

Transaction Processing

CO527 : Advanced Database Systems
Lab 04

Kanchana Jayasinghe
kc43224jayasinghe@eng.pdn.ac.lk

Outline

- What is a Transaction ?
- Properties of Transactions
- Transaction Control
- Transactional Control Commands
 - SET TRANSACTION
 - COMMIT
 - ROLLBACK
 - SAVEPOINT
 - RELEASE SAVEPOINT

What is a Transaction ?

- Transactions group a set of tasks into a single execution unit.
- Each transaction begins with a specific task and ends when all the tasks in the group successfully complete.
- A transaction is the propagation of one or more changes to the database.
- It is important to control these transactions to ensure the data integrity and to handle database errors.
- Incomplete steps result in the failure of the transaction.

Properties of Transactions

A database transaction, by definition, must be atomic, consistent, isolated and durable. These are popularly known as ACID properties.

- **Atomicity**
- **Consistency**
- **Isolation**
- **Durability**

Transaction Control

The following commands are used to control transactions.

- **SET TRANSACTION** – Places a name on a transaction.
- **COMMIT** – Saves all the transactions to the database since the last COMMIT or ROLLBACK command.
- **ROLLBACK** – Only be used to undo transactions since the last COMMIT or ROLLBACK command was issued (to roll back the changes).
- **SAVEPOINT** – creates points within the groups of transactions in which to ROLLBACK.

Transactional Control Commands

Transactional Control Commands

- These commands are only used with the **DML Commands** such as,
 - INSERT
 - UPDATE
 - DELETE only.
- They cannot be used,
 - **while creating tables** or
 - **dropping tables**

1. **SET TRANSACTION** command

- Can be used to initiate a database transaction.
- This command is used to specify characteristics for the transaction that follows.
For example,

you can specify a transaction to be read only or read write.

SYNTAX is as follows:

```
SET TRANSACTION [ READ WRITE | READ ONLY ];
```


2. **COMMIT** command

- Used to save changes invoked by a transaction to the database.
- Command saves all the transactions to the database since the last COMMIT or ROLLBACK command.

SYNTAX is as follows:

```
COMMIT;
```

Example :

Following is an example which would **delete those records from the table which have AGE = 45** and then **COMMIT the changes** in the database.

Consider the USERS table having the following records :

ID	NAME	AGE	ADDRESS	SALARY
1	Chamith	42	Kandy	25000.00
2	Kalana	45	Dehiwala	14500.00
3	Madhuja	35	Kothmale	32000.00
4	Charith	45	Mathara	26500.00
5	Harsha	47	Jaffna	38500.00
6	Keshani	23	Colombo	45000.00
7	Nuwan	31	Galle	10000.00

Command :

```
SQL> DELETE FROM USERS  
WHERE AGE = 45;
```

```
SQL> COMMIT;
```

Output :

Two rows from the table that have AGE=45 would be deleted and the SELECT statement would produce the following result:

```
SQL> SELECT * FROM USERS;
```

ID	NAME	AGE	ADDRESS	SALARY
1	Chamith	42	Kandy	25000.00
3	Madhuja	35	Kothmale	32000.00
5	Harsha	47	Jaffna	38500.00
6	Keshani	23	Colombo	45000.00
7	Nuwan	31	Galle	10000.00

3. **ROLLBACK** command

- Used to undo transactions that have not already been saved to the database.
- This command can only be used to undo transactions since the last COMMIT or ROLLBACK command was issued.

SYNTAX is as follows:

```
ROLLBACK;
```

Example :

Following is an example, which would **delete those records from the table which have the age = 45** and then **ROLLBACK** the changes in the database.

Consider the USERS table having the following records :

ID	NAME	AGE	ADDRESS	SALARY
1	Chamith	42	Kandy	25000.00
2	Kalana	45	Dehiwala	14500.00
3	Madhuj	35	Kothmale	32000.00
4	Charith	45	Mathara	26500.00
5	Harsha	47	Jaffna	38500.00
6	Keshani	23	Colombo	45000.00
7	Nuwan	31	Galle	10000.00

Command :

```
SQL> DELETE FROM USERS  
WHERE AGE = 45;
```

```
SQL> ROLLBACK;
```

Output :

The **delete operation would not impact** the table and the SELECT statement would produce the following result:

```
SQL> SELECT * FROM USERS;
```

ID	NAME	AGE	ADDRESS	SALARY
1	Chamith	42	Kandy	25000.00
2	Kalana	45	Dehiwala	14500.00
3	Madhuj	35	Kothmale	32000.00
4	Charith	45	Mathara	26500.00
5	Harsha	47	Jaffna	38500.00
6	Keshani	23	Colombo	45000.00
7	Nuwan	31	Galle	10000.00

4. **SAVEPOINT** command

- Creates points within the groups of transactions in which to ROLLBACK.
- A SAVEPOINT is a point in a transaction in which you can roll the transaction back to a certain point without rolling back the entire transaction.

SYNTAX is as follows:

```
SAVEPOINT [SAVEPOINT_NAME];
```

This command is used only in the creation of SAVEPOINT among all the transactions.

4. **SAVEPOINT** command cont.

- In general ROLLBACK is used to undo a group of transactions.

Syntax for rolling back to Savepoint command:

```
ROLLBACK TO [SAVEPOINT_NAME];
```

You can ROLLBACK to any SAVEPOINT at any time to return the appropriate data to its original state.

Example :

You plan to **delete the first three records from the USERS table**. You want to **create a SAVEPOINT before each delete**, so that **you can ROLLBACK to any SAVEPOINT at any time** to return the appropriate data to its original state.

ID	NAME	AGE	ADDRESS	SALARY
1	Chamith	42	Kandy	25000.00
2	Kalana	45	Dehiwala	14500.00
3	Madhuja	35	Kothmale	32000.00
4	Charith	45	Mathara	26500.00
5	Harsha	47	Jaffna	38500.00
6	Keshani	23	Colombo	45000.00
7	Nuwan	31	Galle	10000.00

```
SQL> SAVEPOINT SV_P1;
```

Savepoint created.

```
SQL> DELETE FROM USERS  
WHERE ID=1;
```

1 row deleted.

```
SQL> SAVEPOINT SV_P2;
```

Savepoint created.

```
SQL> DELETE FROM USERS  
WHERE ID=2;
```

1 row deleted.

```
SQL> SAVEPOINT SV_P3;
```

Savepoint created.

```
SQL> DELETE FROM USERS  
WHERE ID=3;
```

1 row deleted.



**Code block for the
series of DELETION
and SAVEPOINT
operations**

Output :

After the all 3 deletion took place, SELECT statement would produce the following result:

```
SQL> SELECT * FROM USERS;
```

ID	NAME	AGE	ADDRESS	SALARY
4	Charith	45	Mathara	26500.00
5	Harsha	47	Jaffna	38500.00
6	Keshani	23	Colombo	45000.00
7	Nuwan	31	Galle	10000.00

Assume that you have changed your mind and need to undo last two deletions:

```
SQL> ROLLBACK TO SV_P2;  
Rollback complete.
```

After ROLLBACK to SV_P2, SELECT statement would produce the following result:

SQL> **SELECT * FROM USERS;**

ID	NAME	AGE	ADDRESS	SALARY
2	Kalana	45	Dehiwala	14500.00
3	Madhuja	35	Kothmale	32000.00
4	Charith	45	Mathara	26500.00
5	Harsha	47	Jaffna	38500.00
6	Keshani	23	Colombo	45000.00
7	Nuwan	31	Galle	10000.00

- Please note that **only the first deletion took place since you rolled back to SV_P2**. And the other two deletions are undone.

5. **RELEASE SAVEPOINT** command

- Is used to remove a SAVEPOINT that you have created.

SYNTAX is as follows:

```
RELEASE SAVEPOINT [SAVEPOINT_NAME];
```

Summary

- What is a Transaction ?
- Properties of Transactions
- Transaction Control
- Transactional Control Commands

