

# COMPTE RENDU BDA

## TP02

Création de  
traitements dans la  
base de données

# Sommaire

## Table des matières

1-Introduction.....	3
1) Création de la procédure AddLouer.....	4
2) Modification de la procédure AddLouer.....	10

## *1-Introduction*

Le sujet de ce TP concerne les rappels sur les blocs PL/SQL, des procédures et la gestion d'erreur.

L'objectif est de créer une **procédure permettant l'insertion d'une nouvelle location**. Ici on s'assurera de **respecter les propriétés ACID de la transaction**.

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

## 2-Les requêtes

### 1) Création de la procédure AddLouer

Créez une procédure stockée en PL/SQL conforme aux spécifications suivantes :

Entrée :

- Numéro de client
- Numéro de location
- Date de début de location
- Date de fin de location

Actions :

- Insertion d'une nouvelle location
- Mise à jour du client (NBL, NBJ)
- Mise à jour de la location (NB JL)

Vous veillerez à assurer les propriétés ACID de la transaction avec les instructions COMMIT et ROLLBACK.

Gestion d'erreurs :

- Client inconnu
- Location inconnue
- Dates invalides (Date début > Date de fin)
- Dates hors ouverture (la saison d'ouverture du 15 Avril au 15 Octobre

chaque année)

- Location indisponible sur la période demandée

Vous utiliserez des PRAGMA pour les erreurs client inconnu et location inconnue.

```
create or replace Procedure AddLouer(pNumC Client.NumCli%TYPE,pNumL
LOCATION.NUMLOC%TYPE, pDateD LOUER.DATEDEB%TYPE, pDateF LOUER.DATEFIN%TYPE)
AS
vDateFinLoc LOUER.DATEFIN%TYPE;
vClientTest Number;
Inconnu EXCEPTION;
PRAGMA EXCEPTION_INIT(Inconnu,-2291);
DateFausse EXCEPTION;
DateInd EXCEPTION;
DateHorL EXCEPTION;

BEGIN
--Deb>Fin
if (pDateD>pDateF) then
    Raise DateFausse;
```

```

END IF;

--dates indisponibles

select count(Louer.NumCli) into vClientTest from Louer where
Louer.NUMLOC=pNumL and ((pDateD between Louer.DateDeb and LOUER.DATEFIN)

or (pDateF between Louer.DateDeb and LOUER.DATEFIN));

if(vClientTest>0) then
    raise DateInd;
End if;

--dates hors limites

if( pDateD<='15/04/2017' or pDateD>='15/10/2017') then
    raise dateHorL;
elseif(pDateF<='15/04/2017' or pDateF>='15/10/2017') then
    raise dateHorL;
END IF;

Insert into LOUER values (pNUMc,pNumL,pDateD,pDateF);

--Mises à jour demandées

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);

commit ;

```

```
EXCEPTION

    WHEN Inconnu Then

        DBMS_OUTPUT.PUT_LINE('Numero de client inconnu ou de location
inconnu');

        ROLLBACK;

    WHEN DateFausse Then

        DBMS_OUTPUT.PUT_LINE('Date de début supérieur à celle de fin');

        ROLLBACK;

    WHEN DateInd Then

        DBMS_OUTPUT.PUT_LINE('La location n est pas dispo');

        ROLLBACK;

    WHEN DateHorL Then

        DBMS_OUTPUT.PUT_LINE('Dates hors limites');

        ROLLBACK;

END;
```

Regardons les disponibilités de la location 1 pour la suite de test :

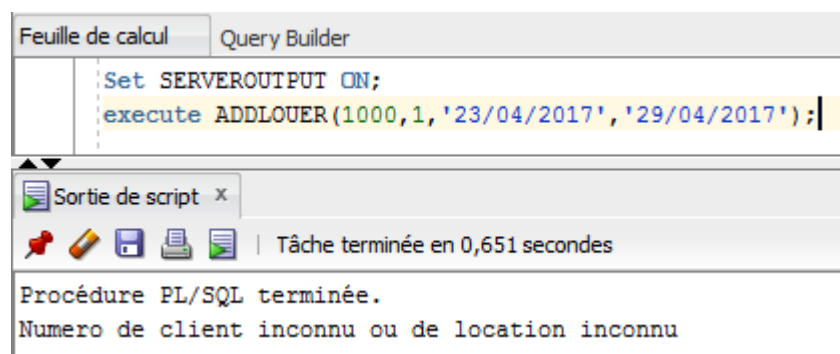
2	307	1	21/04/17	22/04/17
3	789	1	19/05/17	27/05/17

Nous allons donc entrer les dates : 23/04/17 et 29/04/2017 par exemple.

Test n°1 : Le client n'est pas connu :

```
Set SERVEROUTPUT ON;
```

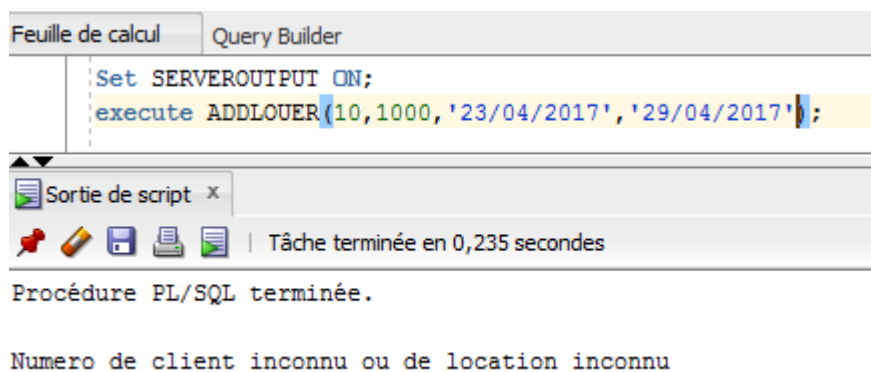
```
execute ADDLOUER(1000,1,'23/04/2017','29/04/2017');
```



Test n°2 : La location n'est pas connue :

```
Set SERVEROUTPUT ON;
```

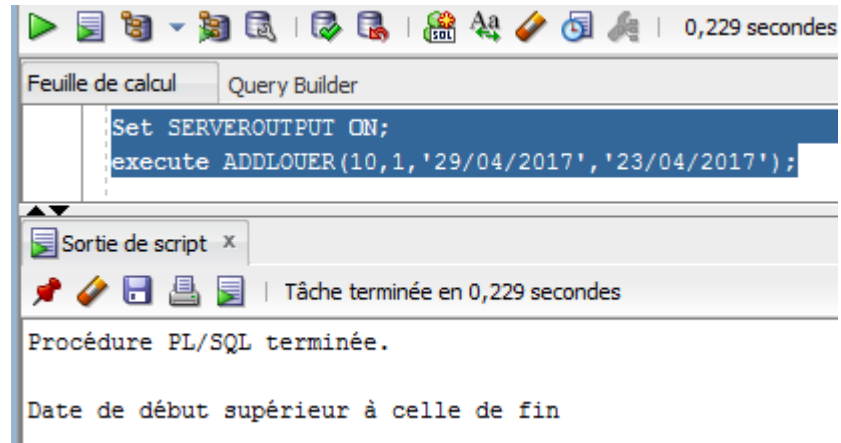
```
execute ADDLOUER(10,1000,'23/04/2017','29/04/2017');
```



Test n°3 : La date de début est supérieure à celle de fin :

```
Set SERVEROUTPUT ON;
```

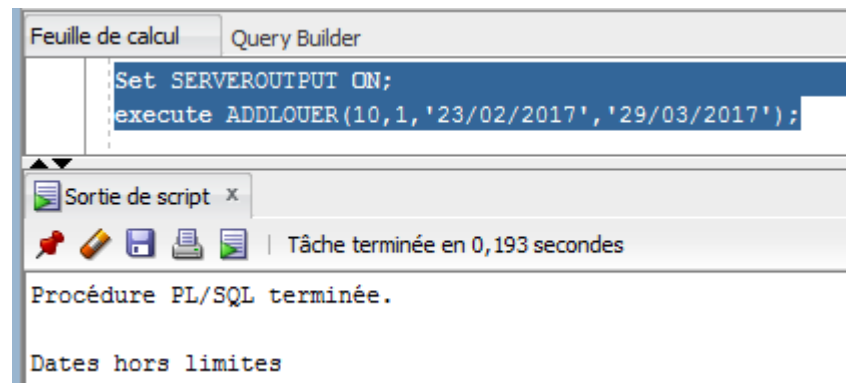
```
execute ADDLOUER(10,1,'29/04/2017','23/04/2017');
```



Test n°4 : Les dates sont hors limites :

```
Set SERVEROUTPUT ON;
```

```
execute ADDLOUER(10,1,'23/02/2017','29/03/2017');
```

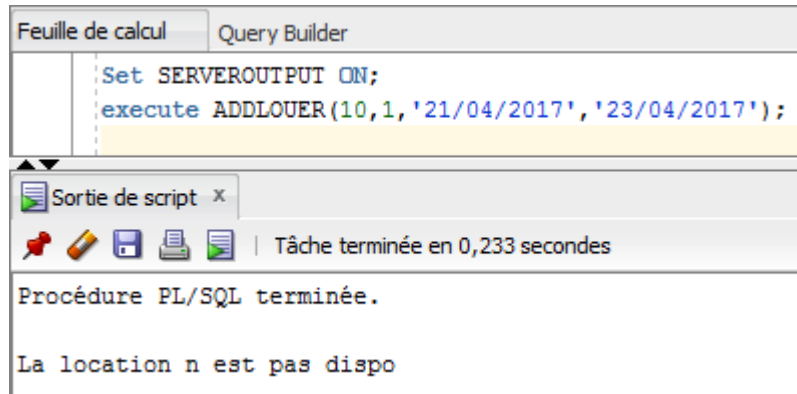




Test n°5 : Les dates sont indisponibles :

```
Set SERVEROUTPUT ON;
```

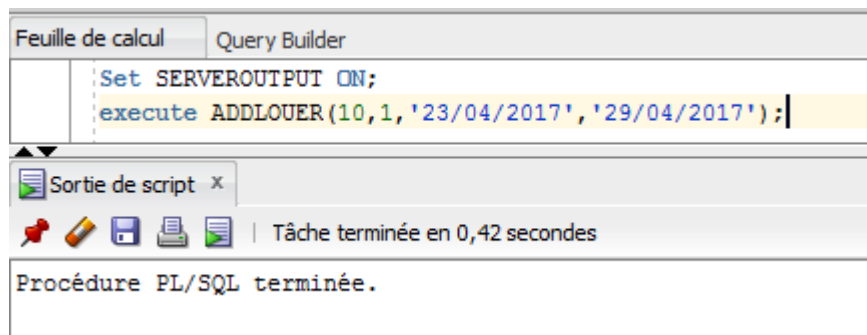
```
execute ADDLOUER(10,1,'21/04/2017','23/04/2017');
```



Test n°6 : Bonne insertion :

```
Set SERVEROUTPUT ON;
```

```
execute ADDLOUER(10,1,'23/04/2017','29/04/2017');
```



16	600	1 23/09/17	06/10/17
17	215	1 06/10/17	07/10/17
18	91	1 07/10/17	10/10/17
19	763	1 10/10/17	15/10/17
20	10	1 23/04/17	29/04/17

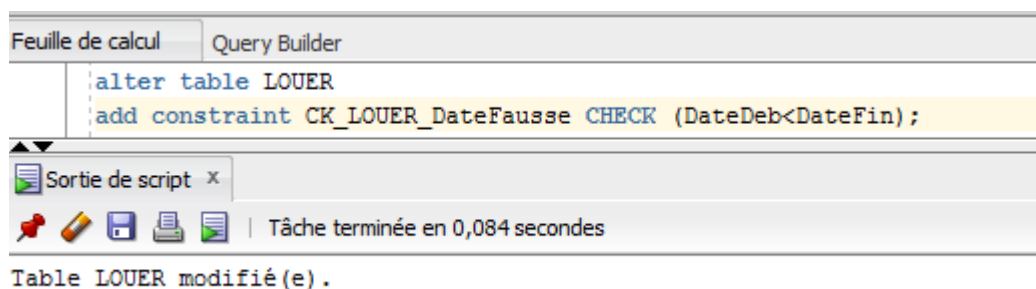
## 2) Modification de la procédure AddLouer

### 1) Mettez en place les contraintes CHECK équivalentes aux exceptions

- Dates invalides (Date début > Date de fin)
- Dates hors ouverture (la saison d'ouverture du 15 Avril au 15 Octobre chaque année)

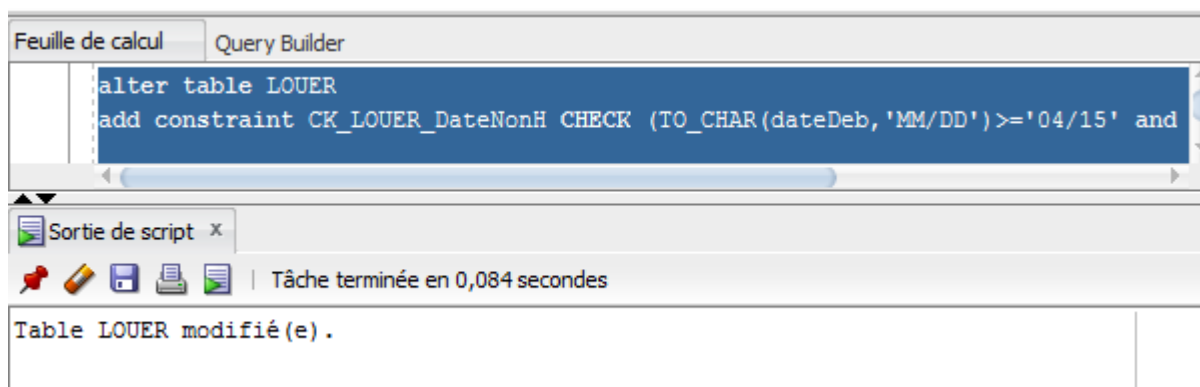
```
alter table LOUER
```

```
add constraint CK_LOUER_DateFausse CHECK (DateDeb<DateFin);
```



```
alter table LOUER
```

```
add constraint CK_LOUER_DateNonH CHECK
(TO_CHAR(dateDeb, 'MM/DD') >= '04/15' and
TO_CHAR(dateFin, 'MM/DD') <= '10/15');
```



**2) Modifiez la procédure stockée en conséquence, en utilisant un PRAGMA sur la contrainte CHECK.**

```
create or replace Procedure AddLouer(pNumC Client.NumCli%TYPE,pNumL LOCATION.NUMLOC
%TYPE, pDateD LOUER.DATEDEB%TYPE, pDateF LOUER.DATEFIN%TYPE) AS
vDateFinLoc LOUER.DATEFIN%TYPE;
vClientTest Number;
DateFausse EXCEPTION;
DateInd EXCEPTION;
CheckDate Exception;
PRAGMA EXCEPTION_INIT (CheckDate, -2290);
Inconnu EXCEPTION;
PRAGMA EXCEPTION_INIT(Inconnu,-2291);
BEGIN

--dates indisponibles

select count(Louer.NumCli) into vClientTest from Louer where Louer.NUMLOC=pNumL and ((pDateD
between Louer.DateDeb and LOUER.DATEFIN)
or (pDateF between Louer.DateDeb and LOUER.DATEFIN));
if(vClientTest>0) then
raise DateInd;
End if;

--select DateFin into vDateFinLoc From LOUER where pNumL=NUMLOC;
Insert into LOUER values(pNUMC,pNumL,pDateD,pDateF);

--Mises à jour demandées
Update Client
Set NBL =
```

```

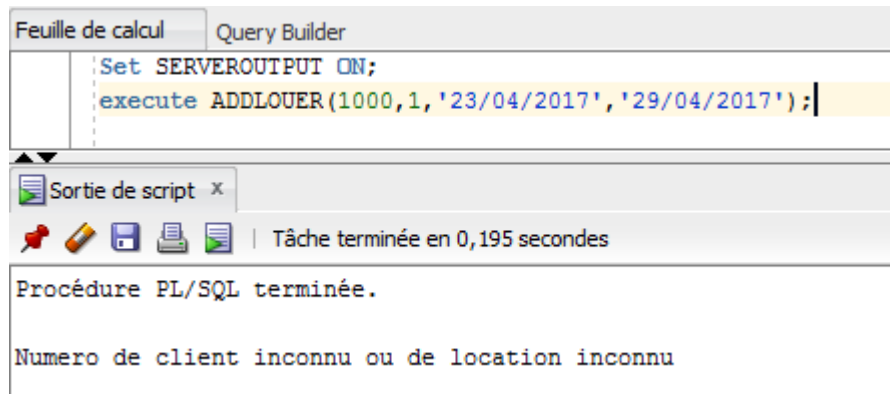
(Select count (*) from Louer where Client.numCli= Louer.NUMCLI group
by NumCli);
Update Client
Set NBJ =
(Select SUM(L.dateFin-L.dateDeb) from Louer L where Client.numCli=
L.NUMCLI group by NumCli);
Update Location
Set NBJL =
(Select 365-SUM(L.dateFin-L.dateDeb) from Louer L where L.NumLoc =
LOCATION.NumLoc);
EXCEPTION
WHEN Inconnu Then
    DBMS_OUTPUT.PUT_LINE('Numero de client inconnu ou de location inconnu');
    ROLLBACK;
WHEN CheckDate Then
    DBMS_OUTPUT.PUT_LINE('Erreur de contrainte check');
    ROLLBACK;
WHEN DateInd Then
    DBMS_OUTPUT.PUT_LINE('La location n est pas dispo');
    ROLLBACK;
END;

```

Test n°1 : Le client n'est pas connu :

```
Set SERVEROUTPUT ON;
```

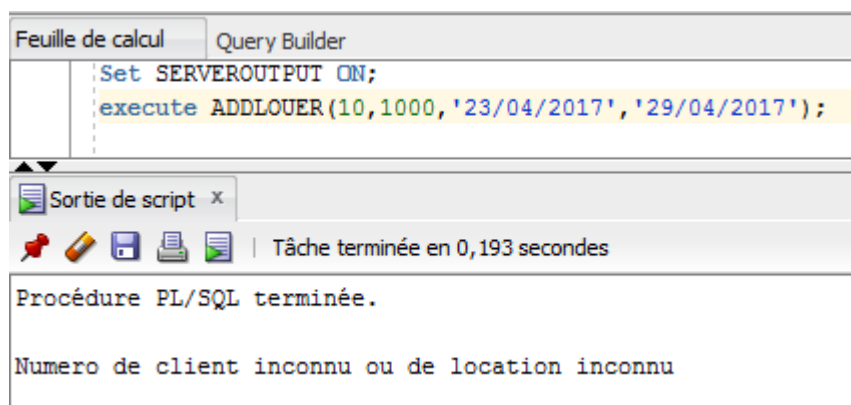
```
execute ADDLOUER(1000,1,'23/04/2017','29/04/2017');
```



Test n°2 : La location n'est pas connue :

```
Set SERVEROUTPUT ON;
```

```
execute ADDLOUER(10,1000,'23/04/2017','29/04/2017');
```



Test n°3 : La date de début est supérieure à celle de fin :

```
Set SERVEROUTPUT ON;
```

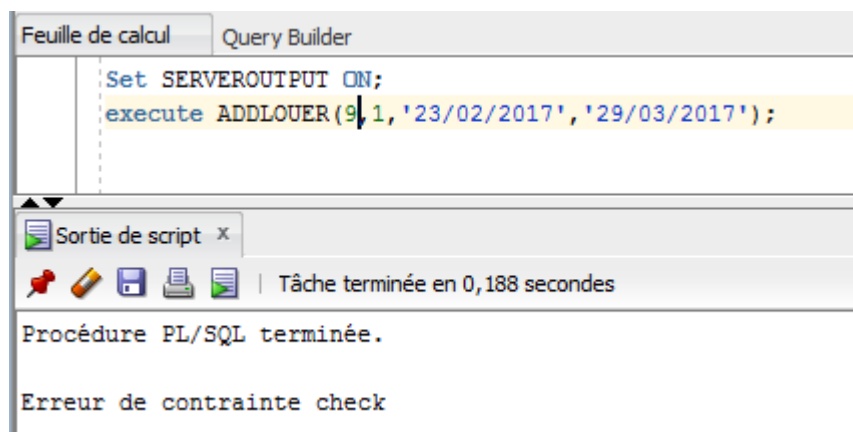
```
execute ADDLOUER(9,1,'26/04/2017','25/04/2017');
```



Test n°4 : Les dates sont hors limites :

```
Set SERVEROUTPUT ON;
```

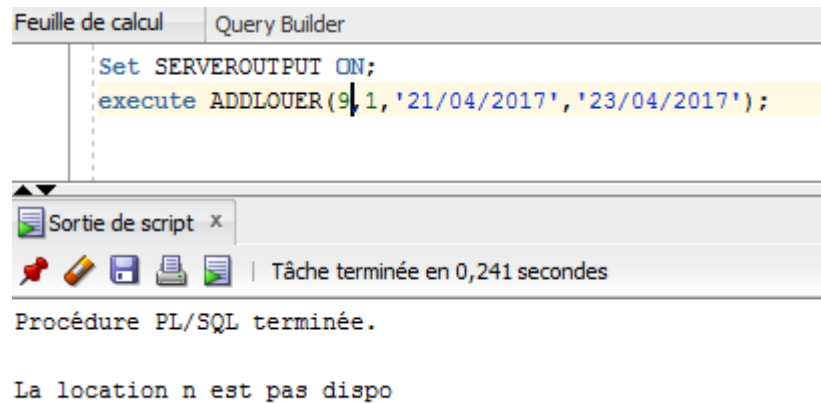
```
execute ADDLOUER(10,1,'23/02/2017','29/03/2017');
```



Test n°5 : Les dates sont indisponibles :

```
Set SERVEROUTPUT ON;
```

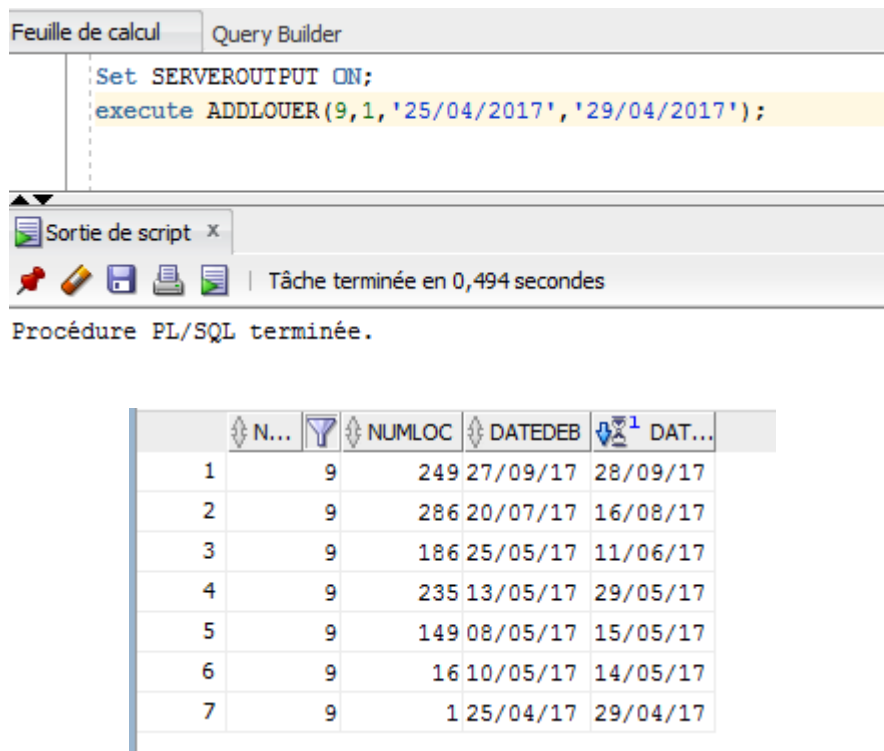
```
execute ADDLOUER(10,1,'21/04/2017','23/04/2017');
```



Test n°6 : Bonne insertion :

```
Set SERVEROUTPUT ON;
```

```
execute ADDLOUER(10,1,'25/04/2017','29/04/2017');
```



### ***3-Conclusion :***

A l'issue de ce TP, nous avons donc remis **en mémoire les règles de transaction** et certaines **notions de gestion d'erreur**.