



ALL ABOUT BEGIN , COMMIT & ROLLBACK

--Begins a transaction in the current session.

-- START TRANSACTION is a synonym for BEGIN.

/* Parameters

WORK | TRANSACTION

Optional keywords that provide compatibility with other database systems.

NAME name

Optional string that assigns a name to the transaction. A name helps identify a transaction, but is not required and does not need to be unique.

*/

BEGIN;

SHOW TRANSACTIONS;

-- 1698438076456000000,ANANLEARNINGACADEMY ,6,554,558,517,5d0b6c5d-2fb1-49b9-9c8a-e92017b4d9ea,2023-10-27 13:21:16.456 -0700 ,running ,0

SELECT CURRENT_TRANSACTION();

-- 1698438076456000000

--Note the system-assigned name, 5d0b6c5d-2fb1-49b9-9c8a-e92017b4d9ea, for the transaction.

--Describes the transaction, including the start time and the state (running, committed, rolled back).

-- DESCRIBE can be abbreviated to DESC.

DESC TRANSACTION 1698438076456000000;

--Begin a transaction with a specified name:

BEGIN NAME T1;

SHOW TRANSACTIONS;



ALL ABOUT BEGIN , COMMIT & ROLLBACK

```
SELECT CURRENT_TRANSACTION();
```

--Same as previous example, but using START TRANSACTION instead of BEGIN:

```
START TRANSACTION NAME T2;
```

```
SHOW TRANSACTIONS;
```

```
--COMMIT
```

-- Commits an open transaction in the current session.

-- If two COMMIT statements in a row are executed (within the same scope), the second one is ignored.

-- For example, in the following code, the second COMMIT has no effect; there is no open transaction to commit.

```
/* begin;  
insert into table1 ...;  
commit;  
commit; -- Ignored! */
```

```
CREATE or replace TABLE cows_one (cnumber int, cbreed varchar(20));
```

```
SELECT COUNT(*) FROM cows_one;
```

```
BEGIN NAME T3;
```

```
SELECT CURRENT_TRANSACTION(); -- 1698472387696000000
```

```
INSERT INTO cows_one VALUES (1,'Holstein');
```

```
INSERT INTO cows_one VALUES (2,'Guernsey');
```

```
INSERT INTO cows_one VALUES (3,'Angus');
```



ALL ABOUT BEGIN , COMMIT & ROLLBACK

```
SELECT COUNT(*) FROM cows_one;  
  
commit;
```

```
SELECT CURRENT_TRANSACTION(); --null
```

```
SELECT LAST_TRANSACTION(); --1698438848575000000
```

```
SELECT COUNT(*) FROM cows_one;
```

```
--ROLLBACK
```

--Rolls back an open transaction in the current session.

-- Begin a transaction, insert some values into a table, and then complete the transaction by rolling back the changes made in the transaction:

```
CREATE OR REPLACE TABLE cows_two (cnumber int, breeds varchar(20));
```

```
SELECT COUNT(*) FROM cows_two; --0
```

```
BEGIN NAME T5;
```

```
SELECT CURRENT_TRANSACTION(); --1698472572403000000
```

```
INSERT INTO cows_two VALUES (2,'Jersey');
```

```
INSERT INTO cows_two VALUES (3,'Brown Swiss');
```

```
INSERT INTO cows_two VALUES (4,'Ayrshire');
```

```
SELECT COUNT(*) FROM cows_two; --3
```

```
ROLLBACK;
```



ALL ABOUT BEGIN , COMMIT & ROLLBACK

```
SELECT COUNT(*) FROM cows_two; --0
```

```
SELECT CURRENT_TRANSACTION();--null
```

```
SELECT LAST_TRANSACTION(); --1698439496443000000
```

```
select * from DEMO_DATABASE.DEMO_SCHEMA."cows_one";
```

ANALYTICSWITHANAND