

TP : Commit et Rollback

Réalisé par : BOUBAKER Siwar

Définitions :

COMMIT en SQL est un langage de contrôle de transaction qui est utilisé pour enregistrer en permanence les modifications effectuées dans la transaction dans des tables / bases de données. La base de données ne peut pas retrouver son état antérieur après son exécution.

ROLLBACK en SQL est un langage de contrôle transactionnel qui est utilisé pour annuler les transactions qui n'ont pas été enregistrées dans la base de données. La commande n'est utilisée que pour annuler les modifications depuis le dernier COMMIT.

Connexion du Python avec SQL Server et affichage du résultat :

```
Entrée [1]: import pyodbc
conn = pyodbc.connect('Driver={SQL Server};'
                    'Server=.';
                    'Database=WideWorldImportersDW;'
                    'Trusted_Connection=yes;')

cursor = conn.cursor()
cursor.execute("select * from Dimension.City where [City Key]=7 ")

for row in cursor:
    print(row)

(7, 5456, 'Carrollton', 'Georgia', 'United States', 'North America', 'Southeast', 'Americas', 'Northern America', b'\xe6\x10\x00\x00\x01\x0cPI\xe7\rA\xca@B\x00\x153\xe7DU\xc0', 24388, '2013-01-01 00:00:00.000000', '2013-01-01 00:02:00.000000', 1)
```

Affichage de résultat avant la transaction : utilisant SQL

```

use WideWorldImportersDW
select * from Dimension.City where [City Key]=7 --selectionner les 10 premiers lignes de la table Dimension City
--UPDATE Dimension.City SET City='Paris' where [State Province] = 'N/A'
--UPDATE Dimension.City SET Region='France'

```

City Key	WWI City ID	City	State Province	Country	Continent	Sales Territory	Region	Subregion	Location	Latest Recorded Population	Valid From
7	5456	Carrollton	Georgia	United States	North America	Southeast	Americas	Northern America	0xE6100000010C5049E70D41CA404042001533E74455C0	24388	2013-01-01 00:00:00

Affichage de résultat après transaction :

```

SQL TP BD.sql - LA...FG171K\bouba (60)*
use WideWorldImportersDW
UPDATE Dimension.City SET City='Paris' where [Sales Territory] = 'Southeast' -- modifier le nom de city paris quand sales territory
select * from Dimension.City where [City Key]=7 --selectionner ligne 7 de la table Dimension City
--UPDATE Dimension.City SET Region='France'

```

City Key	WWI City ID	City	State Province	Country	Continent	Sales Territory	Region	Subregion	Location	Latest Recorded Population	Valid From
7	5456	Paris	Georgia	United States	North America	Southeast	Americas	Northern America	0xE6100000010C5049E70D41CA404042001533E74455C0	24388	2013-01-01 00:00:00

Commit

Affichage de résultat après commit , on remarque qu'il y'a changement dans la base de données

```

Entrée [23]: import pyodbc
conn = pyodbc.connect('Driver={SQL Server};'
                    'Server=.;'
                    'Database=WideWorldImportersDW;'
                    'Trusted_Connection=yes;')

cursor = conn.cursor()
cursor.execute("select * from Dimension.City where [City Key]=7 ")

for row in cursor:
    print(row)
#conn.commit()
print("Database Updated !")

#cursor.execute("UPDATE Dimension.City SET City='Paris' where [Sales Territory] = 'Southeast'")
#print('\n')
#cursor.execute("select * from Dimension.City where [City Key]=7 ")
#for row in cursor:
#    # print(row)
#conn.close()

(7, 5456, 'Paris', 'Georgia', 'United States', 'North America', 'Southeast', 'Americas', 'Northern America', b'\xe6\x10\x00\x00\x01\x0cPI\xe7\rA\xca@B\x00\x153\xe7DU\xc0', 24388, '2013-01-01 00:00:00.000000', '2013-01-01 00:02:00.000000', 1)
Database Updated !

```

SQL TP BD.sql - LA...FG171K\bouba (60)*

```

use WideWorldImportersDW
--UPDATE Dimension.City SET City='Paris' where [Sales Territory] = 'Southeast'
-- modifier le nom de city paris quand sales territory
select * from Dimension.City where [City Key]=7 --selectionner ligne 7 de la table Dimension City

```

100 %

Résultats Résultats spatiaux Messages

	City Key	WWI City ID	City	State Province	Country	Continent	Sales Territory	Region	Subregion	Location	Latest Recorded Population	Valid From
1	7	5456	Paris	Georgia	United States	North America	Southeast	Americas	Northern America	0xE6100000010C5049E70D41CA404042001533E74455C0	24388	2013-01-01 00:00:00

Rollback

```

Entrée [27]: import pyodbc
conn = pyodbc.connect('Driver={SQL Server};'
                    'Server=.;'
                    'Database=WideWorldImportersDW;'
                    'Trusted_Connection=yes;')

cursor = conn.cursor()
cursor.execute("select * from Dimension.City where [City Key]=7 ")
cursor.execute("UPDATE Dimension.City SET City='nice' where [Sales Territory] = 'Southeast'")

cursor.execute("UPDATE Dimension.City SET City='nice' where [Sales Territory] = 'Southeast'")
conn.rollback()

print('\n')
cursor.execute("select * from Dimension.City where [City Key]=7 ")
for row in cursor:
    print(row)

#cursor.execute("UPDATE Dimension.City SET City='Paris' where [Sales Territory] = 'Southeast'")
#print('\n')
#cursor.execute("select * from Dimension.City where [City Key]=7 ")
#for row in cursor:
#    # print(row)
#conn.close()

(7, 5456, 'Paris', 'Georgia', 'United States', 'North America', 'Southeast', 'Americas', 'Northern America', b'\xe6\x10\x00\x00\x01\x0cPI\xe7\rA\xca@@B\x00\x153\xe7DU\xc0', 24388, '2013-01-01 00:00:00.000000', '2013-01-01 00:02:00.000000', 1)

```

Opération échoué le nom de city n’a pas changé

Commit

```

Entrée [28]: import pyodbc
conn = pyodbc.connect('Driver={SQL Server};'
                    'Server=.;'
                    'Database=WideWorldImportersDW;'
                    'Trusted_Connection=yes;')

cursor = conn.cursor()
cursor.execute("select * from Dimension.City where [City Key]=7 ")
cursor.execute("UPDATE Dimension.City SET City='nice' where [Sales Territory] = 'Southeast'")

cursor.execute("UPDATE Dimension.City SET City='nice' where [Sales Territory] = 'Southeast'")
conn.commit()

print('\n')
cursor.execute("select * from Dimension.City where [City Key]=7 ")
for row in cursor:
    print(row)

#cursor.execute("UPDATE Dimension.City SET City='Paris' where [Sales Territory] = 'Southeast'")
#print('\n')
#cursor.execute("select * from Dimension.City where [City Key]=7 ")
#for row in cursor:
#    # print(row)
#conn.close()

(7, 5456, 'nice', 'Georgia', 'United States', 'North America', 'Southeast', 'Americas', 'Northern America', b'\xe6\x10\x00\x00\x01\x0cPI\xe7\rA\xca@@B\x00\x153\xe7DU\xc0', 24388, '2013-01-01 00:00:00.000000', '2013-01-01 00:02:00.000000', 1)

```

Opération réussi, le nom de city a été changé pour Nice

Rollback

```
Entrée [35]: import pyodbc
conn = pyodbc.connect('Driver={SQL Server};'
                    'Server=.;'
                    'Database=WideWorldImportersDW;'
                    'Trusted_Connection=yes;')

cursor = conn.cursor()
cursor.execute("select * from Dimension.City where [City Key]=1 ")

cursor.execute("UPDATE Dimension.City SET City='tunis' where [City Key]=1")
#conn.rollback()

print('\n')
cursor.execute("select * from Dimension.City where [City Key]=1 ")
for row in cursor:
    print(row)

#cursor.execute("UPDATE Dimension.City SET City='Paris' where [Sales Territory] = 'Southeast'")
#print('\n')
#cursor.execute("select * from Dimension.City where [City Key]=7 ")
#for row in cursor:
#    print(row)
#conn.close()

(1, 5450, 'tunis', 'New York', 'United States', 'North America', 'Mideast', 'Americas', 'Northern America', b'\xe6\x10\x00\x00\x01\x0cp>\x1a\xf3\xdf\rE@\xdf\xfd\xfa\x1a\x05\xa9S\xc0', 0, '2013-01-01 00:00:00.000000', '2013-01-01 00:02:00.000000', 1)
```

Modification réussie

On utilise le Rollback

```
Entrée [36]: import pyodbc
conn = pyodbc.connect('Driver={SQL Server};'
                    'Server=.;'
                    'Database=WideWorldImportersDW;'
                    'Trusted_Connection=yes;')

cursor = conn.cursor()
cursor.execute("select * from Dimension.City where [City Key]=1 ")

cursor.execute("UPDATE Dimension.City SET City='tunis' where [City Key]=1")
conn.rollback()

print('\n')
cursor.execute("select * from Dimension.City where [City Key]=1 ")
for row in cursor:
    print(row)

#cursor.execute("UPDATE Dimension.City SET City='Paris' where [Sales Territory] = 'Southeast'")
#print('\n')
#cursor.execute("select * from Dimension.City where [City Key]=7 ")
#for row in cursor:
#    print(row)
#conn.close()

(1, 5450, 'Carrollton', 'New York', 'United States', 'North America', 'Mideast', 'Americas', 'Northern America', b'\xe6\x10\x00\x00\x01\x0cp>\x1a\xf3\xdf\rE@\xdf\xfd\xfa\x1a\x05\xa9S\xc0', 0, '2013-01-01 00:00:00.000000', '2013-01-01 00:02:00.000000', 1)
```

On remarque que Le rollback a effacé la transaction

SQL TP BD.sql - LA...FG171K\bouba (60))*

use WideWorldImportersDW

UPDATE Dimension.City SET City='tunis' where [City Key]=1

select * from Dimension.City

100 %

Résultats

Résultats spatiaux

Messages

	City Key	WWI City ID	City	State Province	Country	Continent	Sales Territory	Region	Subregion	Location	Latest Recorded Population	Valid
1	0	0	Unknown	N/A	N/A	N/A	N/A	N/A	N/A	NULL	0	20
2	1	5450	tunis	New York	United States	North America	Mideast	Americas	Northern America	0xE6100000010C703E1AF3DF0D4540DFFDF15EB5A953C0	0	20

Lorsqu'on a fait la transaction dans la base de donnée

```
Entrée [37]: import pyodbc
conn = pyodbc.connect('Driver={SQL Server};'
                      'Server=.;'
                      'Database=WideWorldImportersDW;'
                      'Trusted_Connection=yes;')

cursor = conn.cursor()
cursor.execute("select * from Dimension.City where [City Key]=1 ")

cursor.execute("UPDATE Dimension.City SET City='tunis' where [City Key]=1")
conn.rollback()

print('\n')
cursor.execute("select * from Dimension.City where [City Key]=1 ")
for row in cursor:
    print(row)

#cursor.execute("UPDATE Dimension.City SET City='Paris' where [Sales Territory] = 'Southeast'")
#print('\n')
#cursor.execute("select * from Dimension.City where [City Key]=7 ")
#for row in cursor:
#    print(row)
#conn.close()
```

(1, 5450, 'tunis', 'New York', 'United States', 'North America', 'Mideast', 'Americas', 'Northern America', b'\xe6\x10\x00\x00\x01\x0c\r\x1a\xfb\xdf\rE\xdf\xdf\xdf1^xb5xa9s\xc0', 0, '2013-01-01 00:00:00.0000000', '2013-01-01 00:02:00.0000000', 1)

On remarque que rollback a échoué d'annuler la transaction parce que cette dernière a été déjà enregistrée dans la base de données