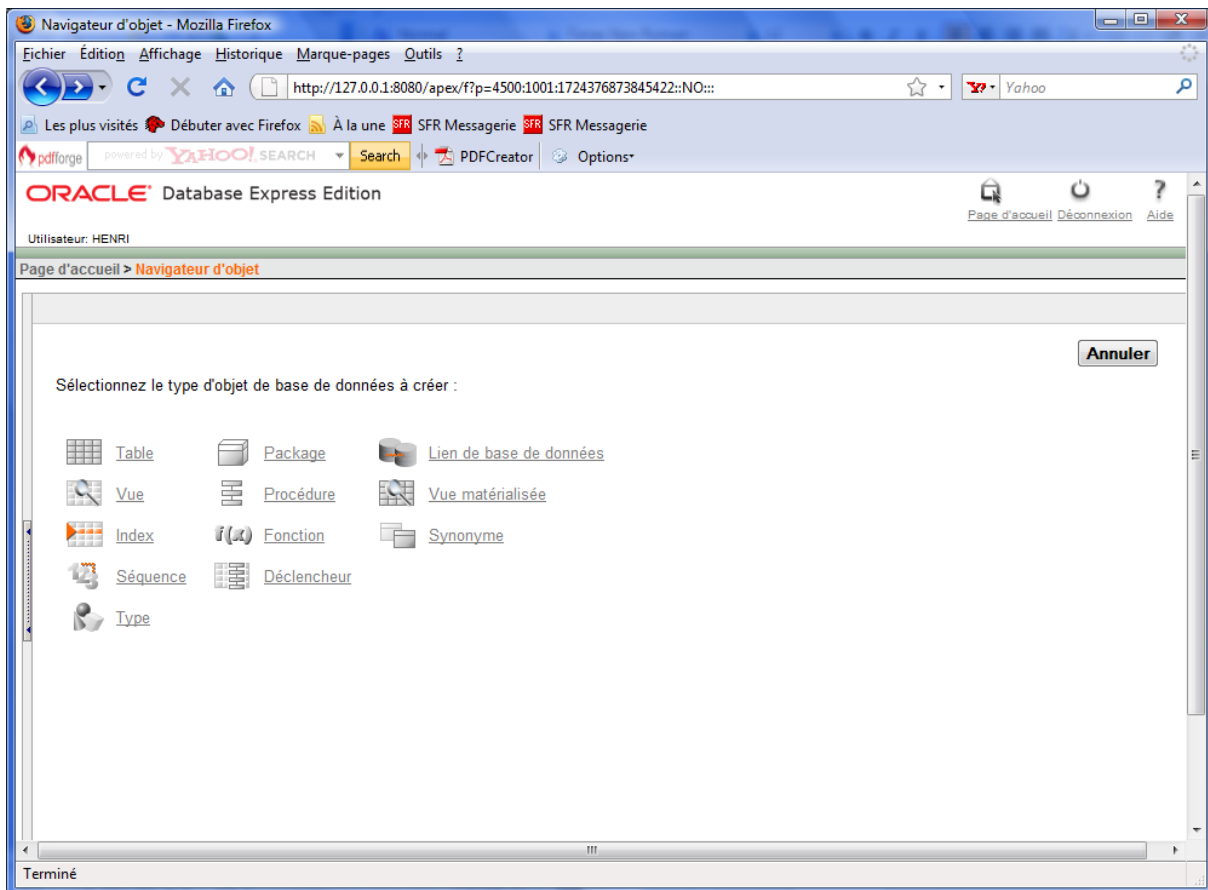
```

OPERAT (
  opeNum number(4,0) not null primary key,
  opeClient char(3) not null,
  opeDate Date,
  opeMontant number(8,2),
  opeType char(1) CONSTRAINT ck_ope CHECK (opeType IN ('D', 'C'))
);

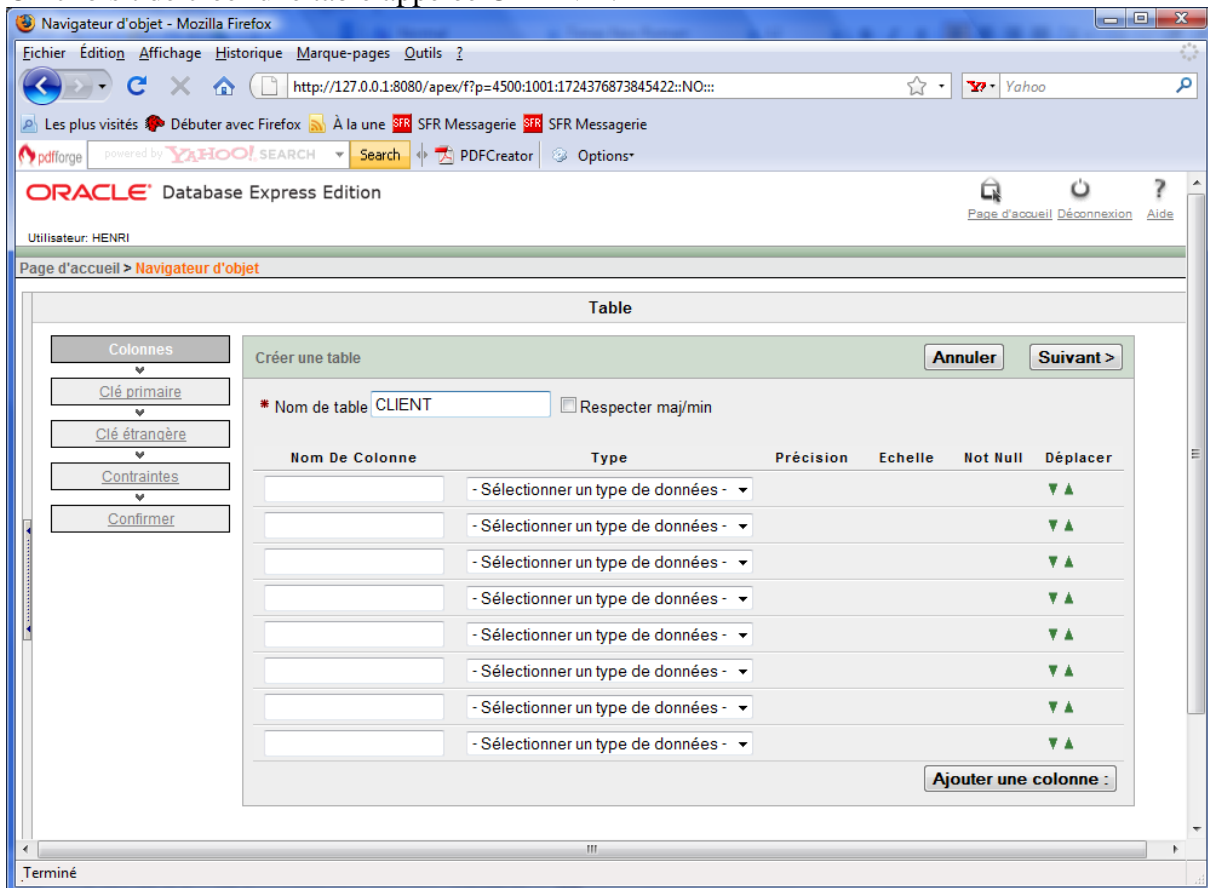
```

Cette table permettra la gestion des op rations, en particulier la MAJ du chiffre d'affaires du client avec contraintes du type D( bit) ou C(r dit), liste des op rations par type, etc.

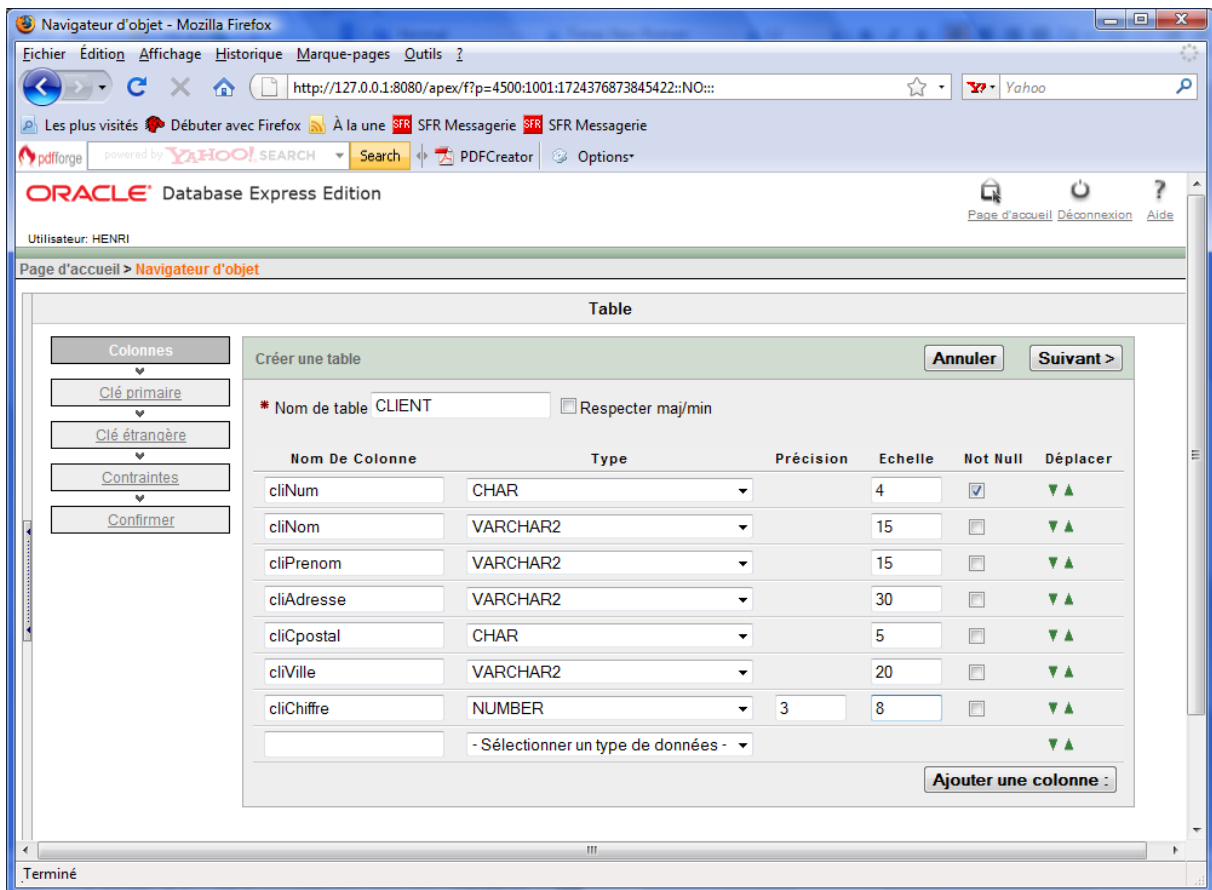




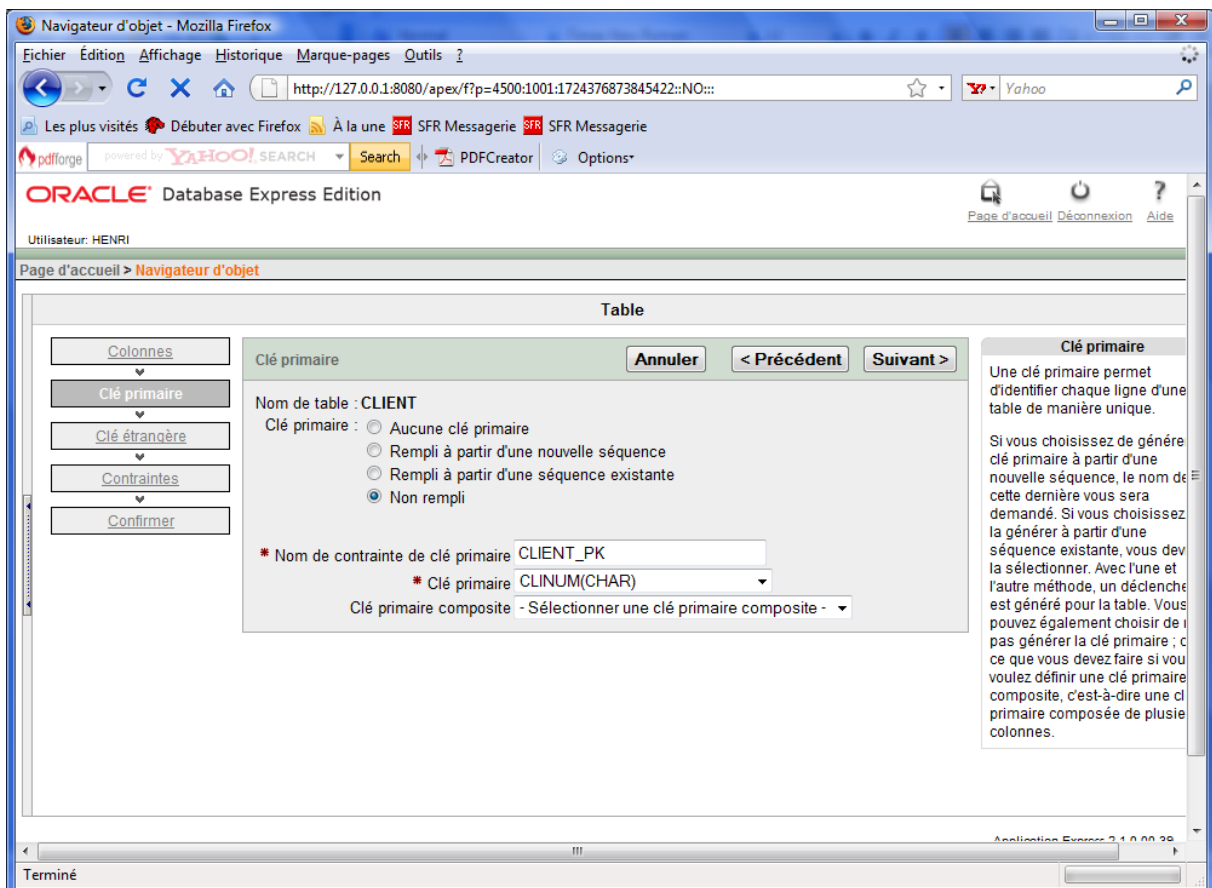
On choisit de créer une table appelée CLIENT :



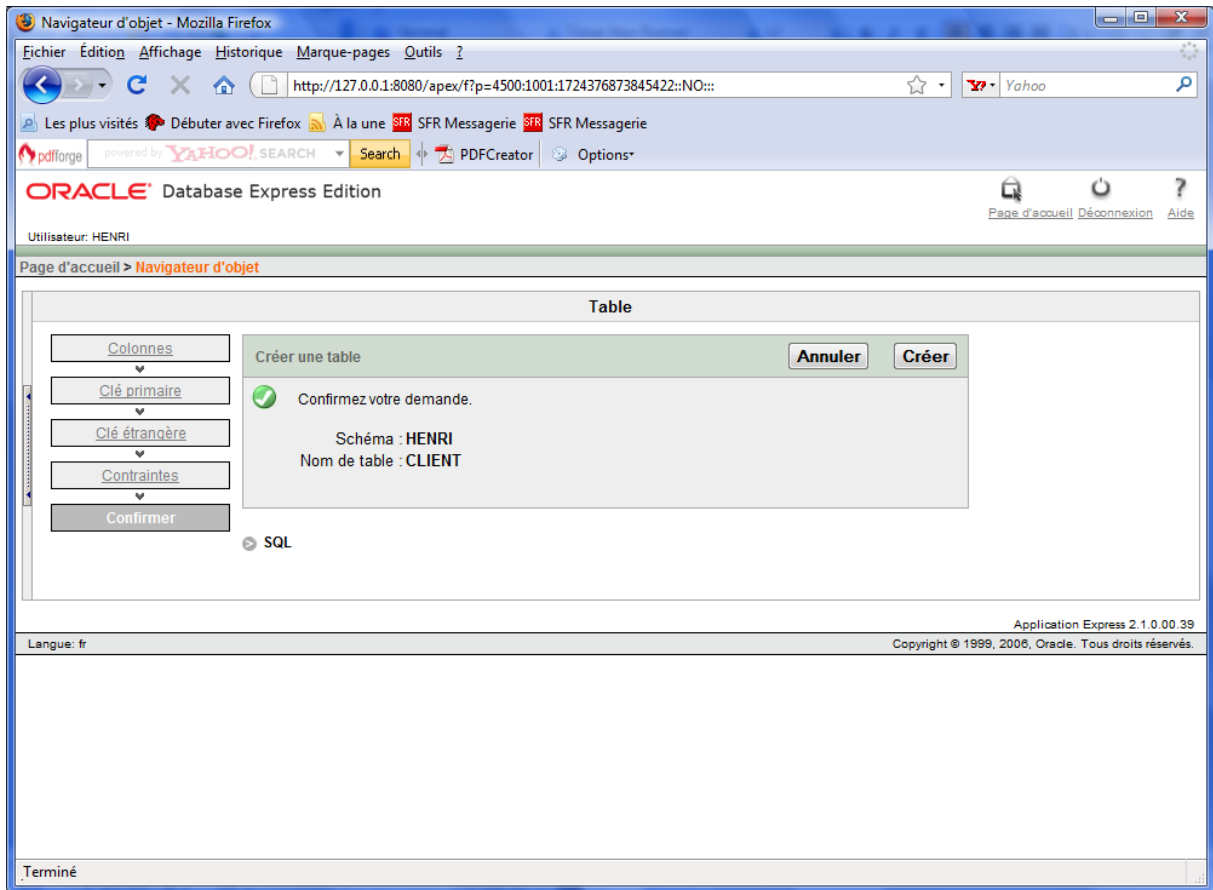
Compléter la structure de la table en renseignant les noms des champs et leurs propriétés.



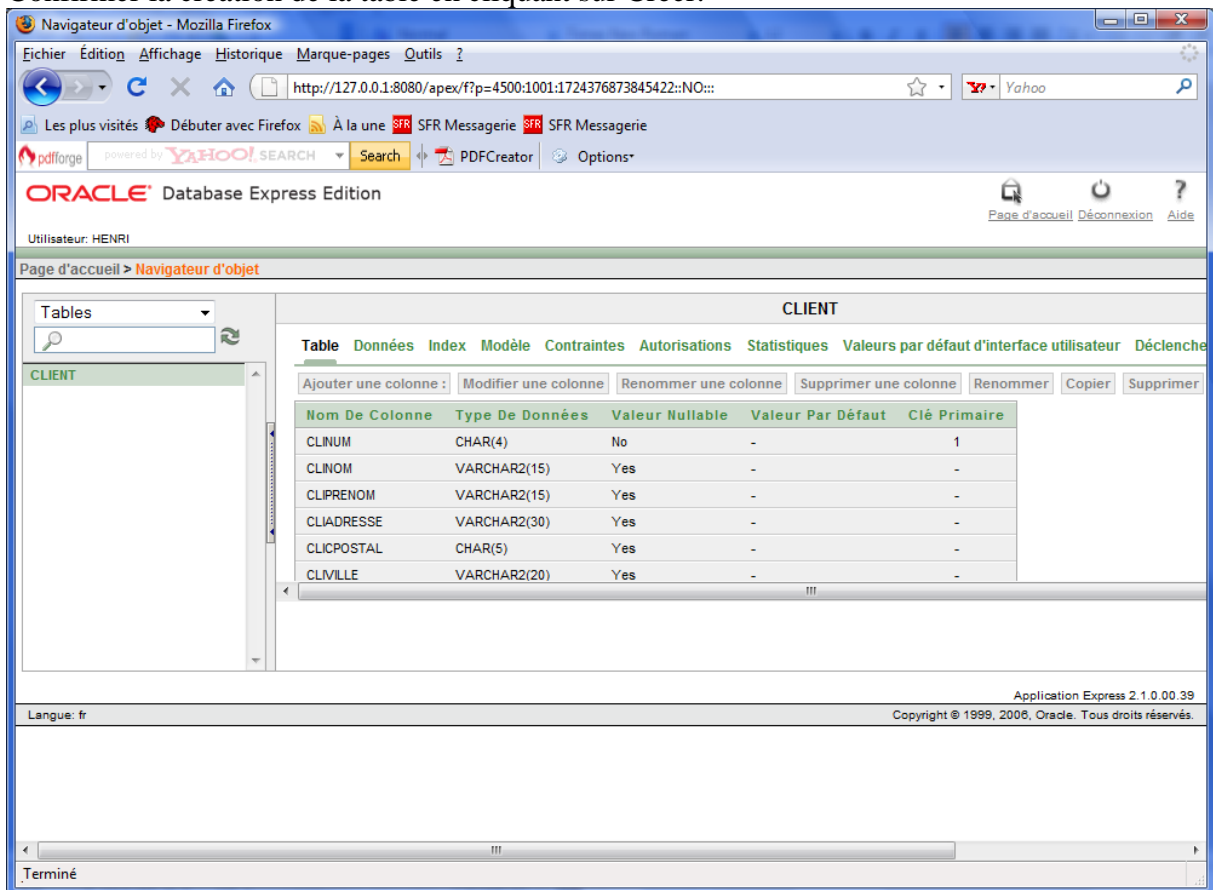
Définition de la clé primaire (PK : Primary Key) :



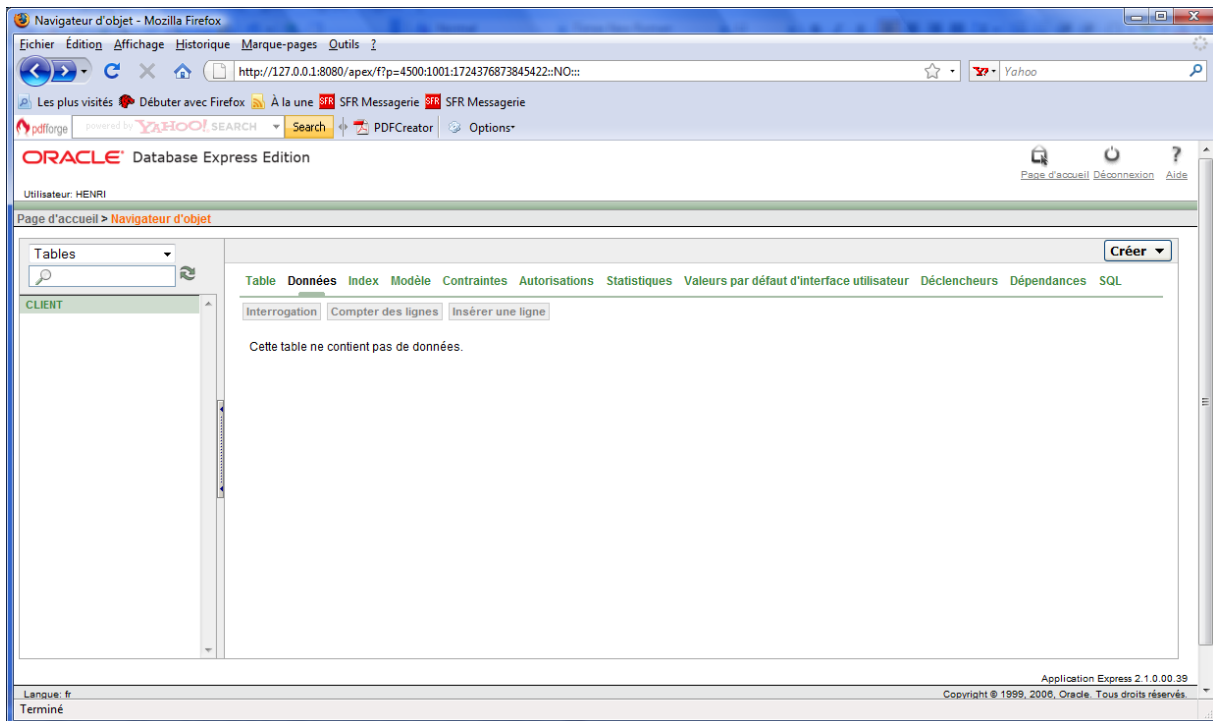
Confirmer cette spécification.



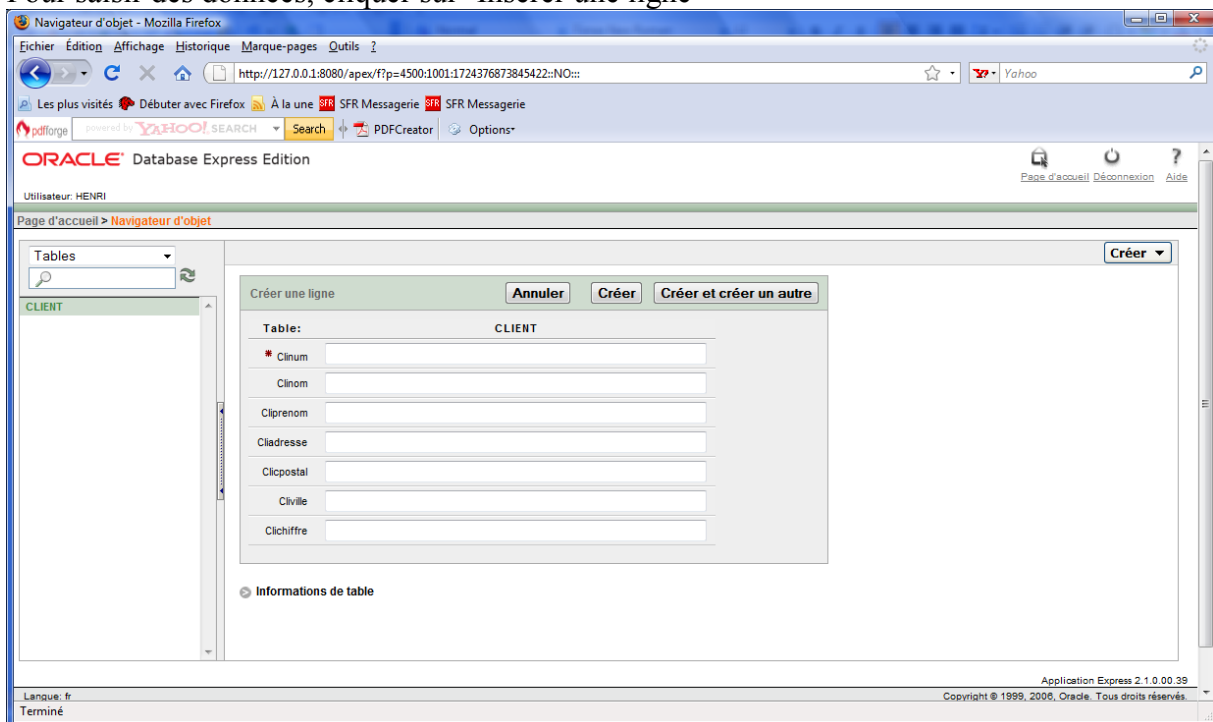
Confirmer la création de la table en cliquant sur Créer.

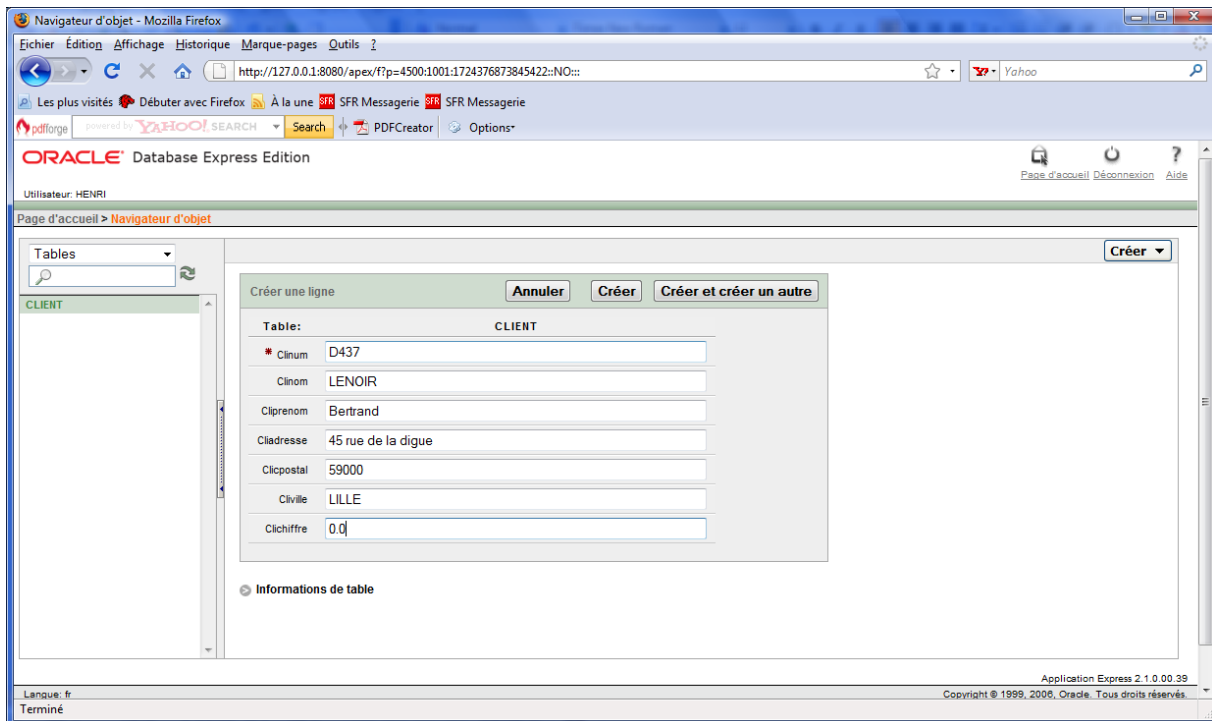


Cliquer sur Données

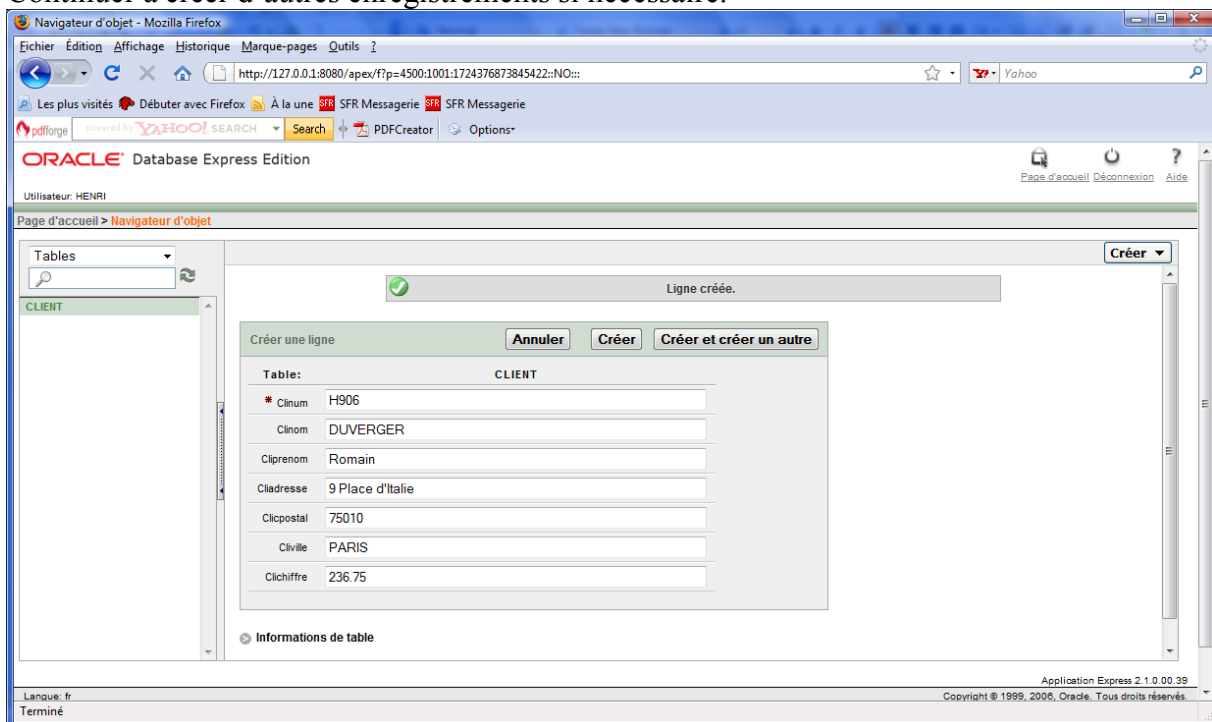


Pour saisir des données, cliquer sur 'Insérer une ligne'

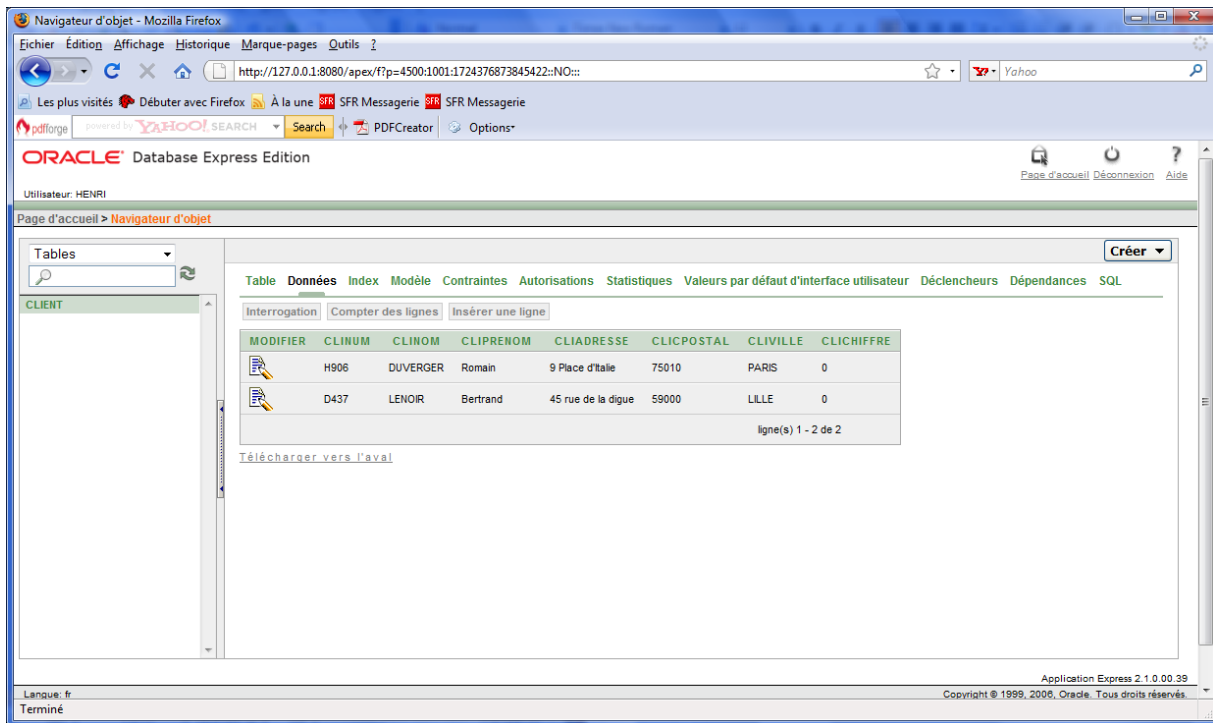




Continuer à créer d'autres enregistrements si nécessaire.



Erreur de données concernant la « précision » à gérer.

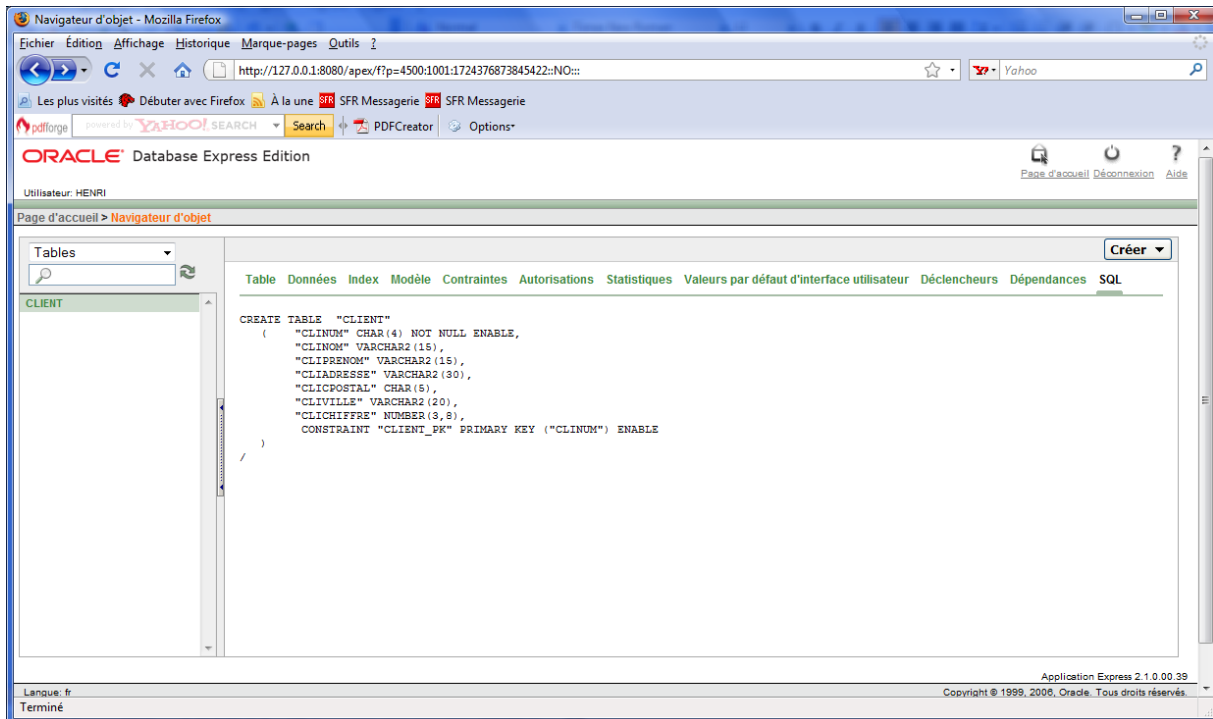


The screenshot shows the Oracle Database Express Edition interface in Mozilla Firefox. The browser address bar shows the URL: `http://127.0.0.1:8080/apex/f?p=4500:1001:1724376873845422::NO::`. The page title is "ORACLE Database Express Edition". The user is logged in as "HENRI". The main content area displays the "CLIENT" table data. The table has columns: MODIFIER, CLINUM, CLINOM, CLIPRENOM, CLIADRESSE, CLICPOSTAL, CLIVILLE, and CLICHIFFRE. The data rows are:

MODIFIER	CLINUM	CLINOM	CLIPRENOM	CLIADRESSE	CLICPOSTAL	CLIVILLE	CLICHIFFRE
	H906	DUVERGER	Romain	9 Place d'Italie	75010	PARIS	0
	D437	LENOIR	Bertrand	45 rue de la digue	59000	LILLE	0

Below the table, there is a link "Télécharger vers l'aval". The status bar at the bottom indicates "Langue: fr" and "Terminé".

Pour visualiser le code SQL généré, cliquer sur SQL

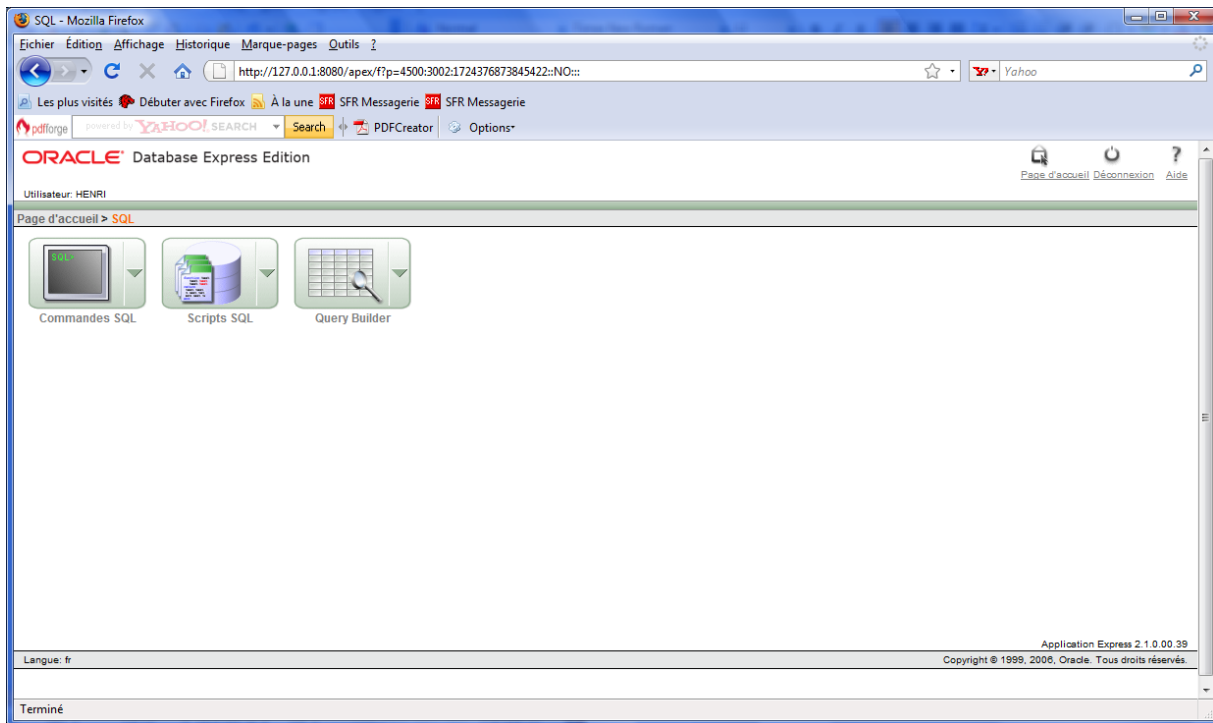


The screenshot shows the Oracle Database Express Edition interface in Mozilla Firefox. The browser address bar shows the URL: `http://127.0.0.1:8080/apex/f?p=4500:1001:1724376873845422::NO::`. The page title is "ORACLE Database Express Edition". The user is logged in as "HENRI". The main content area displays the SQL code for the "CLIENT" table. The code is:

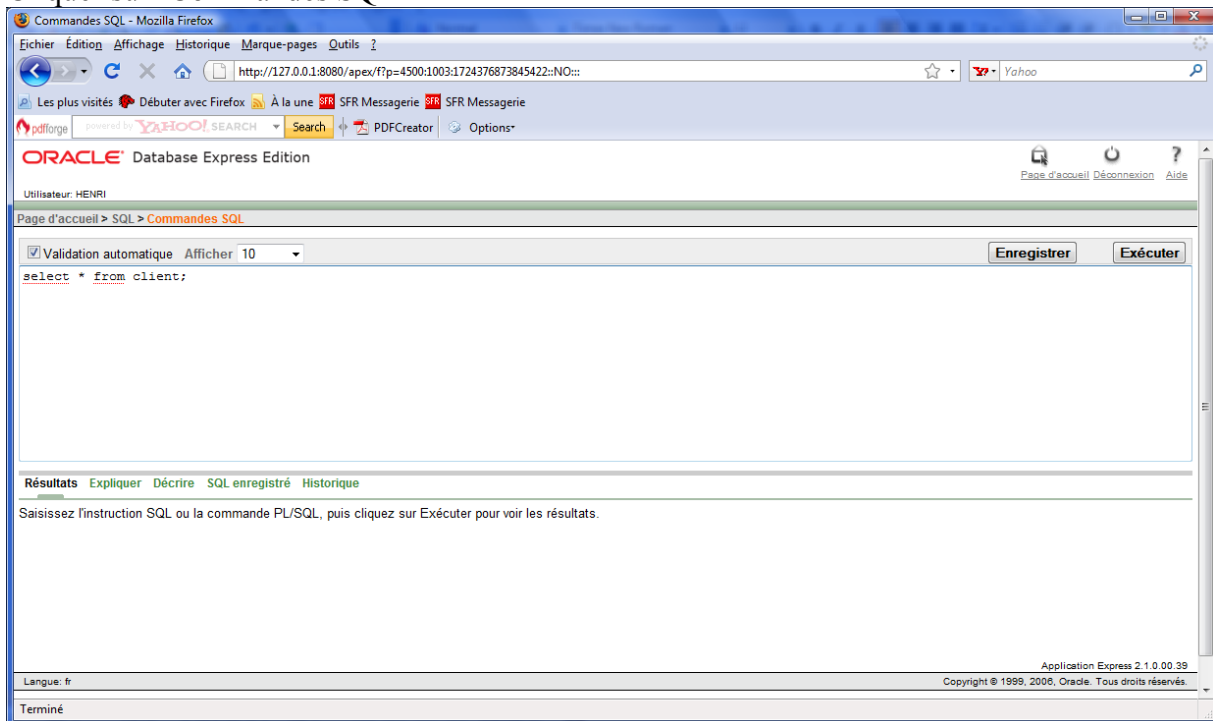
```
CREATE TABLE "CLIENT"  
(  
  "CLINUM" CHAR(4) NOT NULL ENABLE,  
  "CLINOM" VARCHAR2(15),  
  "CLIPRENOM" VARCHAR2(15),  
  "CLIADRESSE" VARCHAR2(30),  
  "CLICPOSTAL" CHAR(5),  
  "CLIVILLE" VARCHAR2(20),  
  "CLICHIFFRE" NUMBER(3,8),  
  CONSTRAINT "CLIENT_PK" PRIMARY KEY ("CLINUM") ENABLE  
)
```

The status bar at the bottom indicates "Langue: fr" and "Terminé".

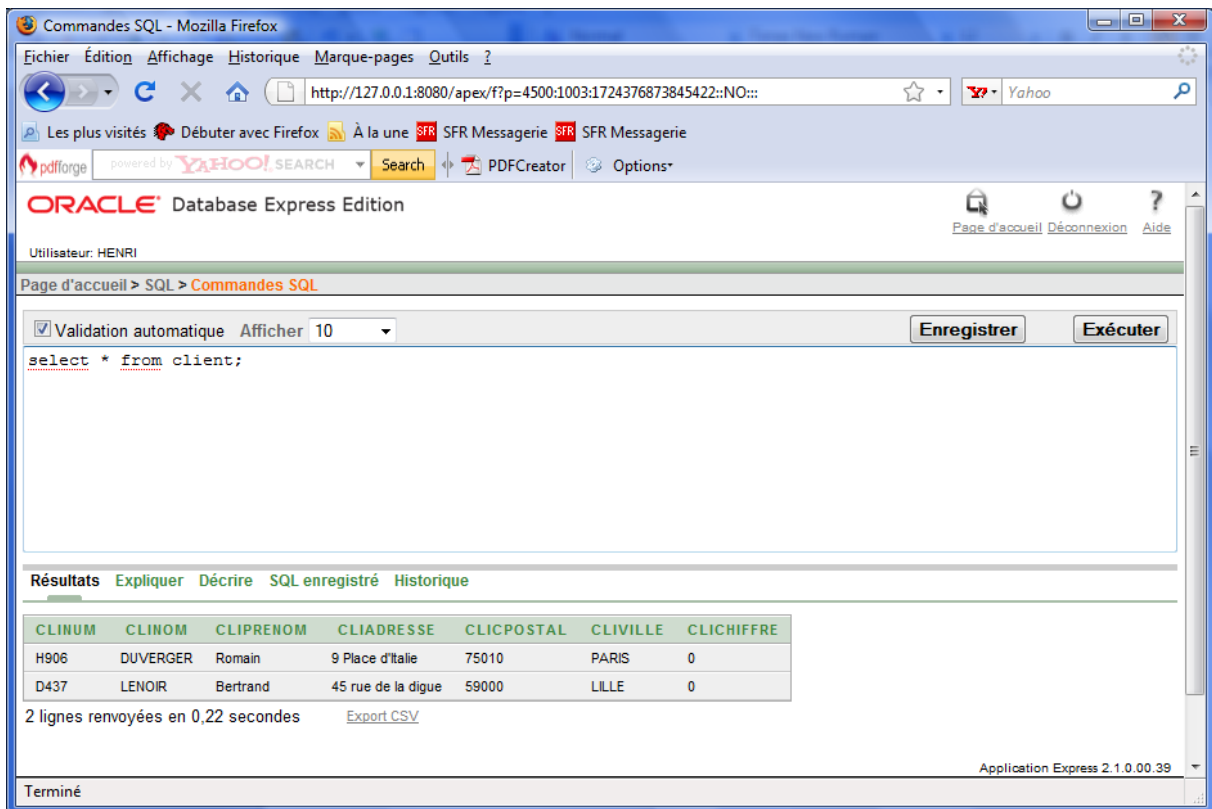
Retour à la page d'accueil



Cliquer sur 'Commandes SQL'



Tapez votre requête, sélectionnez-la avec la souris et cliquez sur 'Exécuter'



Commandes SQL - Mozilla Firefox

http://127.0.0.1:8080/apex/f?p=4500:1003:1724376873845422::NO::

ORACLE Database Express Edition

Utilisateur: HENRI

Page d'accueil > SQL > Commandes SQL

Validation automatique Afficher 10 Enregistrer Exécuter

```
select * from client;
```

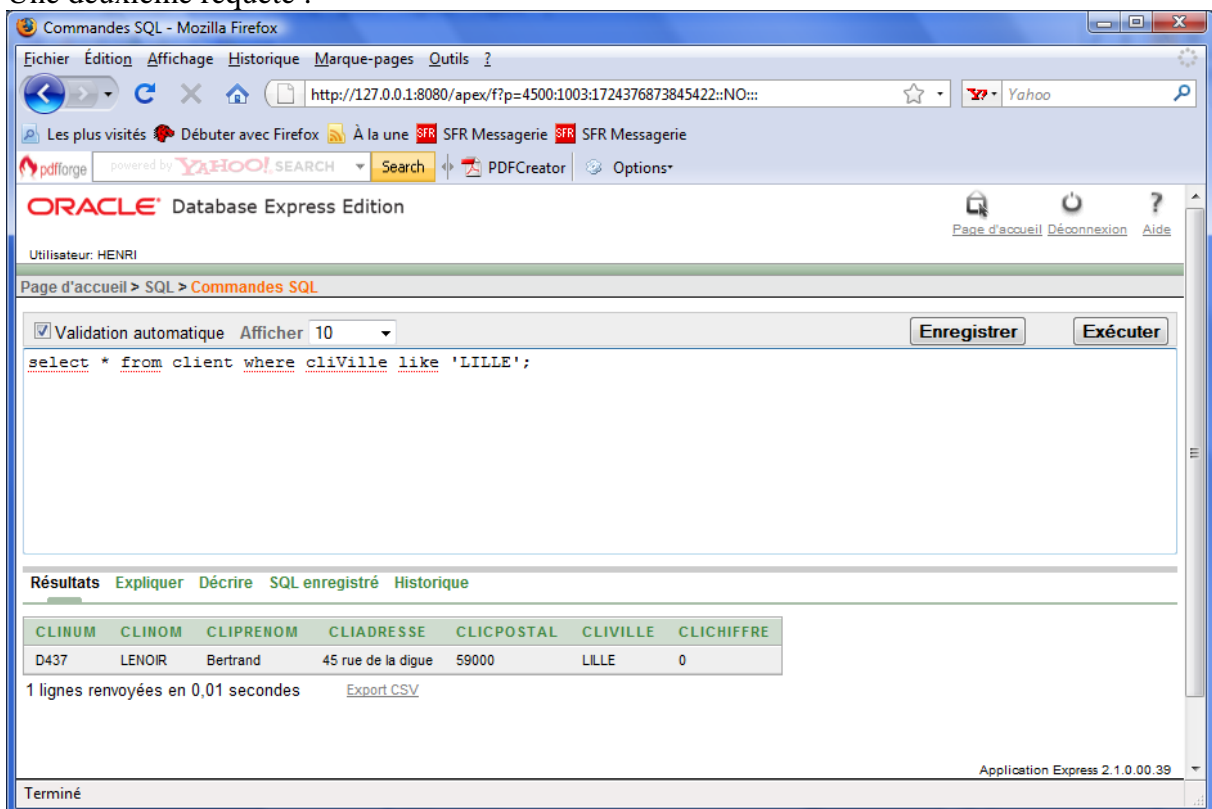
Résultats Expliquer Décrire SQL enregistré Historique

CLINUM	CLINOM	CLIPRENOM	CLIADRESSE	CLICPOSTAL	CLIVILLE	CLICHIFFRE
H906	DUVERGER	Romain	9 Place d'Italie	75010	PARIS	0
D437	LENOIR	Bertrand	45 rue de la digue	59000	LILLE	0

2 lignes renvoyées en 0,22 secondes [Export CSV](#)

Terminé Application Express 2.1.0.00.39

Une deuxième requête :



Commandes SQL - Mozilla Firefox

http://127.0.0.1:8080/apex/f?p=4500:1003:1724376873845422::NO::

ORACLE Database Express Edition

Utilisateur: HENRI

Page d'accueil > SQL > Commandes SQL

Validation automatique Afficher 10 Enregistrer Exécuter

```
select * from client where cliVille like 'LILLE';
```

Résultats Expliquer Décrire SQL enregistré Historique

CLINUM	CLINOM	CLIPRENOM	CLIADRESSE	CLICPOSTAL	CLIVILLE	CLICHIFFRE
D437	LENOIR	Bertrand	45 rue de la digue	59000	LILLE	0

1 lignes renvoyées en 0,01 secondes [Export CSV](#)

Terminé Application Express 2.1.0.00.39

Voilà ! A vous de décoller en trombe.

The screenshot shows the Oracle Database Express Edition interface in a Mozilla Firefox browser window. The browser's address bar displays the URL: `http://127.0.0.1:8080/apex/f?p=4500:1000:1724376873845422:::...`. The page title is "ORACLE Database Express Edition". The user is logged in as "HENRI". The interface features a "Page d'accueil" (Home) section with several navigation icons: Administration, Navigateur d'objet, SQL, Utilitaires, and Application Builder. On the right side, there are two panels: "Liens" (Links) and "Surveillance de l'utilisation" (Usage Monitoring). The "Liens" panel contains a list of links: Accord de licence, Initiation, En savoir plus, Documentation, Inscription au forum, Forum de discussion, and Page de produit. The "Surveillance de l'utilisation" panel shows two progress bars: "Stockage : 870MB" and "Mémoire : 617MB", both with 0% usage. Below these, it shows "Sessions : 6 Total, 1 Actif". The status bar at the bottom of the browser window indicates "Terminé".

Déconnectez-vous du serveur.

The screenshot shows the Oracle Database Express Edition interface in a Mozilla Firefox browser window. The browser's address bar displays the URL: `http://127.0.0.1:8080/apex/f?p=4550:9:4177489768213902::NO::...`. The page title is "ORACLE Database Express Edition". The main content area displays the message: "Vous êtes maintenant déconnecté." (You are now disconnected.). Below this message, there is a link labeled "Connexion". At the bottom of the page, there is a footer with the text: "Application Express", "Langue: fr", and "Copyright © 1999, 2006, Oracle. Tous droits réservés." The status bar at the bottom of the browser window indicates "Terminé".

Aurevoir et à très bientôt.

Déverrouillage et activation du compte de l'utilisateur HR :

Gérer un utilisateur de base de données

Annuler Supprimer

Modifier un utilisateur

Nom utilisateur **HR**

Mot de passe ●●

Confirmer le mot de passe ●●

Mot de passe expiré

Statut du compte Déverrouillé

Tablespace par défaut : USERS

Tablespace temporaire : TEMP

Privilèges utilisateur

Rôles :

CONNECT RESOURCE DBA

Privilèges système accordés directement :

CREATE DATABASE LINK CREATE MATERIALIZED VIEW CREATE PROCEDURE

CREATE PUBLIC SYNONYM CREATE ROLE CREATE SEQUENCE

CREATE SYNONYM CREATE TABLE CREATE TRIGGER

CREATE TYPE CREATE VIEW

[Cocher tout](#) [Désélectionner tout](#)

Tous les privilèges système accordés à HR

Application Express 2.1.0.0.39

Langue: fr Copyright © 1999, 2006, Oracle. Tous droits réservés.

Terminé

On attribue le mot de passe hr à l'utilisateur et pour valider ces modifications, on clique sur 'Modifier un utilisateur'. Résultat : l'icône HR ne comporte plus de cadenas. Cet utilisateur peut désormais se connecter.

Utilisateurs de base de données - Mozilla Firefox

Page d'accueil > Administration > Gérer les utilisateurs de base de données

Mot de passe de l'utilisateur modifié. Utilisateur modifié.

Rechercher le nom utilisateur Vue Icônes Afficher Utilisateurs de base de données Afficher 15 OK Créer >

HENRI HR

Terminé

Visualisation de tous les privilèges de l'utilisateur HR.

Gérer un utilisateur de base de données - Mozilla Firefox

http://127.0.0.1:8080/apex/f?p=4500:2613:3988700734959618::NO:RP,2613:P2613_USER_ID:10921

Les plus visités Débuter avec Firefox À la une SFR SFR Messagerie SFR SFR Messagerie

powered by YAHOO! SEARCH Search PDFCreator Options

Tous les privilèges système accordés à HR
Cet état affiche les privilèges système accordés à cet utilisateur, y compris les privilèges octroyés directement et les privilèges accordés via des rôles.

Privilège	Option D'administration	A Partir Du Rôle
ALTER SESSION	-	-
CREATE CLUSTER	-	RESOURCE
CREATE DATABASE LINK	-	-
CREATE INDEXTYPE	-	RESOURCE
CREATE OPERATOR	-	RESOURCE
CREATE PROCEDURE	-	RESOURCE
CREATE SEQUENCE	-	-
CREATE SEQUENCE	-	RESOURCE
CREATE SESSION	-	-
CREATE SESSION	-	CONNECT
CREATE SYNONYM	-	-
CREATE TABLE	-	RESOURCE
CREATE TRIGGER	-	RESOURCE
CREATE TYPE	-	RESOURCE
CREATE VIEW	-	-
UNLIMITED TABLESPACE	-	-

1 - 16

Application Express 2.1.0.00.39
Langue: fr Copyright © 1999, 2006, Oracle. Tous droits réservés.

Terminé

Exemple de traitements sur une petite table appelée 'TEST'

Structure de la table :

Table	Column	Type De Données	Longueur	Précision	Echelle	Clé Primaire	Valeur Nullable	Valeur Par Défaut	Commentaire
TEST	NUM	Char	3	-	-	1	-	-	-
	NOM	Varchar2	15	-	-	-	✓	-	-
	SALAIRE	Number	-	8	2	-	✓	-	-

1 - 3

The screenshot shows the Oracle Database Express Edition interface in a Mozilla Firefox browser. The user is HENRI. The SQL Command window contains the following commands:

```

from client;
desc client;
alter table client modify cliChiffre (Number(8,3));
alter table test modify (salaire number(8,2));
desc test;
insert into test values ('Y10', 'LANCIA', 849.28);
select * from test;
select Num MATRICULE, Nom NOM, salaire SALAIRE
from test;

```

The results of the last query are displayed in a table:

MATRICULE	NOM	SALAIRE
F45	DECHANTRE	5648
F03	LANNOY	5648,64
Z51	MOHAMAD	1674,5
Y10	LANCIA	849,28

4 lignes renvoyées en 0,01 secondes [Export CSV](#)

Application Express 2.1.0.00.39
Copyright © 1999, 2008, Oracle. Tous droits réservés.

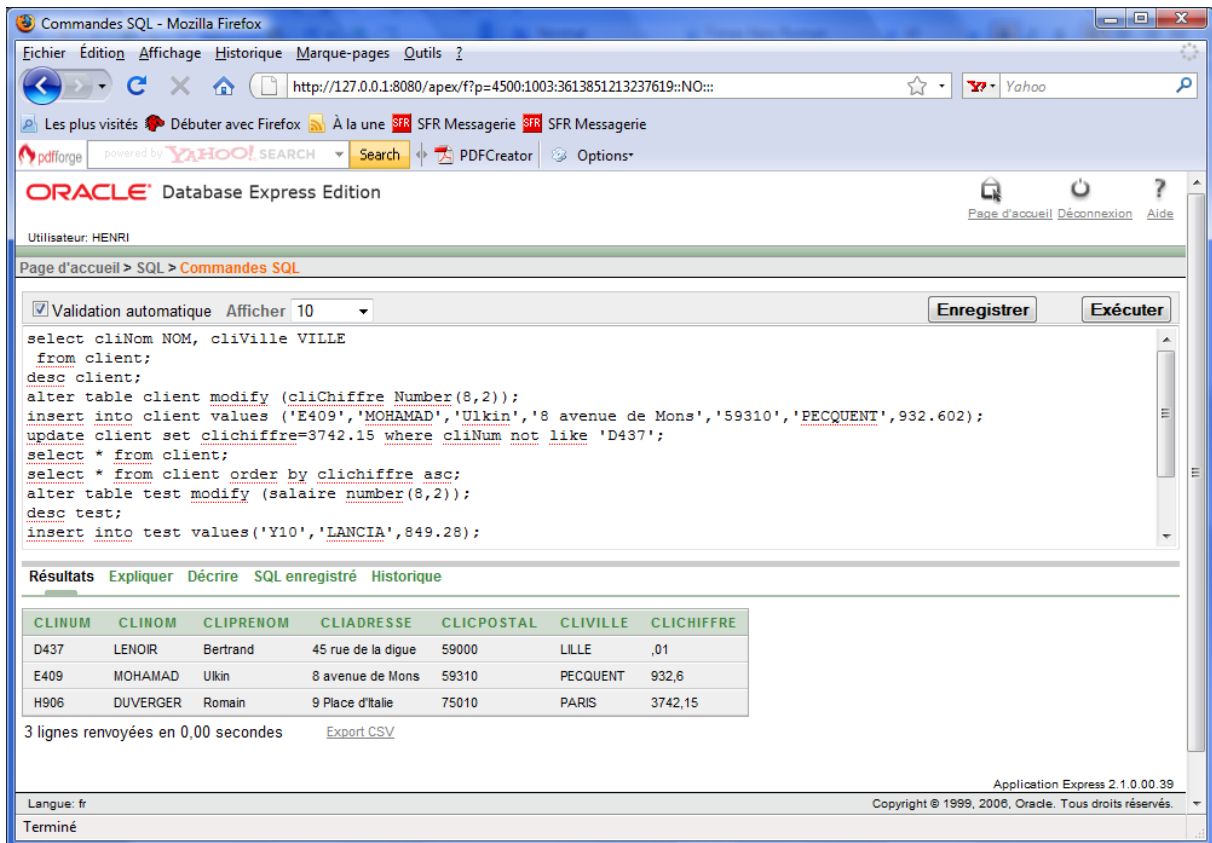
Terminé

Requêtes

- 1-Affichage de la structure (description) de la table (cf. capture ci-dessus)
describe test ;
- 2-Liste de toutes les colonnes avec Titre
**SELECT Num MATRICULE, Nom NOM, salaire SALAIRE
FROM test;**
- 3-Modification de la structure (colonne Salaire) de la table
ALTER TABLE test MODIFY (salaire number(8,2));
- 4-Affichage des noms de toutes les tables
select table_name from tabs;

TABLE_NAME
CLIENT
TEST
HTMLDB_PLAN_TABLE

En résumé, voici la capture finale :



The screenshot shows the Oracle Database Express Edition interface in a Mozilla Firefox browser. The browser address bar shows the URL: `http://127.0.0.1:8080/apex/f?p=4500:1003:3613851213237619::NO::`. The page title is "ORACLE Database Express Edition". The user is identified as "HENRI".

The "Commandes SQL" window is active, displaying the following SQL commands:

```
select cliNom NOM, cliVille VILLE
  from client;
desc client;
alter table client modify (cliChiffre Number(8,2));
insert into client values ('E409','MOHAMAD','Ulkin','8 avenue de Mons','59310','PECQUENT',932.602);
update client set cliChiffre=3742.15 where cliNum not like 'D437';
select * from client;
select * from client order by cliChiffre asc;
alter table test modify (salaire number(8,2));
desc test;
insert into test values ('Y10','LANCIA',849.28);
```

The results of the SQL commands are displayed in a table with the following columns: CLINUM, CLINOM, CLIPRENOM, CLIADRESSE, CLICPOSTAL, CLIVILLE, and CLICHIFFRE. The table contains three rows of data:

CLINUM	CLINOM	CLIPRENOM	CLIADRESSE	CLICPOSTAL	CLIVILLE	CLICHIFFRE
D437	LENOIR	Bertrand	45 rue de la digue	59000	LILLE	,01
E409	MOHAMAD	Ulkin	8 avenue de Mons	59310	PECQUENT	932,6
H906	DUVERGER	Romain	9 Place d'Italie	75010	PARIS	3742,15

Below the table, it indicates "3 lignes renvoyées en 0,00 secondes" and provides a link to "Export CSV".

The footer of the application shows "Application Express 2.1.0.00.39", "Langue: fr", and "Copyright © 1999, 2006, Oracle. Tous droits réservés." The status bar at the bottom indicates "Terminé".

Place aux Travaux Pratiques !

Base GESCOM

On considère la base GESCOM relative à une gestion de commandes, constituée des relations suivantes :

CLIENT(cliNum, clinom, cliVille, cliDept)

PRODUIT(prodNum, prodLibel, prodPuht, prodStock)

COMMANDE(comNum, comClient, comDate)

LIGNECOM(lignCmde, lignProd, lignQte)

**** On pourrait aussi ajouter une table des fournisseurs ...**

Tables de données

CLIENT

	cliNum	cliNom	cliVille	cliDept
<input type="checkbox"/>	x23	Dumont	Calais	62
<input type="checkbox"/>	k06	Lilian	Lille	59
<input type="checkbox"/>	k16	Cauet	Anzin	59
<input type="checkbox"/>	x10	Ribeau	Lens	62
<input type="checkbox"/>	k25	Fresnoy	Lille	59
<input type="checkbox"/>	p17	Amar	St Quentin	02
<input type="checkbox"/>	d65	Lemaitre	Arras	62

COMMANDE

	comNum	comClient	comDate
<input type="checkbox"/>	c01	x23	2005-10-20
<input type="checkbox"/>	c05	x23	2006-01-23
<input type="checkbox"/>	c03	x10	2005-07-18
<input type="checkbox"/>	c04	k25	2005-08-05
<input type="checkbox"/>	c02	k16	2004-12-15
<input type="checkbox"/>	c06	k16	2005-12-08

PRODUIT

	prodNum	prodLibel	prodPuht	prodStock
<input type="checkbox"/>	p125	crayons	4.50	50
<input type="checkbox"/>	p237	lampe bureau	15.25	49
<input type="checkbox"/>	p429	couvertures	8.42	20
<input type="checkbox"/>	p334	rame x 500 60g	3.27	15
<input type="checkbox"/>	p502	stylo plume	9.00	35
<input type="checkbox"/>	p144	lot chemises	5.75	40
<input type="checkbox"/>	p018	equerre	4.10	60
<input type="checkbox"/>	p223	boîte multiusages	3.74	110
<input type="checkbox"/>	p420	ramette x 500 80g	3.50	32

LIGNECOM

	lignCmde	lignProd	lignQte
<input type="checkbox"/>	c01	p237	2
<input type="checkbox"/>	c01	p018	5
<input type="checkbox"/>	c02	p502	1
<input type="checkbox"/>	c02	p237	8
<input type="checkbox"/>	c02	p334	2
<input type="checkbox"/>	c03	p223	10
<input type="checkbox"/>	c04	p125	4
<input type="checkbox"/>	c04	p223	5
<input type="checkbox"/>	c04	p502	3
<input type="checkbox"/>	c05	p144	8
<input type="checkbox"/>	c06	p018	2

Enoncés des requêtes

- 1) Clients du Nord et du Pas De Calais :
- 2) Liste des clients triée sur Nom, ville,department :
- 3) Nombre de clients par department :
- 4) Liste des produits pordre croissant du prix :
- 5) Quantité totale en stock (nombre de produits):
- 6) Produits de la commande « C02 » :
- 7) Numéro, nom,ville,date commande du client ayant passé la commande « c03 » :
- 8) Montant de chacune des commandes
- 9) Nombre de commandes en 2005 :
- 10) Commandes par an :
- 11) Nombre de clients ayant commandé le produit “p018” :
- 12) Clients qui n’ont pas (jamais) passé commande :
- 13) Nombre de commandes par produit

Solutions

...

Exercice(d’après un DS en BTS IG 1^{ère} année)

On propose la base de données ‘BANQUE’ composée des relations suivantes :

TYPECPTE(TypeCpteCode, TypeCpteLibelle)
COMPTE(CpteNum, CpteClient, CpteType, CpteDate, CpteDispo, CpteDecouvert)
CLIENT(CliNum, CliNom, CliPrenom, CliAdresse, CliVille)
OPERATION(OperNum, OperCpte, OperType, OperPersonne, OperMontant, OperDate)
TYPEOPER(TypeOperCode, TypeOperLibelle)

- A) Ecrire les requêtes SQL permettant d’obtenir les résultats suivants :
- 1) Liste des n° de compte, n° de client, nom du client, type de compte, Date d’ouverture, disponible.
 - 2) Liste des n° de compte, n° de client, ville, disponible, pour les comptes des clients de «Valenciennes ».
 - 3) N° de compte, nom du client, pour lesquels le découvert dépasse 750 euros et pour les comptes de type « C » (chèque).
 - 4) Liste de tous les n° de compte du type « E » (Epargne) des clients de Lille dont l’ouverture est postérieure à 2005.
- B) Quels résultats donnent les requêtes suivantes :
- 5)

```
SELECT CliNum, CliNom, CpteDispo
FROM Client CLI, Compte CPT
WHERE CLI.CliNum=CPT.CpteClient
AND CPT.CpteDispo < (-CpteDecouvert);
```
 - 6)

```
SELECT CpteClient, COUNT(CpteNum) FROM COMPTE
GROUP BY CpteClient;
```

Exercice (d'après un DS en BTS IG 1^{ère} année)

Soit la base suivante

ETUDIANT(NumEtud, NomEtud, Sexe, DateNaiss, Ville, Dept)

INSCRIPTION(NumEtud, NumCours, Frais, Note)

COURS(NumCours, Titre, HeuresLabo)

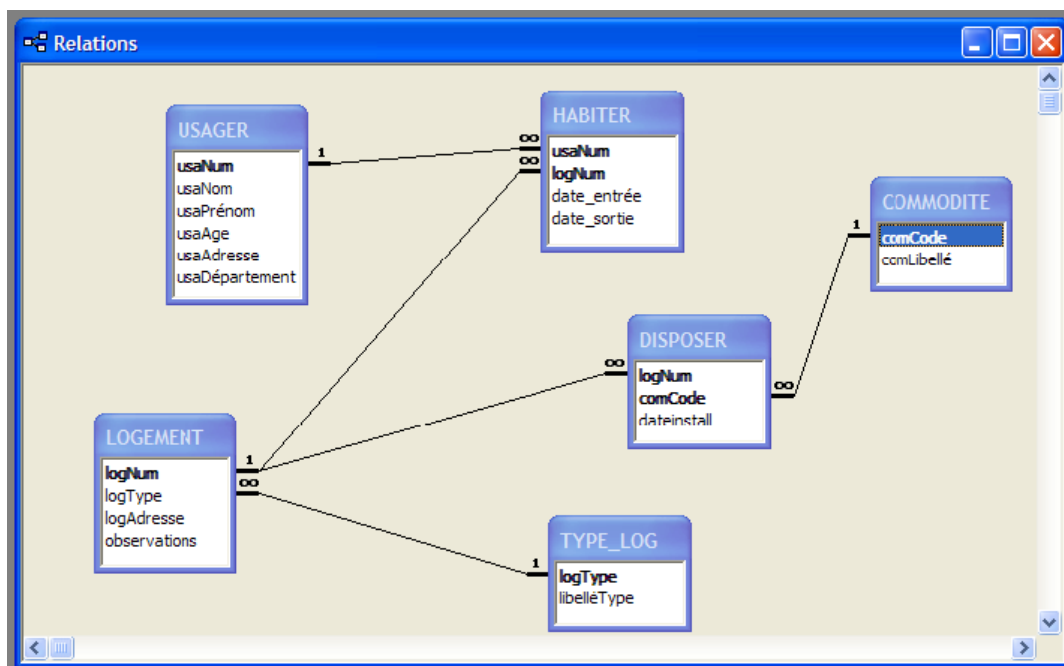
Ecrire les requêtes pour obtenir

- 1) La liste des noms des étudiants, sexe, Frais pour les étudiants du Nord(Dept).
- 2) Noms et ville des étudiants habitant la même ville que « McDonald ».
- 3) Liste des noms, Date de naissance des étudiants de la ville de « QUEBEC » de sexe masculin('M').
- 4) La somme de tous les frais d'inscription à tous les cours.

Base de données STATION

Situation

Les usagers d'une station de sports d'hiver habitent des logements caractérisés par certaines commodités : téléviseur, petit réfrigérateur, WC individuel, douche, etc. Les appartements sont de types F1, F1 bis, F2, F3, studio, etc. On vous propose le schéma de la base ci-dessous.

Tables et liens

USAGER : Table						
	usaNum	usaNom	usaPrénom	usaAge	usaAdresse	usaDépartement
▶ +	DM001	MARTIN	Daniel	53	45, rue du Général Leclerc	75
+	EF320	FADIGAN	Eric	22	6, place Belleville	59
+	FE418	ESTIEN	Frédéric	23	2, avenue Arthur	59
+	HT142	TURNOUT	Hélias	29	16, rue Monge	80
+	MA803	ANTOINE	Marc	28	11, rue Carpentier	75
+	RB600	BAUDUIN	Robert	32	25, rue de Dunkerque	62
*				0		0

Enr : 1 sur 6

Base STATION

USAGER(usaNum, usaNom, usaPrenom, usaAge, usaAdresse, usaDept)
HABITER(usaNum,logNum,date_entree, date_sortie)
LOGEMENT(logNum, logType, logAdresse, logLoyer, Observations)
TYPE_LOG(logType, libelleType)
COMMODITE(comCode, comLibelle)
DISPOSER(logNum,comCode,D_install)

Requêtes

- 1) Liste des n° d'utilisateur, nom, n° de logement occupé et adresse de ce logement triée sur le nom d'utilisateur.
- 2) Liste des logements de type « studio » disposant d'un réfrigérateur.
- 3) Nombre de logement actuellement inoccupés.
- 4) Numéro et adresse des logements comportant au moins deux commodités.
- 5) Nombre de logements « F2 » comportant une douche individuelle.
- 6) Quels usagers (noms, âge) ont un âge strictement supérieur à l'âge moyen des usagers.
- 7) Quels usagers (nom, prénom, département) sont originaires du même département que l'utilisateur de n° « DM001 » ?
- 8) Quelle est la masse monétaire de tous les loyers mensuels ?
- 9) Combien rapportent les « studios » ?
- 10) Nombre de logements par type(liste type logement, nombre).
- 11) Liste des n° appartement et nombre de commodités par appartement.
- 12) Liste des n° d'utilisateur, nom, prénom, type de logement occupé, adresse, loyer, date d'entrée pour tous les usagers, triée sur le nom d'utilisateur.

	logNum	logType	logAdresse	observations
▶ +	A009	4	12 bis, rue de la digue	Logement récent (1996)
	A184	3	12, rue de la digue	Logement récent (1996)
	A715	3	9, impasse Carvin	RAS
	B503	2	5, place de la Mairie	RAS
	C143	1	28, rue du molinel	Café à 50 m
	D108	1	3, square St François	Chapelle à 200 m
*				

Enr : 1 sur 6

	usaNum	logNum	date_entree	date_sortie
▶	DM001	A184		
	EF320	A715		
	FE418	B503		
	HT142	D108		
	MA803	C143		
	RB600	A009		
*				

Enr : 1 sur 6

	logType	libelleType
▶ +	1	F1
	2	F1 bis
	3	Studio
	4	Type 3
	5	Type 4
*		

Enr : 1 sur 5

The image shows two screenshots of Oracle database tables. The left window, titled 'DISPOSER : Table', displays a table with columns 'logNum', 'comCode', and 'dateinstall'. The right window, titled 'COMMODITE : Table', displays a table with columns 'comCode' and 'comLibellé'.

logNum	comCode	dateinstall
A009	C02	
A184	C03	23/10/1999
A184	C04	
A715	C01	11/08/1999
A715	C02	11/08/1999
A715	C03	11/08/1999
A715	C05	11/08/1999
B503	C01	01/12/1998
B503	C03	01/12/1998
B503	C04	15/06/1999
C143	C03	
C143	C04	
D108	C01	
*		

comCode	comLibellé
+ C01	Téléviseur
+ C02	Frigo
+ C03	WC
+ C04	Douche
+ C05	Baignoire
*	

Base EPICERIE

4 nov 2011

On vous propose la BDD dont les tables sont données ci-dessous.

FOURNISSEUR	NOM	REMISE	VILLE
1	BORNIBUS	5	PARIS
2	MERCIER	7	PARIS
3	COLBERT	3	REIMS
4	BOSSUET	6	DIJON
5	TANGUY	10	RIEC
6	DUPOND	0	PARIS
7	CABART	-	CAEN

FOU (Fournisseurs)

PRODUIT	PROD NOM	COULEUR	ORIGINE	PRIX
1	CASSIS	ROUGE	DIJON	35,5
2	CHAMPAGNE	BLANC	REIMS	87,32
3	HUITRE	VERT	RIEC	55,34
4	MOUTARDE	JAUNE	DIJON	5,65
5	SALADE	VERT	NICE	2,1
6	CORNICHON	VERT	DIJON	6,14
7	MUSCADET	BLANC	NANTES	25,99

PROD (Produits)

FOUR	PROD	QUANTITE
1	1	1
1	4	1
1	5	8
1	6	2
2	2	1
2	4	1
3	2	5
3	4	1
4	4	2
4	5	7
4	6	3
5	3	10

FOUPROD (Stocks)

Travail à faire

I – Donner le MCD correspondant.

Proposer des types de données cohérents pour chacune des tables.

Créer les tables de cette base sous Oracle-XE.

** NOTE : vous pouvez renommer les champs pour faciliter votre travail.

II- Ecrire les requêtes SQL suivantes, les tester et faire des captures des résultats :

Série 1

- 1) Liste des infos produits pour les produits dont le nom comporte la chaîne « CA ».
- 2) Liste des produits de couleur verte ou jaune.
- 3) Liste des infos pour les produits dont le prix est inférieur à celui, inconnu, du cornichon.
- 4) Noms et couleurs des produits de DIJON.
- 5) Liste des produits dont le prix est supérieur à celui, inconnu, du muscadet.
- 6) Liste des produits et quantité (en stock).
- 7) Liste des produits, fournisseur (code et nom) pour les produits qui ne sont pas originaires de DIJON.
- 8) Liste des produits (N°, nom, fournisseur) de Paris en quantité supérieure à 2, strictement.
- 9) Infos des produits dont le prix est supérieur au prix moyen (inconnu).
- 10) Nom du produit le plus cher et nom de son fournisseur.

Série 2

- 11) Nombre de produits par couleur
- 12) Nombre de produits par fournisseur
- 13) Liste N° Fournisseurs, Nom fournisseur, Nom produits par ordre croissant du code Fournisseur
- 14) Liste des fournisseurs sans produits en stock
- 15) Fournisseurs qui proposent au moins 3 produits différents en stock
- 16) Donnez la liste des produits en fonction de la couleur à saisir en paramètre
- 17) Créer une vue « provinciaux » des fournisseurs non parisiens et exécutez-la**

- 18) Code, Nom et ville des fournisseurs qui proposent le produit nommé 'SALADE'
 19) Infos des produits qui ont la même couleur que le produit le moins cher (prix inconnu)
 20) Nombre total de produits en stock
 21) Donner le nom, le taux et la ville des fournisseurs ne proposant aucune remise
 22) Tableau des produits après une augmentation de 1,5% appliquée à tous les prix

Base MARKETING

23/11/2006

Structure de la base

SECTEUR	AGENT	VENTE
sectCode char(2) sectNom varchar(15)	agNum char(2) agNom varchar(15) agPrenom varchar(15) agAdresse varchar(25) agCpostal char(5) agVille varchar(20) agTel1 varchar(10) agTel2 varchar(10) agSecteur char(2)	vNum int(4) vAgent char(2) vDate date vChiffre decimal(8,2)

NB : Types PostgreSQL à convertir pour Oracle.

Edition des tables

Table secteur	Table vente																								
<table border="1"> <tr><td>s1</td><td>Valenciennes</td></tr> <tr><td>s2</td><td>Dunkerque</td></tr> <tr><td>s3</td><td>Lille</td></tr> <tr><td>s4</td><td>Roubaix</td></tr> </table>	s1	Valenciennes	s2	Dunkerque	s3	Lille	s4	Roubaix	<table border="1"> <tr><td>1</td><td>A2</td><td>2006-03-15</td><td>11542.98</td></tr> <tr><td>2</td><td>A4</td><td>2006-04-20</td><td>8452.50</td></tr> <tr><td>3</td><td>A2</td><td>2006-03-23</td><td>5362.14</td></tr> <tr><td>4</td><td>A1</td><td>2006-04-14</td><td>6500.00</td></tr> </table>	1	A2	2006-03-15	11542.98	2	A4	2006-04-20	8452.50	3	A2	2006-03-23	5362.14	4	A1	2006-04-14	6500.00
s1	Valenciennes																								
s2	Dunkerque																								
s3	Lille																								
s4	Roubaix																								
1	A2	2006-03-15	11542.98																						
2	A4	2006-04-20	8452.50																						
3	A2	2006-03-23	5362.14																						
4	A1	2006-04-14	6500.00																						

Table agent								
A1	DEMONAT	Francis	4 rue Lomprez	59300	Valenciennes	0327154284	0865321588	s1
A2	KAMEL	Jacques	86 av l'Epine	59410	Dunkerque	0328452673	0669852146	s2
A4	GERBOIS	Michel	15 place Jean Bart	59640	Grande-Synthe	0328741142	0874524867	s2
A5	LEGROS	Yves Martin	7 av du peuple belge	59000	Lille	0320147859	0845320454	s3
A6	MERLAUT	Caroline	32 bd de la libération	59780	Roubaix Cedex	0320417762	0668520233	s4

REQUÊTES SQL

Ecrire et tester les requêtes suivantes dans SQLPlus :

- 1- Nom, prénom des agents commerciaux par ordre croissant du nom ;
- 2- Coordonnées des agents et chiffre réalisé pour les ventes entre le 15 mars 2006 et le 15 avril 2006 ;
- 3- Ajouter l'agent A7 appelé BALDI Rose habitant 2, rue de la digue 62000 à CALAIS et travaillant sur le secteur de Dunkerque ;
- 4- Nombre d'agents n'habitant pas dans le Pas De Calais (62);
- 5- Nom de l'agent, nom de secteur et somme des ventes par agent ;
- 6- Afficher le cumul de toutes les ventes dans une colonne CUMUL DE VENTES ;
- 7- Numéro d'agent et nom pour les agents ayant réalisé une vente d'une valeur supérieure à la moyenne ;
- 8- Noms, secteur des agents travaillant dans le même secteur que l'agent A2 ;
- 9- Nom, prénom de l'agent ayant réalisé le meilleur chiffre et valeur de ce chiffre ;
- 10- Noms des agents ayant réalisé des chiffres supérieurs à celui de l'agent A1 ;
- 11- Modifier les informations concernant l'agent A4 qui habite désormais la ville de Calais (62000) ;
- 12- Supprimer les agents domiciliés à Roubaix ;

13- Créer par requête, une nouvelle table permettant de gérer les primes des commerciaux, préciser la clé primaire :

si ventes < 1500 => P1 = 3 % des ventes
 si 1500 < ventes < 2800 => P2 = 5.4 % des ventes
 si 2800 < ventes < 3500 => P3 = 6.5 % des ventes
 si ventes > 3500 => P4 = 10 % des ventes

14- Ajoutez une colonne **agSalaire** à la table AGENT, remplissez-la de salaires fictifs et listez toutes les données de cette table dans des colonnes en utilisant des alias et dans l'ordre décroissant des salaires ;

15- Afficher la table AGENT en prenant en compte une augmentation des salaires de 2.3 %.

Synthèse du langage SQL

Les données constituant la **base** sont organisées en tables. Une table contient des enregistrements composés de différents champs. Le type associé au champ définit le domaine sur lequel le champ pourra prendre ses valeurs (date, entier, chaîne de caractères).

Chaque **table** est désignée de manière unique par un identificateur (son nom) au sein de la base de données de même que chaque champ au sein d'une table.

Usuellement on représente une table par un tableau dont les colonnes correspondent aux champs et les lignes aux enregistrements :

La **clé primaire** d'une table est un champ (ou un ensemble de champs) qui identifie de manière unique chaque enregistrement dans la table. Chaque table devrait avoir une clé primaire.

Une **clé étrangère** dans une table est un champ (ou un ensemble de champs) qui fait référence à un champ (ou un ensemble de champs) d'une autre table (généralement la clé primaire).

Les **requêtes** permettent d'interroger une base de données et d'en modifier les informations.

Les requêtes

Définitions

Valeur NULL

Un champ qui n'est pas renseigné, donc vide, contient la valeur NULL. Cette valeur est différente de zéro et représente l'absence de valeur.

Expressions

Les expressions valides mettent en jeu des noms de champs, le mot clé « * » (qui signifie « tous les champs »), des constantes, des fonctions et des opérateurs arithmétiques classiques. Il existe des fonctions arithmétiques, de manipulations de chaînes, de dates et des fonctions agrégat. Les fonctions agrégat permettent de calculer un résultat atomique (un entier ou un réel) à partir d'un ensemble de valeurs. Les agrégats sont au nombre de 5 : COUNT, SUM, MIN, MAX, AVG.

Définition des données

Création

```
CREATE TABLE table (col1 type1, col2 type2...)
AS SELECT...
```

table est le nom donné à la nouvelle table.

col1, col2, ... sont les noms des colonnes.

type1, type2, ... sont les types des données contenues dans les colonnes.

Modification de la structure

Moyennant quelques contraintes, il est possible de modifier la structure d'une base de données existante :

```
ALTER TABLE table ADD (col1 type1, col2, type2...)
```

ajoute les colonnes spécifiées à une table existante.

```
ALTER TABLE table MODIFY (col1 type1, col2 type2...)
```

modifie la définition des colonnes spécifiées.

```
ALTER TABLE table DROP col
```

supprime la colonne *col*.

Contraintes d'intégrité

Lors de la création ou la modification de la structure d'une table, il est possible de spécifier des contraintes d'intégrité c'est-à-dire des conditions que devront vérifier les enregistrements. Elles sont de deux types :

Contrainte sur une colonne : suit la définition d'une colonne.

Contrainte sur une table (une ou plusieurs colonne) : apparaît au même niveau que les définitions des colonnes.

```
CONSTRAINT nom_contrainte contrainte
```

définit et nomme une contrainte sur une ou plusieurs colonnes.

Les types de contraintes sont :

```
PRIMARY KEY (col1, col2, ...) (contrainte sur une table)
PRIMARY KEY (contrainte sur une colonne)
```

indique la clé primaire de la table . Aucune des colonnes de cette clé ne doit avoir une valeur NULL.

```
UNIQUE (col1, col2, ...) (contrainte sur une table)
UNIQUE (contrainte sur une colonne)
```

interdit qu'une colonne (ou la concaténation de plusieurs colonnes) contienne deux valeurs identiques.

```
FOREIGN KEY (col1, col2, ...)
REFERENCES table [(col'1,col'2,...)] (contrainte sur une table)
REFERENCES table [(col1)] (contrainte sur une colonne)
```

indique que la concaténation de *col1*, *col2*,... est une clé étrangère faisant référence à la concaténation de *col'1*, *col'2*,... de table. Si *col'1*,*col'2*,... sont omises, la clé primaire de table est prise par défaut.

```
CHECK (condition)
```

donne une condition devant être vérifiée par une ou plusieurs colonnes.

Destruction

```
DROP TABLE table
```

supprime la définition d'une table et par conséquent l'ensemble des enregistrements qu'elle contient.

Interrogation d'une base

La syntaxe générale est la suivante :

```
SELECT ... FROM... WHERE... GROUP BY... HAVING... ORDER BY...
```

SELECT...

```
SELECT [DISTINCT] expr1 [[AS] nom1], expr2 [[AS] nom2],...
```

réalise une projection.

Expr1, *expr2*,... indiquent quelles expressions devront être renvoyées, par exemple des noms de champs. S'il y a une ambiguïté (cas de deux tables contenant des champs de même nom), il est nécessaire de préfixer le nom du champ par celui de la table et d'un point.

Nom1, *nom2*,... sont des noms facultatifs qui constitueront les titres des colonnes renvoyées.

Le mot clé DISTINCT permet de supprimer les doublons.

FROM...

```
FROM table1 [alias1], table2 [alias2],...
```

donne la liste des tables participant à l'interrogation. *alias1*, *alias2*,... sont des alias facultatifs attribués aux tables pour le temps de la requête. Quand une table se voit attribuer un alias, elle n'est plus reconnue sous son nom d'origine dans la requête.

WHERE...

```
WHERE prédicat
```

permet d'effectuer une restriction, c'est-à-dire de spécifier quels enregistrements sélectionner dans une table ou un produit cartésien de tables.

Prédicats simples

Un prédicat simple est la comparaison de plusieurs expressions au moyen d'un opérateur logique :

```
WHERE expr1 = / != / < / > / <= / >= expr2           (opérateurs classiques)
WHERE expr1 BETWEEN expr2 AND expr3 (appartenance à un intervalle bornes incluses)
WHERE expr1 [NOT] LIKE expr2                 (utilisation des caractères joker _ et % dans
expr2)
WHERE expr1 [NOT] IN (expr2,expr3,...)       (appartenance à la liste d'expressions)
WHERE expr1 IS [NOT] NULL                     (valeur NULL)
```

Prédicats composés

Les opérateurs logiques AND et OR permettent de combiner plusieurs prédicats. AND est prioritaire par rapport à OR mais l'utilisation de parenthèses permet de modifier l'ordre d'évaluation.

Sous-requêtes

Le critère de recherche employé dans une clause WHERE (l'expression à droite d'un opérateur de comparaison) peut être le résultat d'un SELECT. Dans le cas des opérateurs classiques, la sous-interrogation ne doit ramener qu'une ligne et une colonne. Elle peut ramener plusieurs lignes à la suite de l'opérateur [NOT] IN, ou à la suite des opérateurs classiques moyennant l'ajout d'un mot clé (ANY ou ALL).

ANY : la comparaison est vraie si elle est vraie pour au moins un élément de l'ensemble (donc fausse si l'ensemble des enregistrements est vide).

ALL : la comparaison est vraie si elle est vraie pour tous les éléments de l'ensemble (donc vraie si l'ensemble des enregistrements est vide).

```
WHERE expr1 = / != / < / > / <= / >= expr2ALL (SELECT...)
WHERE expr1 = / != / < / > / <= / >= expr2ANY (SELECT...)
```

Jointures**Jointures naturelles**

```
SELECT expr1, expr2,...
FROM table1, table2
WHERE table1.champ1 = table2.champ2
```

Lorsque la clause FROM contient plusieurs tables, on en obtient le produit cartésien. La clause WHERE permet d'effectuer une restriction. Si cette clause est une égalité, on dit que l'on réalise une jointure naturelle ou équi-jointure.

Jointures interne

```
SELECT table.expr1, alias.expr2
FROM table, table alias
WHERE table.champ1 = alias.champ2
```

Il s'agit d'une jointure naturelle mettant deux fois en jeu la même table. Ceci permet de rassembler venant d'un enregistrement d'une table avec les informations venant d'un autre enregistrement. A noter qu'il est obligatoire de spécifier deux fois la table source dans la clause FROM, en attribuant un alias à l'une des occurrences.

Jointures externes

```
SELECT expr1, expr2,...
FROM table1 LEFT / RIGHT / FULL OUTER JOIN table2
ON table1.champ1 = table2.champ2
```

Lorsqu'on effectue une jointure naturelle sur deux tables, il est possible qu'un enregistrement d'une table n'ait pas de correspondant dans l'autre. Dans ce cas, l'enregistrement en question n'est pas affiché. La jointure externe permet de résoudre ce problème par l'ajout de lignes fictives, qui réalisent la correspondance avec les

enregistrements de l'autre table n'ayant pas de correspondant réel (LEFT OUTER JOIN correspond à l'ajout d'enregistrements fictifs dans *table1*).

GROUP BY...

```
GROUP BY expr1, expr2,...
```

permet de subdiviser la table en groupes, chaque groupe étant l'ensemble des enregistrements ayant une valeur commune pour les expressions spécifiées. Les champs de la clause SELECT doivent alors être des fonctions agrégat ou des expressions figurant dans la clause GROUP BY.

HAVING...

```
HAVING prédicat
```

sert à préciser quels groupes doivent être sélectionnés. Cette clause se place après GROUP BY et le prédicat ne peut porter que sur des fonctions agrégat ou des expressions figurant dans la clause GROUP BY.

ORDER BY...

```
ORDER BY expr1 [DESC], expr2 [DESC],...
```

classe les enregistrements retournés selon l'ordre croissant de *expr1* puis *expr2*. La clause facultative DESC inverse l'ordre de classement. Pour préciser lors d'un tri sur quelle expression va porter le tri, il est possible de donner sa position dans la liste des expressions de la clause SELECT ou encore le nom qu'on lui a attribué.

Les opérateurs ensemblistes

```
requête1 UNION / INTERSECT / MINUS requête2 ...
```

permettent de réaliser des opérations ensemblistes sur les résultats de plusieurs interrogations. Les champs renvoyés par les requêtes impliquées doivent être identiques. Les opérations sont évaluées de gauche à droite mais l'utilisation de parenthèses permet de modifier cet ordre.

Manipulation des données

Insertion

```
INSERT INTO table [(col1, col2,...)] VALUES (val1, val2,...)
INSERT INTO table [(col1, col2,...)] SELECT...
```

insère un nouvel enregistrement dans *table*. Si (*col1*, *col2*,...) est omise, l'ordre utilisé par défaut est celui spécifié lors de la création de la table. A l'inverse, si cette liste est donnée, les colonnes n'y figurant pas auront la valeur NULL.

Mise à jour

```
UPDATE table SET col1 = expr1, col2 = expr2,... WHERE prédicat
UPDATE table SET (col, col2,...) = (SELECT...) WHERE prédicat
```

modifie les enregistrements de *table* qui vérifient *prédicat*. Si *prédicat* est omis, **tous** les enregistrements sont mis à jour.

Suppression

```
DELETE FROM table WHERE prédicat
```

supprime les enregistrements de *table* qui vérifient *prédicat*. Si *prédicat* est omis, **tous** les enregistrements sont supprimés.

Contrôle de l'accès aux données

```
GRANT privilege ON table TO utilisateur [WITH GRANT OPTION]
```

permet au propriétaire de *table* de donner à *utilisateur* des droits d'accès sur celle-ci. Si l'option WITH GRANT OPTION est spécifiée, *utilisateur* pourra à son tour transmettre ces droits.

```
REVOKE privilege ON table FROM utilisateur
```

permet de reprendre un privilège à un utilisateur.

Vues : un ordre SELECT renvoie une table temporaire. Il est cependant possible d'enregistrer le résultat d'un tel ordre dans une vue. Ceci permet de découpler la structure réelle de la base de données de ce que voient les utilisateurs.

Transactions : Une transaction est un ensemble de modifications de la base qui forment un tout. Ces modifications doivent être effectuées entièrement ou pas du tout sous peine de laisser la base dans un état incohérent. Une commande (COMMIT) permet de valider la session en cours, une autre (ROLLBACK) permet de l'annuler.

Procédures stockées : Une procédure stockée est un programme qui comprend des instructions SQL pré compilées et qui est enregistré dans la base de données. Ceci permet d'accélérer les requêtes puisque les optimisations ont lieu lors de l'enregistrement de celles-ci.

Triggers : Ce sont des procédures stockées déclenchées automatiquement par des événements liés à des actions sur la base.