Lab 06 – Transactions and Security

Objectives:

The purpose of this lab is to introduce the student to both transactions and security. In the real-world, databases tasks often involve multiple steps and if any step in the middle fails, the procedure is a failure. This lap walks the student through a couple transactions and lets them learn how various steps have varying consequences that they need to be aware of.

By the end of this lab, the student will be able to:

- Describe the steps of a transaction, how a transaction begins and ends and walk through live scenarios of a variety of transactions
- Understand and act appropriately on what needs to be done in the case of transaction failure
- Grant and revoke permissions to and from other users and public users from the database

Submission:

Your submission will be a single WORD file with the query and result screenshot from Oracle SQL developer

Make sure every SQL statement terminates with a semicolon.

- You will use following data to complete the given tasks:
- ***SET TRANSACTION READ WRITE*** starts a new transaction.

- ***COMMIT*** commits the current transaction, making its changes permanent.
- ***SAVEPOINT <name>*** sets a pointer to a location that can be rolled back to.
- ***ROLLBACK*** rolls back the current transaction, canceling its changes.
- ***SET autocommit*** disables or enables the default ***autocommit*** mode for the current session.

Tasks:

It is very important that these tasks/questions be performed in the order presented here for maximum learning.

PART A - Transactions

1. Execute the following commands.

```
SET AUTCOMMIT OFF;
SET TRANSACTION READ WRITE;
```

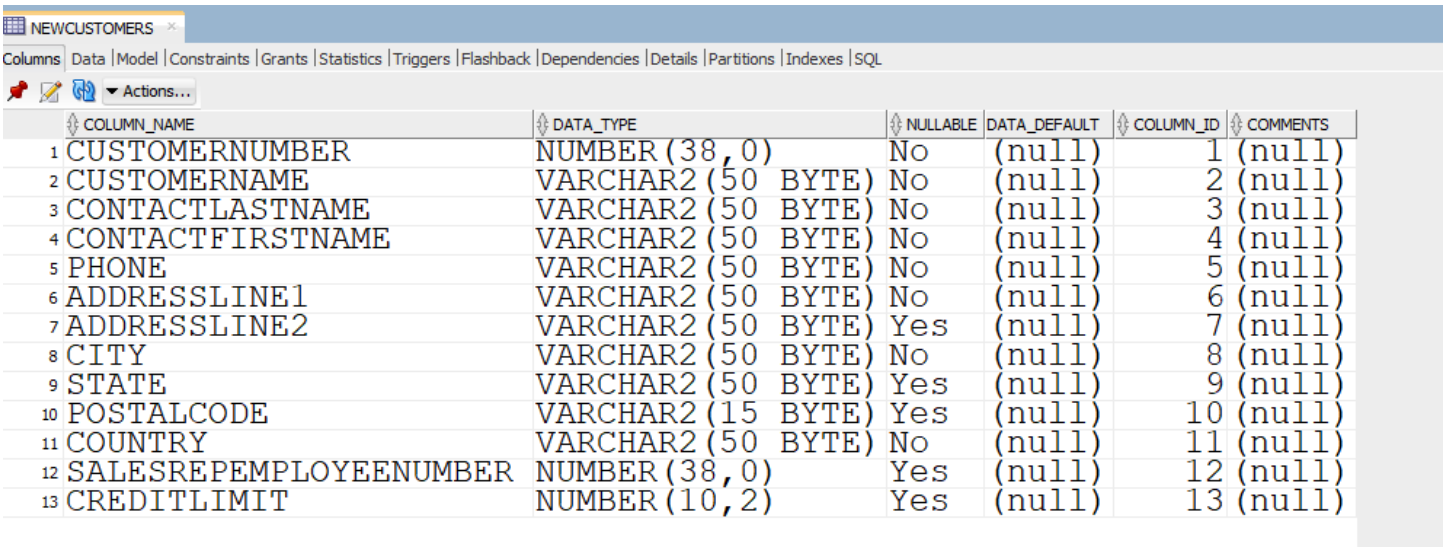
Using SQL, create an **empty** table, that is the same as the RETAILCUSTOMERS table, and name it

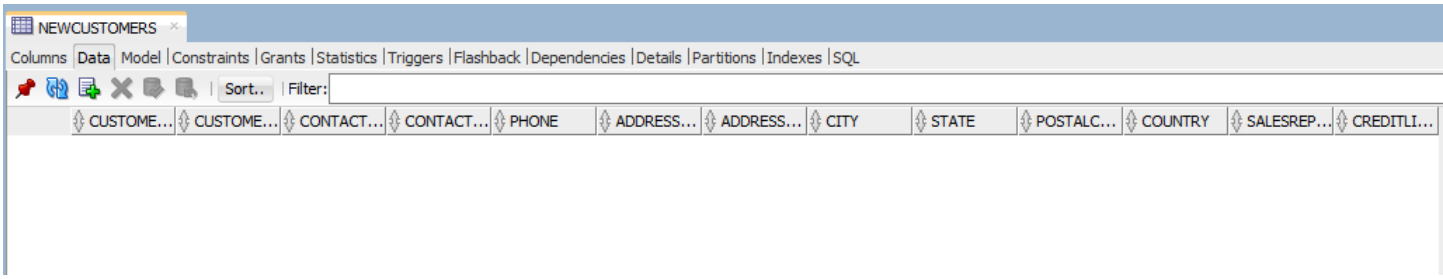
newCustomers.

CREATE TABLE NEWCUSTOMERS AS (SELECT * FROM RETAILCUSTOMERS WHERE 0=1) ;

You just need to add a where clause which will never set to be true or always false for example 0=1 OR 1=2

Table NEWCUSTOMERS created.





no rows selected

2. Write an INSERT statement to populate the **newCustomers** table with the rows of the sample data. (Write a single INSERT statement to insert all the rows, combine firstname and last name to get full name during insert)

customerNumber	contactLastName	contactFirstName	Phone	addressLine1	city	country
100	Patel	Ralph	2233355555	10 SenecaWay	Paris	France
101	Denis	Betty	3344455555	110 SenecaWay	Chicago	USA
102	Biri	Ben	44555445544	13000 SenecaWay	Toronto	Canada
103	Newman	Chad	66777332233	12 SenecaWay	Mexico city	Mexico
104	Ropeburn	Audrey	7788811212	15000 SenecaWay	Havana	Cuba
105	Lucy	Preston	45555511111	12 SenecaWay	Charlotte	USA

```
INSERT ALL
INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE,
ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)
VALUES (100, 'Patel', 'Ralph', 2233355555, '10 Seneca Way', 'Paris', 'France', 'Ralph Patel')
INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE,
ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)
VALUES (101, 'Denis', 'Betty', 3344455555, '110 Seneca Way', 'Chicago', 'USA', 'Betty Denis')
INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE,
ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)
VALUES (102, 'Biri', 'Ben', 444555445544, '13000 Seneca Way', 'Toronto', 'Canada', 'Ben Biri')
INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE,
ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)
VALUES (103, 'Newman', 'Chad', 66777332233, '12 Seneca Way', 'Mexico City', 'Mexico', 'Chad
Newman')
INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE,
ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)
VALUES (104, 'Ropeburn', 'Audrey', 7788811212, '15000 Seneca Way', 'Havana', 'Cuba', 'Audrey
Ropeburn')
INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE,
ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)
VALUES (105, 'Lucy', 'Preston', 45555511111, '12 Seneca Way', 'Charlotte', 'USA', 'Preston Lucy')

SELECT 1 FROM DUAL;
```

Columns | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL

Sort.. | Filter:

	CUSTOMERNUMBER	CUSTOMERNAME	CONTACTLASTNAME	CONTACTFIRSTNAME	PHONE	ADDRESSLINE1	ADDRESSLINE2	CITY	STATE
1	100	Ralph Patel	Patel	Ralph	2233355555	10 Seneca Way	(null)	Paris	(null)
2	101	Betty Denis	Denis	Betty	3344455555	110 Seneca Way	(null)	Chicago	(null)
3	102	Ben Biri	Biri	Ben	444555445544	13000 Seneca Way	(null)	Toronto	(null)
4	103	Chad Newman	Newman	Chad	66777332233	12 Seneca Way	(null)	Mexico City	(null)
5	104	Audrey Ropeburn	Ropeburn	Audrey	7788811212	15000 Seneca Way	(null)	Havana	(null)
6	105	Preston Lucy	Lucy	Preston	45555511111	12 Seneca Way	(null)	Charlotte	(null)

3. Create a query that shows all the inserted rows from the newCustomers table. How many rows are selected?

```
6 rows are selected
SELECT * FROM NEWCUSTOMERS
WHERE CUSTOMERNUMBER IN (100,101,102,103,104,105);
```

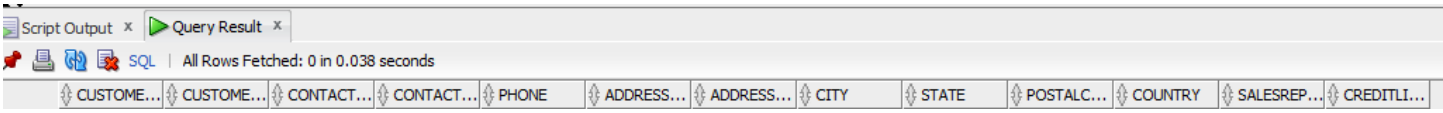
Script Output | Query Result

SQL | All Rows Fetched: 6 in 0.031 seconds

	CUSTOMERNUMBER	CUSTOMERNAME	CONTACTLASTNAME	CONTACTFIRSTNAME	PHONE	ADDRESSLINE1
1	100	Ralph Patel	Patel	Ralph	2233355555	10 Seneca Way
2	101	Betty Denis	Denis	Betty	3344455555	110 Seneca Way
3	102	Ben Biri	Biri	Ben	444555445544	13000 Seneca Way
4	103	Chad Newman	Newman	Chad	66777332233	12 Seneca Way
5	104	Audrey Ropeburn	Ropeburn	Audrey	7788811212	15000 Seneca Way
6	105	Preston Lucy	Lucy	Preston	45555511111	12 Seneca Way

4. Execute the rollback command. Display all rows and columns from the newCustomers table. How many rows are selected?

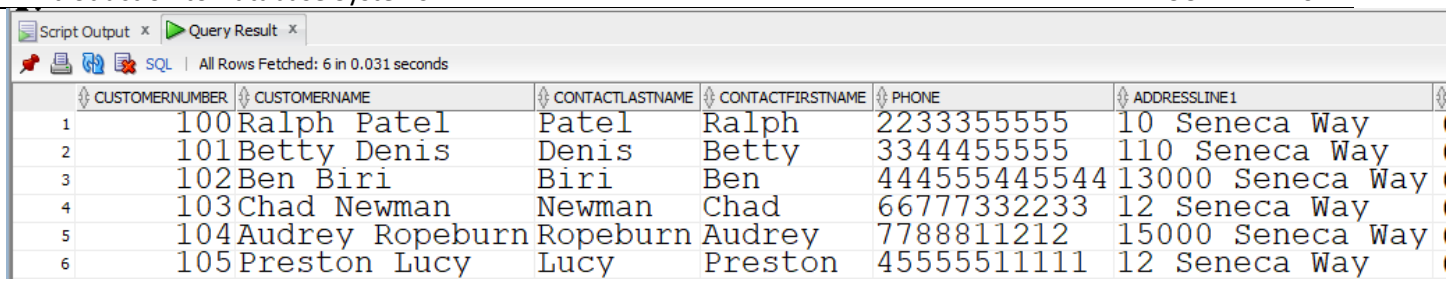
ROLLBACK;
Rollback complete.
SELECT * FROM NEWCUSTOMERS;



5. Repeat Question 2. Make the insertion permanent to the table newCustomers. Display all rows and columns from the newCustomers table. How many rows are selected?

INSERT ALL
INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE, ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)
VALUES (100, 'Patel', 'Ralph', 2233355555, '10 Seneca Way', 'Paris', 'France', 'Ralph Patel')
INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE, ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)
VALUES (101, 'Denis', 'Betty', 3344455555, '110 Seneca Way', 'Chicago', 'USA', 'Betty Denis')
INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE, ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)
VALUES (102, 'Biri', 'Ben', 444555445544, '13000 Seneca Way', 'Toronto', 'Canada', 'Ben Biri')
INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE, ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)
VALUES (103, 'Newman', 'Chad', 66777332233, '12 Seneca Way', 'Mexico City', 'Mexico', 'Chad Newman')
INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE, ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)
VALUES (104, 'Ropeburn', 'Audrey', 7788811212, '15000 Seneca Way', 'Havana', 'Cuba', 'Audrey Ropeburn')
INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE, ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)
VALUES (105, 'Lucy', 'Preston', 45555511111, '12 Seneca Way', 'Charlotte', 'USA', 'Preston Lucy')

SELECT 1 FROM DUAL;
6 rows are selected
SELECT * FROM NEWCUSTOMERS


Script Output | Query Result

SQL | All Rows Fetched: 6 in 0.031 seconds

	CUSTOMERNUMBER	CUSTOMERNAME	CONTACTLASTNAME	CONTACTFIRSTNAME	PHONE	ADDRESSLINE1
1	100	Ralph Patel	Patel	Ralph	2233355555	10 Seneca Way
2	101	Betty Denis	Denis	Betty	3344455555	110 Seneca Way
3	102	Ben Biri	Biri	Ben	444555445544	13000 Seneca Way
4	103	Chad Newman	Newman	Chad	66777332233	12 Seneca Way
5	104	Audrey Ropeburn	Ropeburn	Audrey	7788811212	15000 Seneca Way
6	105	Preston Lucy	Lucy	Preston	45555511111	12 Seneca Way

6. Write an update statement to update the value of column addressLine1 to ‘unknown’ for all the customers in the newCustomers table.

UPDATE NEWCUSTOMERS

SET ADDRESSLINE1 = 'unknown'

6 rows updated.

7. Make your changes permanent.

COMMIT;
Commit complete.

8. Execute the rollback command.

ROLLBACK;

Rollback complete.

a. Display all customers from the newCustomers table whose address is ‘unknown’. How many rows are still updated?

SELECT * FROM NEWCUSTOMERS WHERE ADDRESSLINE1 = 'unknown';

All were selected.

b. Was the rollback command effective?

The rollback command was not effective.

c. What was the difference between the result of the rollback execution from Question 6 and the result of the rollback execution of this task?

The difference was that the rollback removed all of the data inserted before, and not it did not change any of the data that was committed.

9. Begin a new transaction and then create a statement to delete the customers from the newCustomers table

COMMIT;

DELETE FROM NEWCUSTOMERS;

10. Perform a rollback to undo the deletion of the customers

    a. How many customers are now in the newCustomers table?

    SELECT * FROM NEWCUSTOMERS

    6 ROWS WERE SELECTED

    b. Was the rollback effective and why?

    Yes. The rollback was effective because it was not committed after delete operation

11. Begin a new transaction and rerun the data insertion from Question 2 (copy the code down to Question 11 and run it)

```
BEGIN

INSERT ALL

INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE, ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)

VALUES (100, 'Patel', 'Ralph', 2233355555, '10 Seneca Way', 'Paris', 'France', 'Ralph Patel')

INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE, ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)

VALUES (101, 'Denis', 'Betty', 3344455555, '110 Seneca Way', 'Chicago', 'USA', 'Betty Denis')

INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE, ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)

VALUES (102, 'Biri', 'Ben', 444555445544, '13000 Seneca Way', 'Toronto', 'Canada', 'Ben Biri')

INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE, ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)

VALUES (103, 'Newman', 'Chad', 66777332233, '12 Seneca Way', 'Mexico City', 'Mexico', 'Chad Newman')

INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE, ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)

VALUES (104, 'Ropeburn', 'Audrey', 7788811212, '15000 Seneca Way', 'Havana', 'Cuba', 'Audrey Ropeburn')

INTO NEWCUSTOMERS(CUSTOMERNUMBER, CONTACTLASTNAME, CONTACTFIRSTNAME, PHONE, ADDRESSLINE1, CITY, COUNTRY, CUSTOMERNAME)

VALUES (105, 'Lucy', 'Preston', 45555511111, '12 Seneca Way', 'Charlotte', 'USA', 'Preston Lucy')
```

SELECT 1 FROM DUAL;

SQL | All Rows Fetched: 12 in 0.025 seconds

	CUSTOMERNUMBER	CUSTOMERNAME	CONTACTLASTNAME	CONTACTFIRSTNAME	PHONE	ADDRESSLI
1	100	Ralph Patel	Patel	Ralph	2233355555	10 Se
2	101	Betty Denis	Denis	Betty	3344455555	110 S
3	102	Ben Biri	Biri	Ben	444555445544	13000
4	103	Chad Newman	Newman	Chad	66777332233	12 Se
5	104	Audrey Ropeburn	Ropeburn	Audrey	7788811212	15000
6	105	Preston Lucy	Lucy	Preston	45555511111	12 Se
7	100	Ralph Patel	Patel	Ralph	2233355555	10 Se
8	101	Betty Denis	Denis	Betty	3344455555	110 S
9	102	Ben Biri	Biri	Ben	444555445544	13000
10	103	Chad Newman	Newman	Chad	66777332233	12 Se
11	104	Audrey Ropeburn	Ropeburn	Audrey	7788811212	15000
12	105	Preston Lucy	Lucy	Preston	45555511111	12 Se

12. Set a Savepoint, called ***insertion***, after inserting the data

--12 ROWS WAS INSERTED

SAVEPOINT INSERTION;

13. Rerun the update statement from Question 6 and run a query to view the data (copy the code down and run it again)

UPDATE NEWCUSTOMERS

SET ADDRESSLINE1 = 'unknown'

--12 ROWS UPDATED

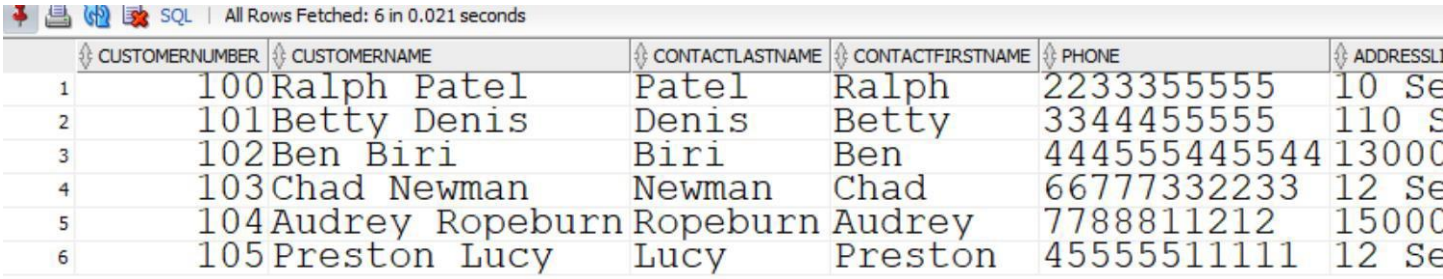
SQL | All Rows Fetched: 12 in 0.029 seconds

	CUSTOMERNAME	CONTACTLASTNAME	CONTACTFIRSTNAME	PHONE	ADDRESSLINE1
1	Ralph Patel	Patel	Ralph	2233355555	unknown
2	Betty Denis	Denis	Betty	3344455555	unknown
3	Ben Biri	Biri	Ben	444555445544	unknown
4	Chad Newman	Newman	Chad	66777332233	unknown
5	Audrey Ropeburn	Ropeburn	Audrey	7788811212	unknown
6	Preston Lucy	Lucy	Preston	45555511111	unknown
7	Ralph Patel	Patel	Ralph	2233355555	unknown
8	Betty Denis	Denis	Betty	3344455555	unknown
9	Ben Biri	Biri	Ben	444555445544	unknown

14. Rollback the transaction to the Savepoint created in Question 12 above and run a query to view the data.
What does the data look like (i.e. describe what happened?

ROLLBACK INSERTION;

select * from newcustomers



SQL | All Rows Fetched: 6 in 0.021 seconds

	CUSTOMERNUMBER	CUSTOMERNAME	CONTACTLASTNAME	CONTACTFIRSTNAME	PHONE	ADDRESSLI
1	100	Ralph Patel	Patel	Ralph	2233355555	10 Se
2	101	Betty Denis	Denis	Betty	3344455555	110 S
3	102	Ben Biri	Biri	Ben	444555445544	13000
4	103	Chad Newman	Newman	Chad	66777332233	12 Se
5	104	Audrey Ropeburn	Ropeburn	Audrey	7788811212	15000
6	105	Preston Lucy	Lucy	Preston	45555511111	12 Se

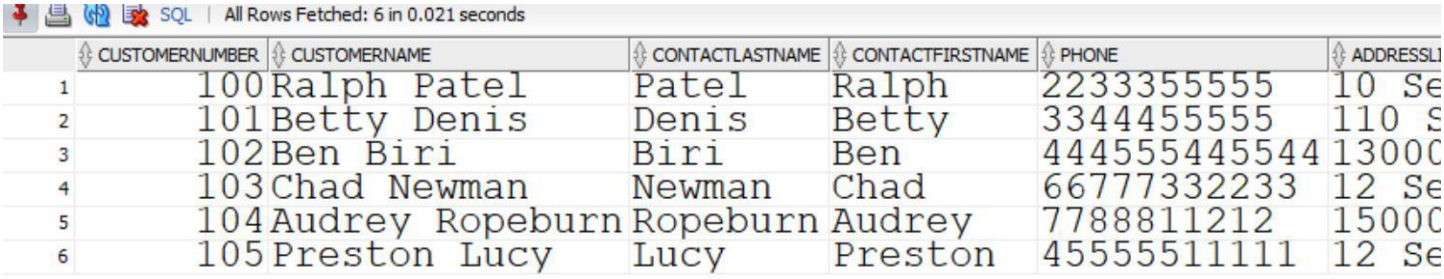
The update was undone and the new data was not inserted.

15. Use the rollback statement and again view the data. Describe what the results look like and what happened.

ROLLBACK INSERTION2;

select * from newcustomers

same result. Nothing happened



SQL | All Rows Fetched: 6 in 0.021 seconds

	CUSTOMERNUMBER	CUSTOMERNAME	CONTACTLASTNAME	CONTACTFIRSTNAME	PHONE	ADDRESSLI
1	100	Ralph Patel	Patel	Ralph	2233355555	10 Se
2	101	Betty Denis	Denis	Betty	3344455555	110 S
3	102	Ben Biri	Biri	Ben	444555445544	13000
4	103	Chad Newman	Newman	Chad	66777332233	12 Se
5	104	Audrey Ropeburn	Ropeburn	Audrey	7788811212	15000
6	105	Preston Lucy	Lucy	Preston	45555511111	12 Se

Part B - Permissions

16. Write a statement that denies all access to the newCustomers table for all public users

REVOKE ALL ON newCustomers FROM PUBLIC;

17. Write a statement that allows a person ‘RGNANAOLIVU’ read only access to the newCustomers table.

GRANT SELECT ON newCustomers TO RGNANAOLIVU

18. Write a statement that allows the same person ‘RGNANAOLIVU’ to modify (insert, update and delete) the data of the newCustomers table.

GRANT INSERT,UPDATE,DELETE ON newCustomers TO RGNANAOLIVU

19. Write a statement the denies all access to the newCustomers table for the same person 'RGNANAOLIVU'.

REVOKE ALL TO newCustomers FROM RGNANAOLIVU

Part C – Clean up

20. Write statements to permanently remove the table created for this lab

DROP TABLE newCustomers