

BDA

COMPTE RENDU TP01

Création et manipulation de la base de données

Introduction :

Le sujet de ce TP est le rappel des notions de l'année dernière.

L'objectif est de répondre à différentes demandes avec les notions acquises au cours de notre première année.

Pour ce faire nous avons utilisé un logiciel : *Oracle SQL Developer*.

Création de la BD initiale :

- (1) Exécuter le script createbase.sql pour créer les tables
- (2) Exécuter le script insertbase.sql pour insérer les tuples
- (3) Vérifier le contenu des tables avec la commande suivante

```
Select * From Type_Location ;
```

```
Select * From Location ;
```

```
Select * From Client ;
```

```
Select * From Louer;
```

```
Select * From Participer;
```

Feuille de calcul		Query Builder		
		Select * From Type_Location ;		
Sortie de script x		Tâche terminée en 0,062 secondes		
NUMTYPELOC	TYPELOC	NOMLOC	TAILLE	CAPACITE
1	BUNGALOW	Caribe Club	52	4 80,
2	BUNGALOW	Caribe	52	4 67,
3	BUNGALOW	Bahia	55	4
4	BUNGALOW	Oasis	35	4 41,
5	BUNGALOW	Yucatan	55	4 41,
6	BUNGALOW	Costal Azul	25	4 44,
7	BUNGALOW	Eden	50	6
8	MOBIL-HOME	Paradise	23	4
9	MOBIL-HOME	Siesta	20	2
10	PARCELLE	Standard	70	2
11	PARCELLE	Premium	80	2
NUMTYPELOC	TYPELOC	NOMLOC	TAILLE	CAPACITE
12	PARCELLE	Premium Plus	90	2

12 lignes sélectionnées.

A partir du dictionnaire de données Oracle, affichez les métadonnées suivantes :

- à partir d'USER_OBJECTS, les tables (OBJECT_NAME) et leur date de création (CREATED)

```
Select OBJECT_NAME From USER_OBJECTS ;
```

```
Select CREATED From USER_OBJECTS ;
```

The image shows two side-by-side screenshots of the Oracle Query Builder interface. The left window displays the query `Select OBJECT_NAME From USER_OBJECTS;` and its results in a table with 11 rows. The right window displays the query `Select CREATED From USER_OBJECTS;` and its results in a table with 11 rows, all showing the date 22/09/17. Both windows show a status bar indicating the task was completed in 0.078 seconds.

OBJECT_NAME
TYPE_LOCATION
PK_TYPE_LOCATION
UK_TYPE_LOCATION
LOCATION
PK_LOCATION
CLIENT
PK_CLIENT
LOUER
PK_LOUER
PARTICIPER
PK_PARTICIPER

11 lignes sélectionnées.

CREATED
22/09/17
22/09/17
22/09/17
22/09/17
22/09/17
22/09/17
22/09/17
22/09/17
22/09/17
22/09/17
22/09/17

11 lignes sélectionnées.

– à partir d'USER_TAB_COLUMNS, les attributs (COLUMN_NAME, DATA_TYPE, DATA_LENGTH, DATA_PRECISION) ordonnés par table (TABLE_NAME).

```
Select COLUMN_NAME, DATA_TYPE, DATA_LENGTH, DATA_PRECISION  
FROM USER_TAB_COLUMNS  
order by TABLE_NAME;
```

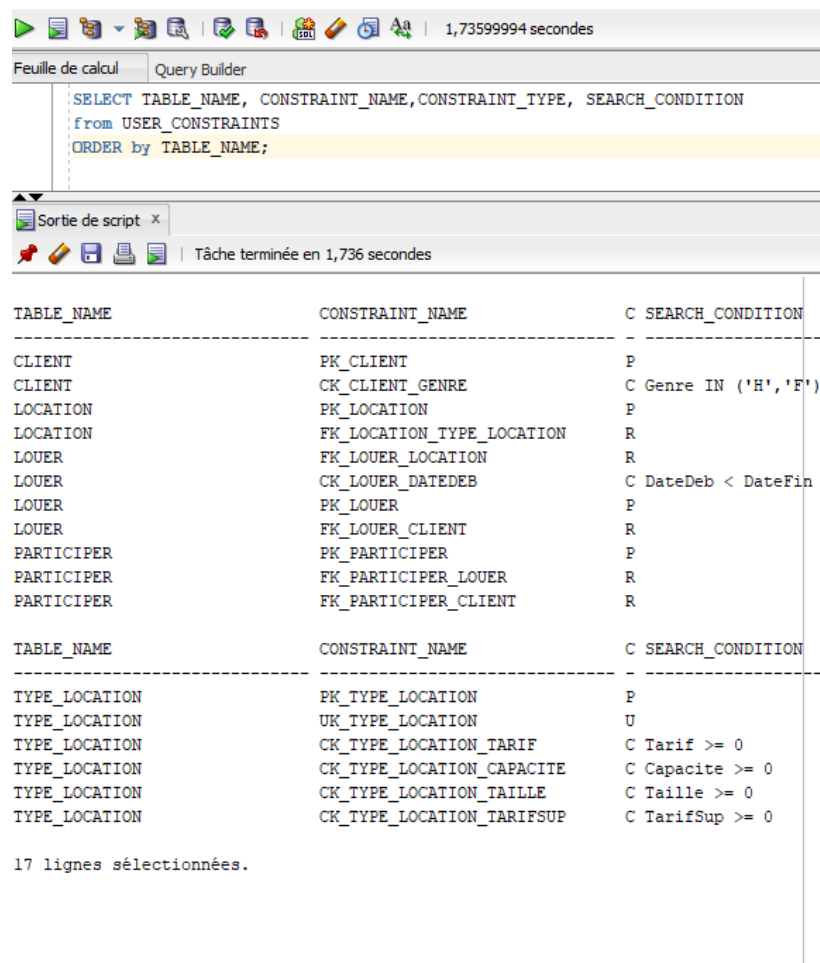
Feuille de calcul	
Query Builder	
<pre>select COLUMN_NAME, DATA_TYPE, DATA_LENGTH, DATA_PRECISION FROM USER_TAB_COLUMNS order by TABLE_NAME;</pre>	
Sortie de script x	
Tâche terminée en 0,36 secondes	
COLUMN_NAME	DATA_TYPE
NOM	VARCHAR2
NUMCLI	NUMBER
PRENOM	VARCHAR2
GENRE	CHAR
DATEN	DATE
PAYS	CHAR
NUMLOC	NUMBER
NUMERO	NUMBER
NUMTYPELOC	NUMBER
NUMLOC	NUMBER
NUMCLI	NUMBER
COLUMN_NAME	DATA_TYPE
DATEDEB	DATE
DATEFIN	DATE
NUMCLI	NUMBER
NUMCLIP	NUMBER
DATEDEB	DATE
NUMLOC	NUMBER
CAPACITE	NUMBER
TARIF	NUMBER
TAILLE	NUMBER
NOMLOC	VARCHAR2
TARIFSUP	NUMBER
COLUMN_NAME	DATA_TYPE
NUMTYPELOC	NUMBER
TYPELOC	VARCHAR2
24 lignes sélectionnées.	

– à partir d'USER_CONSTRAINTS, les contraintes d'intégrités (TABLE_NAME, CONSTRAINT_NAME, CONSTRAINT_TYPE, SEARCH_CONDITION) ordonnées par table (TABLE_NAME), puis par type de contrainte.

```
SELECT TABLE_NAME, CONSTRAINT_NAME, CONSTRAINT_TYPE,
SEARCH_CONDITION

from USER_CONSTRAINTS

ORDER by TABLE_NAME ;
```



TABLE_NAME	CONSTRAINT_NAME	C SEARCH_CONDITION
CLIENT	PK_CLIENT	P
CLIENT	CK_CLIENT_GENRE	C Genre IN ('H','F')
LOCATION	PK_LOCATION	P
LOCATION	FK_LOCATION_TYPE_LOCATION	R
LOUER	FK_LOUER_LOCATION	R
LOUER	CK_LOUER_DATEDEB	C DateDeb < DateFin
LOUER	PK_LOUER	P
LOUER	FK_LOUER_CLIENT	R
PARTICIPER	PK_PARTICIPER	P
PARTICIPER	FK_PARTICIPER_LOUER	R
PARTICIPER	FK_PARTICIPER_CLIENT	R

TYPE_LOCATION	PK_TYPE_LOCATION	P
TYPE_LOCATION	UK_TYPE_LOCATION	U
TYPE_LOCATION	CK_TYPE_LOCATION_TARIF	C Tarif >= 0
TYPE_LOCATION	CK_TYPE_LOCATION_CAPACITE	C Capacite >= 0
TYPE_LOCATION	CK_TYPE_LOCATION_TAILLE	C Taille >= 0
TYPE_LOCATION	CK_TYPE_LOCATION_TARIFSUP	C TarifSup >= 0

17 lignes sélectionnées.

```

SELECT TABLE_NAME, CONSTRAINT_NAME, CONSTRAINT_TYPE,
SEARCH_CONDITION

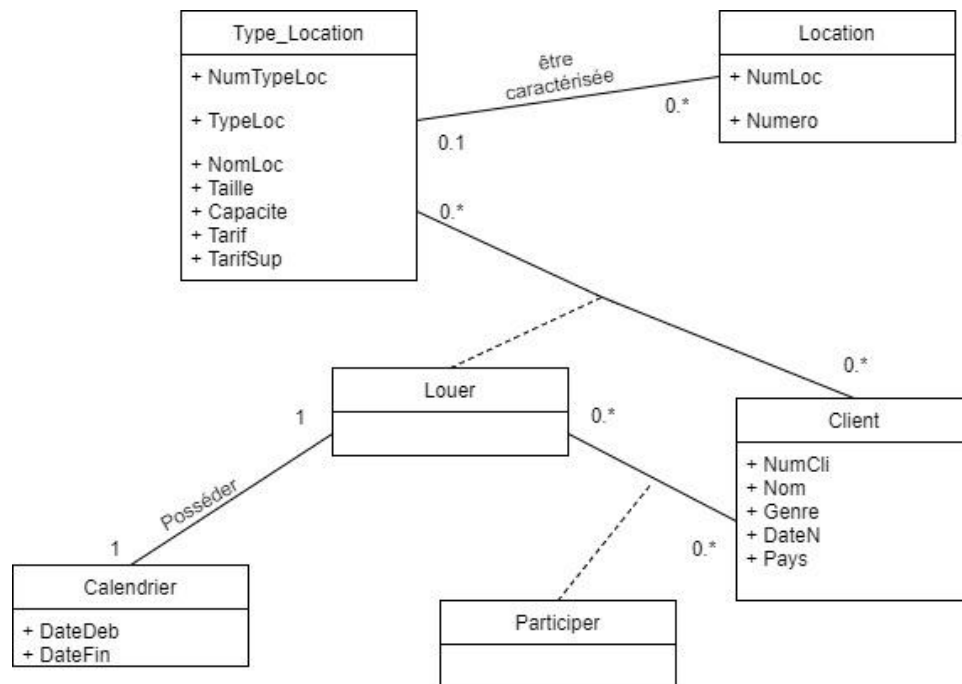
from USER_CONSTRAINTS

ORDER by CONSTRAINT_TYPE ;

```

Feuille de calcul		Query Builder
		<pre> SELECT TABLE_NAME, CONSTRAINT_NAME, CONSTRAINT_TYPE, SEARCH_CONDITION from USER_CONSTRAINTS ORDER by CONSTRAINT_TYPE; </pre>
Sortie de script x		Tâche terminée en 0,766 secondes
TABLE_NAME	CONSTRAINT_NAME	C SEARCH_CONDITION
TYPE_LOCATION	CK_TYPE_LOCATION_TAILLE	C Taille >= 0
TYPE_LOCATION	CK_TYPE_LOCATION_CAPACITE	C Capacite >= 0
TYPE_LOCATION	CK_TYPE_LOCATION_TARIF	C Tarif >= 0
TYPE_LOCATION	CK_TYPE_LOCATION_TARIFSUP	C TarifSup >= 0
CLIENT	CK_CLIENT_GENRE	C Genre IN ('H','F')
LOUER	CK_LOUER_DATEDEB	C DateDeb < DateFin
LOUER	PK_LOUER	P
PARTICIPER	PK_PARTICIPER	P
CLIENT	PK_CLIENT	P
LOCATION	PK_LOCATION	P
TYPE_LOCATION	PK_TYPE_LOCATION	P
TABLE_NAME	CONSTRAINT_NAME	C SEARCH_CONDITION
LOCATION	FK_LOCATION_TYPE_LOCATION	R
LOUER	FK_LOUER_LOCATION	R
PARTICIPER	FK_PARTICIPER_CLIENT	R
LOUER	FK_LOUER_CLIENT	R
PARTICIPER	FK_PARTICIPER_LOUER	R
TYPE_LOCATION	UK_TYPE_LOCATION	U
17 lignes sélectionnées.		

Proposer un diagramme conceptuel (DC UML) du schéma de la BD.



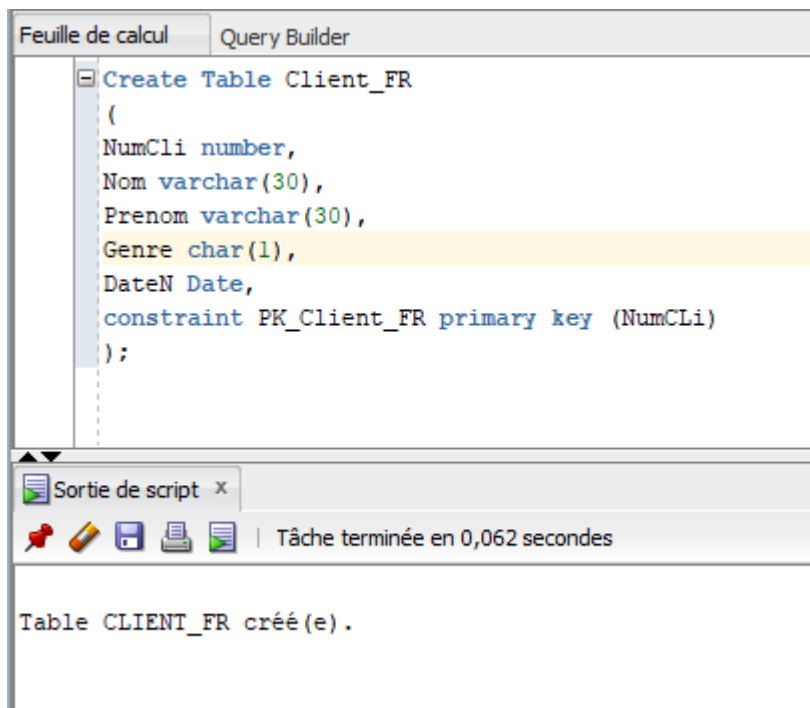
SQL/LDD :

Créer une nouvelle table dans la BD en ajoutant :

CLIENT_FR(NumCli, Nom, Prenom, Genre, DateN)

- NumCli : identifiant du client
- Nom : nom du client
- Prenom : prénom du client
- Genre : genre du client (Homme ou Femme)
- DateN : date de naissance du client (Jour-Mois-Année)

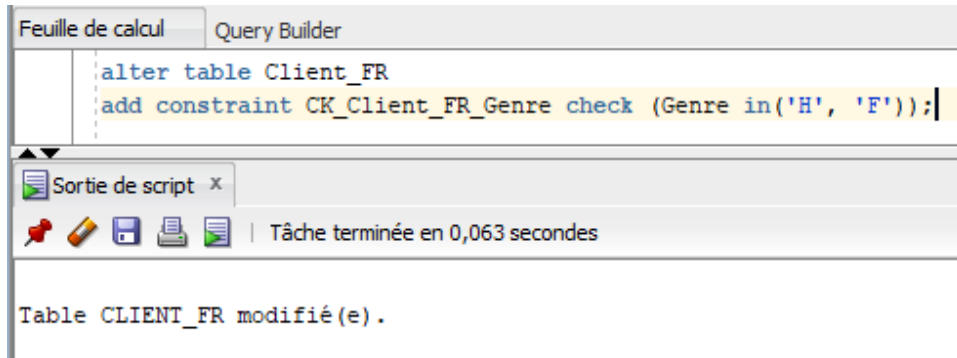
```
Create Table Client_FR  
(  
  NumCli number,  
  Nom varchar (30),  
  Prenom varchar (30),  
  Genre char (1),  
  DateN Date,  
  constraint PK_Client_FR primary key (NumCLI)  
);
```



Ajouter dans cette relation une contrainte d'intégrité qui limite le genre aux valeurs 'H' et 'F' respectivement pour les hommes et les femmes.

Alter table Client_FR

add constraint CK_Client_FR_Genre check (Genre in('H', 'F'));



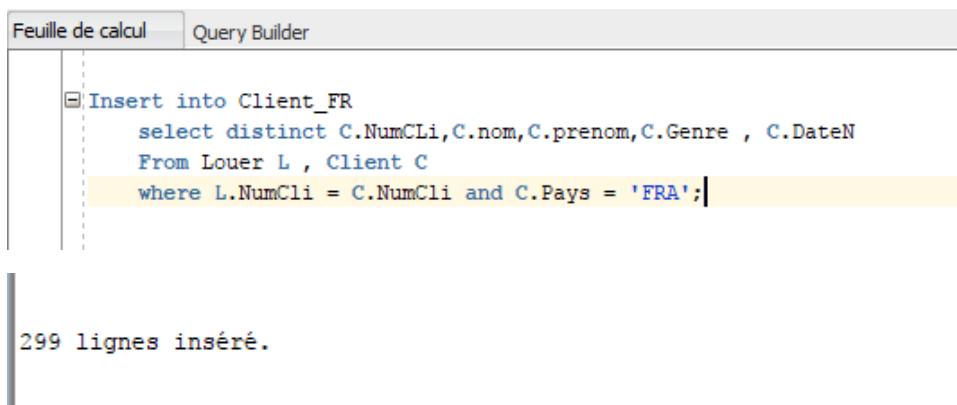
Insérer dans CLIENT_FR tous les clients français ayant effectué une location (LOUER).

INSERT INTO CLIENT_FR

SELECT FROM ... WHERE ;

Insert into Client_FR

```
select distinct C.NumCli,C.nom,C.prenom,C.Genre, C.DateN
From Louer L, Client C
where L.NumCli = C.NumCli and C.Pays = 'FR';
```



Vérifier le contenu de la table.

```
Select * from Client_FR;
```

Feuille de calcul

Query Builder

Select * from Client_FR;

Sortie de script x

Tâche terminée en 0,343 secondes

24	Payet	Félix	H	23/01/86
26	Dubois	Félix	H	23/02/96
105	Morel	Chloé	F	28/07/98
113	Leroux	Philippe	H	16/09/79
116	Meunier	Alicia	F	19/04/85
123	Leclercq	Gabrielle	F	11/03/55
68	Morel	Antoine	H	04/01/69
NUMCLI	NOM	PRENOM	G	DATEN
71	Picard	Nathan	H	30/04/82
77	Guérin	Emy	F	03/01/08
144	Marie	Loïc	H	28/06/66
151	Bonnet	Maxime	H	29/12/88
174	Dufour	Simon	H	11/11/94
176	Bertrand	Camille	F	30/10/85
177	Leroy	Océane	F	01/01/70
181	Leroy	Maya	F	02/06/56
182	Bernard	Nathan	H	19/02/09
261	Robin	Alex	H	16/08/05
271	Morin	David	H	07/11/05
NUMCLI	NOM	PRENOM	G	DATEN
282	Rivière	Émilie	F	07/06/94

Supprimer la table.

```
Drop Table Client_FR;
```

Feuille de calcul		Query Builder	
	Drop Table Client_FR;		
Sortie de script x			
Tâche terminée en 0,094 se			
Table CLIENT_FR supprimé(e).			

SQL/LMD :

Compléter la BD en ajoutant vous ajoutant comme nouveau client.

Insertion dans CLIENT (NumCli, Nom, Prenom, Genre, DateN, Pays)

```
Insert into Client  
values (2224, 'Deliot', 'Maxence', 'H', '01/05/1998', 'FR') ;
```

The screenshot shows a 'Query Builder' window with the following SQL query:

```
Insert into Client values(2224, 'Deliot', 'Maxence', 'H', '01/05/1998', 'FRA');  
select * from CLIENT where NUMCLI=2224;
```

Below the query, a status bar indicates 'Tâche terminée en 0,078 secondes'. The results pane shows '1 ligne inséré.' and a table with the following data:

NUMCLI	NOM	PRENOM	G	DATEN	PAY
2224	Deliot	Maxence	H	01/05/98	FRA

Effectuer une location pour vous-même d'un bungalow 'Caribe Club' pour 1 semaine du 28-oct au 4-nov 2017.

```
Insert into Location values (5555,02,1);
```

```
Insert into Louer values (2222,5555, '28/10/2017', '4/11/2017') ;
```

The screenshot shows a 'Query Builder' window with the following SQL queries:

```
Insert into Location values(5555,20,1);  
Insert into Louer values (2222,5555, '28/10/2017', '4/11/2017');
```

Below the queries, a status bar indicates 'Tâche terminée en 0,094 secondes'. The results pane shows '1 ligne inséré.' twice, indicating that both the location and the rental record were successfully inserted.

Ajouter les colonnes suivantes :

– dans la table CLIENT,

NBL ENTIER (5) : nombre de locations (dans LOUER)

NBJ ENTIER(5) : nombre de jours loués (dans LOUER)

– dans la table LOCATION,

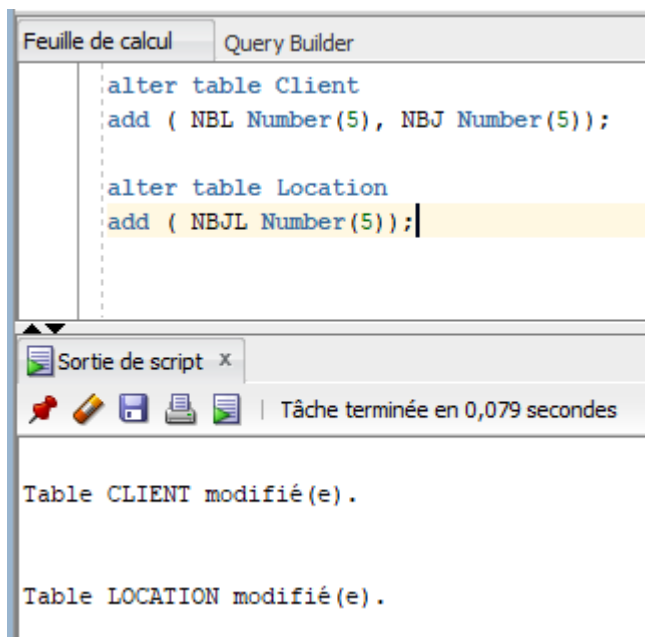
NBJL ENTIER (5) : nombre de jours libres sur toute l'année

Alter table Client

```
add ( NBL Number(5), NBJ Number(5));
```

Alter table Location

```
add ( NBJL Number(5));
```



Mettre à jour ces nouvelles colonnes

Update Client

Set NBL =

```
(Select count (*) from Louer where Client.numCLi= Louer.NUMCLI group by NumCli);
```

Update Client

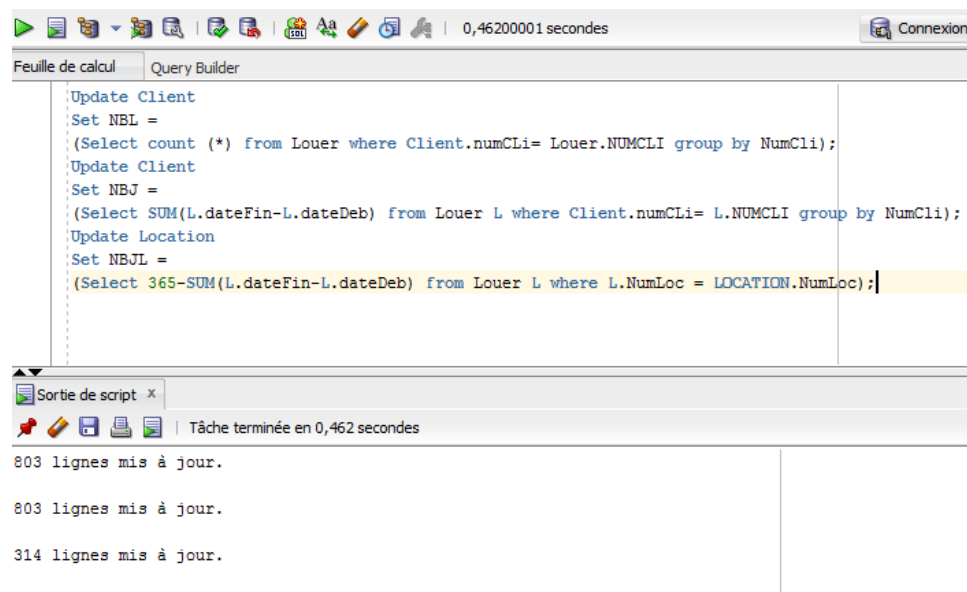
Set NBJ =

```
(Select SUM(L.dateFin-L.dateDeb) from Louer L where Client.numCLi= L.NUMCLI group by NumCli);
```

Update Location

Set NBJL =

```
(Select 365-SUM(L.dateFin-L.dateDeb) from Louer L where L.NumLoc = LOCATION.NumLoc);
```



Conclusion :

A l'issue de ce TP, nous avons donc revu les **différentes grandes notions du SQL** tel que la **création de table, de contraintes et l'UPDATE de celles-ci** par exemple.