Experiment-3

Student Name: Gauri Prabhakar
UID: 18BCS6201
Branch: 18AITAIML-2
Section/Group: B
Semester: 7
Date of Performance: 1st September, 2021
Subject Name: Advanced Database Management Lab
Subject Code: CSP - 434

1. Aim/Overview of the practical:

To analyze and create different types of locks.

2. Task to be done:

To analyze and create different types of locks.

3. Steps to be followed:

Setting transaction isolation level for previously curated table- STUDENTS and then Inserting, Updating the tuples then finally COMMITTING the values :

1. SELECT * FROM STUDENTS;

```
Connected.
SQL> SELECT * FROM STUDENTS;

   ROLL_NO NAME                                               AGE
---------- ---------------------------------------------------------- ----------
         1 Gauri                                              21
         2 Lilly                                              22
         3 Rose                                               18
```

2. SET TRANSACTION ISOLATION LEVEL READ COMMITTED;
INSERT INTO STUDENTS VALUES(4,'Chelsea',24);
UPDATE STUDENTS SET Age=40 where Roll_No=1;
COMMIT;

```
SQL> SET TRANSACTION ISOLATION LEVEL READ COMMITTED;

Transaction set.

SQL> INSERT INTO STUDENTS VALUES(4,'Chelsea',24);

1 row created.

SQL> UPDATE STUDENTS SET Age=40 where Roll_No=1;

1 row updated.

SQL> COMMIT;
```

Displaying and then LOCKING the table in SHARE MODE:

3. SELECT * FROM STUDENTS;
LOCK TABLE STUDENTS IN SHARE MODE;

```
SQL> SELECT * FROM STUDENTS;

   ROLL_NO NAME                                               AGE
---------- ---------------------------------------------------- ----------
         1 Gauri                                                40
         2 Lilly                                                22
         3 Rose                                                 18
         4 Chelsea                                              24

SQL> LOCK TABLE STUDENTS IN SHARE MODE;

Table(s) Locked.
```

Again Inserting and updating tuples and finally COMMITTING them to the table and displaying the table:

4. INSERT INTO STUDENTS VALUES(5,'Ross',30);
UPDATE STUDENTS SET Age=20 where Roll_No=5;
COMMIT;
SELECT * FROM STUDENTS;

```
SQL> INSERT INTO STUDENTS VALUES(5,'Ross',30);

1 row created.

SQL> UPDATE STUDENTS SET Age=20 where Roll_No=5;

1 row updated.

SQL> COMMIT;

Commit complete.

SQL> SELECT * FROM STUDENTS;

   ROLL_NO NAME                                               AGE
---------- ---------------------------------------------------- ----------
         1 Gauri                                                40
         2 Lilly                                                22
         3 Rose                                                 18
         4 Chelsea                                              24
         5 Ross                                                 20
```

LOCKING the table in EXCLUSIVE MODE then inserting and updating tuples and finally COMMITTING them to the table and displaying the table:

5. LOCK TABLE STUDENTS IN EXCLUSIVE MODE;
INSERT INTO STUDENTS VALUES(6,'Joey',25);
UPDATE STUDENTS SET Age=50 where Roll_No=2;
COMMIT;
SELECT * FROM STUDENTS;

```
SQL> LOCK TABLE STUDENTS IN EXCLUSIVE MODE;

Table(s) Locked.

SQL> INSERT INTO STUDENTS VALUES(6,'Joey',25);

1 row created.

SQL> UPDATE STUDENTS SET Age=50 where Roll_No=2;

1 row updated.

SQL> COMMIT;

Commit complete.
```

```
SQL> SELECT * FROM STUDENTS;

   ROLL_NO NAME                                               AGE
---------- ---------------------------------------------------- ----------
         1 Gauri                                                40
         2 Lilly                                                50
         3 Rose                                                 18
         4 Chelsea                                              24
         5 Ross                                                 20
         6 Joey                                                 25

6 rows selected.
```

6. LOCKING the table in SHARE MODE , updating a tuple, ROLLBACKING and displaying the table:

```
LOCK TABLE STUDENTS IN SHARE MODE;
UPDATE STUDENTS SET Age=20 where Roll_No=6;
SELECT * FROM STUDENTS;
```

```
SQL> LOCK TABLE STUDENTS IN SHARE MODE;

Table(s) Locked.

SQL> UPDATE STUDENTS SET Age=20 where Roll_No=6;

1 row updated.

SQL> ROLLBACK;

Rollback complete.

SQL> SELECT * FROM STUDENTS;

   ROLL_NO NAME                                               AGE
---------- -------------------------------------------------- ----------
         1 Gauri                                              40
         2 Lilly                                              50
         3 Rose                                               18
         4 Chelsea                                            24
         5 Ross                                               20
         6 Joey                                               25

6 rows selected.
```

4. Result/Output/Writing Summary:

- Successfully implemented Locks.
- Successfully understood the functioning and importance of Locks.
- Successfully implemented TCL Commands: Commit and Rollback.
- Successfully understood the working of Transaction Isolation Level, Locks and TCL commands.

5. Learning outcomes (What I have learnt):

- How to implement locks on SQL Command Line.
- TCL Commands.
- How to implement Commit and Rollback command and returned their outcomes.
- How to set Transaction Isolation Level and Locks on a table.

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			