|  |
| --- |
|  |
|  |

**Azin Rezaeian- Lab 8**

1. Start two connections to your database server (note that both connections can be made with the same user i.e. root). On the first instance/connection, execute the following. Red text will indicate the first instance; blue text will indicate the second.
2. Write a query that calculates the number of records in the example1 table and provide this query. (1 mark)

SELECT COUNT (\*) FROM example1;

1. Execute the query from 1. How many records are stored in the example1 table? (1 mark)

100 records.

Graphical user interface, text, application, email

Description automatically generated

1. Execute the query below.
2. Execute the insert statements from 1 again. How many records are stored in the example1 table?

Graphical user interface, text, application

Description automatically generated

1. Switch to the second instance/connection. Set the default database to transtest.
2. How many records are stored in the example1 table? Why? Which letter of ACID does it demonstrate? (1 mark) 100 records.

When one instance starts a transaction, the changes made during that transaction are not visible to other instances until the transaction is committed. This is because each instance has its own view of the database (the Isolation property) and does not see changes made by other transactions until they are committed.

Graphical user interface, text, application, email

Description automatically generated

1. Switch to the first instance/connection and ensure safe updates is disabled.
2. Run the following command:
3. How many records are stored in the example1 table in the first instance? (1 mark)

Graphical user interface, application

Description automatically generated

1. Commit the transaction from the first instance then switch to the second instance/connection.
2. How many records are stored in the example1 table in the second instance? Explain. (1 mark)

Once the first instance committed the transaction that inserted into the example1 table, those changes were made permanent and were visible to all subsequent instances, including the second instance.

Graphical user interface, text, application

Description automatically generated

1. Switch back to the first instance/connection and run the following:
2. How many records are stored in the example1 table in the first instance? (1 mark)

Graphical user interface, application

Description automatically generated

1. Switch to the second instance/connection.
2. What is the number of records in the example1 table in the second instance? (1 mark)

Graphical user interface, text, application

Description automatically generated

1. Which letter of ACID does it demonstrate? (1 mark)

it's the Letter A "Atomicity". Even though the first instance was unable to commit its transaction due to an error, the changes made by that transaction were not visible to the second instance until the transaction was successfully committed. This means that if any part of a transaction fails, the entire transaction is rolled back, and the database is returned to its state prior to the start of the transaction.

1. Switch back to the first instance and run.
2. How many records are stored in the example1 table in the first instance? (1 mark)

Graphical user interface, application

Description automatically generated

1. Switch to the second instance/connection.
2. How many records are stored in the example1 table in the second instance? Explain. (1 mark)

After running commit on the first instance, all changes made by the transaction are now permanent and visible to other transactions. Therefore, the number of records in the example1 table in the first instance and the second instance should now be the same.

Graphical user interface, text

Description automatically generated

1. Switch to the first instance/connection. Execute the entire transaction below:
2. How many records are stored in the example1 table in the first instance? (1 mark)

Graphical user interface, application

Description automatically generated

1. Switch to the second instance/connection.
2. How many records are stored in the example1 table in the second instance? Explain. (1 mark)

94 records are stored in the example1 table in the second instance. This is because the entire transaction was rolled back in the first instance before it was committed, so the changes were not actually made to the database, and nothing changed in the database and both instances maintain the same number of records from before.

Graphical user interface, application

Description automatically generated

1. Switch to the first instance/connection and run:
2. How many records are stored in the example1 table in the first instance? (1 mark)

Graphical user interface, application

Description automatically generated

1. Switch to the second instance/connection.
2. How many records are stored in the example1 table in the second instance? (1 mark)

The second instance was not affected by the rollback to the save point in the first instance, as the changes had not been committed yet. Therefore, the second instance still shows 94 records.

Table

Description automatically generated with medium confidence

1. Switch to the first instance/connection and runt he is following:
2. How many records are stored in the example1 table in the first instance? (1 mark)

Graphical user interface, application

Description automatically generated

1. Switch to the second instance/connection.
2. How many records are stored in the example1 table in the second instance? Explain (1 mark)

The second instance can now see the changes made by the committed transaction in the first instance, resulting in the same number of records in both instances. Because of “rollback to savepoint i15;” in first instance shows 95 records, and then after “commit” the second instance get changed from 94 to 95.

Graphical user interface, application

Description automatically generated