K. J. Somaiya College of Engineering, Mumbai-77
(Autonomous College Affiliated to University of Mumbai)

Batch: A1 **Roll No.:**
16010120015 -Yash Gavade

16010120006 - Dikshita chalke

16010120018 - Jinay jain

Experiment / assignment / tutorial No. 9

Grade: AA / AB / BB / BC / CC / CD /DD

Title: Implementation of Concurrency Control Protocols

Objective: To understand Transaction, Transaction Control Protocols and its implementation. Implement Lock based protocol.

Expected Outcome of Experiment:
CO 5: Formulate and demonstrate the transaction, concurrency and recovery techniques

Books/ Journals/ Websites referred:

1. Dr. P.S. Deshpande, SQL and PL/SQL for Oracle 10g.Black book, Dreamtech Press
2. www.db-book.com
3. Korth, Slberchatz, Sudarshan : "Database Systems Concept", 5th Edition , McGraw Hill
4. Elmasri and Navathe,"Fundamentals of database Systems", 4th Edition,PEARSON Education.
5. https://dev.mysql.com/doc/refman/8.0/en/innodb-transaction-isolation-levels.html

Resources used:

Theory

In a multiprogramming environment where multiple transactions can be executed simultaneously, it is highly important to control the concurrency of transactions.

Concurrency control is provided in a database to:

i. enforce isolation among transactions.
ii. preserve database consistency through consistency preserving execution of transactions.

iii. resolve read-write and write-read conflicts.

Concurrency control protocols can be broadly divided into two categories −

1. Lock based protocols
2. Time stamp based protocols

Lock-based Protocols

Database systems equipped with lock-based protocols use a mechanism by which any transaction cannot read or write data until it acquires an appropriate lock on it. Locks are of two kinds −

1. Binary Locks − A lock on a data item can be in two states; it is either locked or unlocked.
2. Shared/exclusive − This type of locking mechanism differentiates the locks based on their uses. If a lock is acquired on a data item to perform a write operation, it is an exclusive lock. Allowing more than one transaction to write on the same data item would lead the database into an inconsistent state. Read locks are shared because no data value is being changed.

Timestamp Ordering Protocol

The timestamp-ordering protocol ensures serializability among transactions in their conflicting read and write operations. This is the responsibility of the protocol system that the conflicting pair of tasks should be executed according to the timestamp values of the transactions.

The timestamp of transaction Ti is denoted as TS(Ti).
Read time-stamp of data-item X is denoted by R-timestamp(X).
Write time-stamp of data-item X is denoted by W-timestamp(X).

Timestamp ordering protocol works as follows −

If a transaction Ti issues a read(X) operation −
If TS(Ti) < W-timestamp(X)
Operation rejected.
If TS(Ti) >= W-timestamp(X)
Operation executed.
All data-item timestamps updated.

If a transaction Ti issues a write(X) operation −
If TS(Ti) < R-timestamp(X)

Operation rejected.

If TS(Ti) < W-timestamp(X)

Operation rejected and Ti rolled back.

Otherwise, operation executed.

1.Read Lock

select * from patient

pid integer	allergy character (25)	fname character (25)	mname character (25)	lname character (25)	address character (25)	dob date	sex character (1)	previous_diseases character (50)
1	peanut	Sharma	Aditya	M	Mumbai	2002-02...	M	N
2	dust	Chawla	Gaurav	N	Mumbai	2002-03...	M	N
3	milk	Dias	Arvin	W	Mumbai	2002-12...	M	N
4	milk	Khetan	Shreya	V	Mumbai	2002-10...	F	N
5	peanut	Mehta	Aarav	N	Mumbai	2002-04...	M	N

```
begin;
lock table patient IN SHARE MODE;
end;
```

```
COMMIT

Query returned successfully in 98 msec.
```

2.Write Lock

```
begin;
lock table patient in access exclusive mode;
End;
```

```
COMMIT

Query returned successfully in 89 msec.
```

```
INSERT INTO patient VALUES(22,'nuts','ramchandra','saiguru','subramaniam',
'lokandwala', '2002-2-16','M','N')
```

```
INSERT 0 1

Query returned successfully in 94 msec.
```

select * from patient

pid integer	allergy character (25)	fname character (25)	mname character (25)	lname character (25)	address character (25)	dob date	sex character (1)	previous_diseases character (50)
2	dust	Chawla	Gaurav	N	Mumbai	2002-03...	M	N
3	milk	Dias	Arvin	W	Mumbai	2002-12...	M	N
4	milk	Khetan	Shreya	V	Mumbai	2002-10...	F	N
5	peanut	Mehta	Aarav	N	Mumbai	2002-04...	M	N
22	nuts	ramchandra	saiguru	subramaniam	lokandwala	2002-02...	M	N
22	nuts	Perscy	Jonathan	Gonzalvez	Mira Road	2002-02...	M	N

3.ROW LOCK USING FOR UPDATE:

select * from patient where pid=2 for UPDATE

Data Output Explain Messages Notifications

	pid [PK] integer	allergy character (50)	lastname character (20)	firstname character (20)	mname character (20)	address character (50)	dob date	sex character (10)
1	2	dust	Chawla	Gaurav	N	Mumbai	2002-03...	M

4.ROW LOCK USING LOCK IN SHARE MODE

```
begin;
lock table patient IN SHARE MODE;
end;
```

```
COMMIT

Query returned successfully in 98 msec.
```

	pid [PK] integer	allergy character (50)	lastname character (20)	firstname character (20)	mname character (20)	address character (50)	dob date	sex character (10)
1	2	dust	Chawla	Gaurav	N	Mumbai	2002-03...	M

Implementation of Lock Protocol:

Read Lock

SELECT * FROM SOLDIERS

```
1  Read Lock
2  SELECT * FROM SOLDIERS
3
```

Data Output | Explain | Messages | Notifications

	ssno [PK] integer	rno integer	name character varying (50)	dateofbirth date	rank character varying (25)	location character (25)	salary integer
1	601	14	SOMNATH SHARMA	1947-02-12	MAJOR	KASHMIR	800000
2	602	13	VIKRAM BATRA	1950-02-11	GENREAL MAJOR	GOA	300000
3	603	15	SATISH KOLI	1949-02-10	CAPTAIN	JAMMU	80000
4	604	16	SOHAM PATIL	1948-10-09	BRIGADIER	PUNJAB	140000
5	605	17	VISHESH NAIK	1947-09-08	COLONEL	DELHI	120000
6	606	12	NIRMAL SINGH	1946-08-07	GENREAL MAJOR	GOA	300000
7	607	18	YOGENDRA SINGH	1949-07-06	CAPTAIN	JAMMU	80000
8	610	21	OMKAR SINGH	1947-04-03	COLONEL	DELHI	120000
9	611	22	NIRMALJIT SINGH	1948-05-04	MAJOR	GUJARAT	90000
10	612	-22	NIRMA SINGH	1945-02-02	CAPTAIN	KASHMIR	100000
11	608	19	DYANCHAND THAPA	1948-06-05	BRIGADIER	PUNJAB	130000
12	609	20	VISHAL NAIK	1947-05-04	COLONEL	DELHI	130000

Write lock

BEGIN;

LOCK TABLE SOLDIERS IN ACCESS EXCLUSIVE MODE;

END;

```
1  BEGIN;
2  LOCK TABLE SOLDIERS IN ACCESS EXCLUSIVE MODE;
```

Data Output | Explain | Messages | Notifications

```
LOCK TABLE

Query returned successfully in 42 msec.
```

BEGIN;

LOCK TABLE SOLDIERS IN ACCESS EXCLUSIVE MODE;

END;

INSERT INTO SOLDIERS(

VALUES (629, 29, 'MANN SHAH', '1947/02/12', 'SUB MAJOR', 'KASHMIR', 80000));

```
8  BEGIN;
9  LOCK TABLE SOLDIERS IN ACCESS EXCLUSIVE MODE;
10 END;
11
12 INSERT INTO SOLDIERS(
13 VALUES (629, 29, 'MANN SHAH', '1947/02/12', 'SUB MAJOR', 'KASHMIR', 80000));
```

Data Output | Explain | Messages | Notifications

```
INSERT 0 1

Query returned successfully in 122 msec.
```

Row lock

SELECT * FROM SOLDIERS

```
1  Read Lock
2  SELECT * FROM SOLDIERS
3
```

Data Output | Explain | Messages | Notifications

	ssno [PK] integer	rno integer	name character varying (50)	dateofbirth date	rank character varying (25)	location character (25)	salary integer
1	601	14	SOMNATH SHARMA	1947-02-12	MAJOR	KASHMIR	800000
2	602	13	VIKRAM BATRA	1950-02-11	GENREAL MAJOR	GOA	300000
3	603	15	SATISH KOLI	1949-02-10	CAPTAIN	JAMMU	80000
4	604	16	SOHAM PATIL	1948-10-09	BRIGADIER	PUNJAB	140000
5	605	17	VISHESH NAIK	1947-09-08	COLONEL	DELHI	120000
6	606	12	NIRMAL SINGH	1946-08-07	GENREAL MAJOR	GOA	300000
7	607	18	YOGENDRA SINGH	1949-07-06	CAPTAIN	JAMMU	80000
8	610	21	OMKAR SINGH	1947-04-03	COLONEL	DELHI	120000
9	611	22	NIRMALJIT SINGH	1948-05-04	MAJOR	GUJARAT	90000
10	612	-22	NIRMA SINGH	1945-02-02	CAPTAIN	KASHMIR	100000
11	608	19	DYANCHAND THAPA	1948-06-05	BRIGADIER	PUNJAB	130000
12	609	20	VISHAL NAIK	1947-05-04	COLONEL	DELHI	130000

SELECT * FROM SOLDIERS WHERE SSNO=609 FOR UPDATE

END;

```
1   SELECT * FROM SOLDIERS WHERE SSNO=609 FOR UPDATE
```

Data Output | Explain | Messages | Notifications

	ssno [PK] integer	rno integer	name character varying (50)	dateofbirth date	rank character varying (25)	location character (25)	salary integer
1	609	20	VISHAL NAIK	1947-05-04	COLONEL	DELHI	130000

BEGIN;

LOCK TABLE SOLDIERS IN ACCESS SHARE MODE;

END;

```
4
5   BEGIN;
6   LOCK TABLE SOLDIERS IN ACCESS SHARE MODE;
7
```

Data Output | Explain | Messages | Notifications

```
LOCK TABLE

Query returned successfully in 33 msec.
```

```
15  SELECT * FROM SOLDIERS WHERE SSNO=629 FOR UPDATE
```

Data Output | Explain | Messages | Notifications

	ssno [PK] integer	rno integer	name character varying (50)	dateofbirth date	rank character varying (25)	location character (25)	salary integer
1	629	29	MANN SHAH	1947-02-12	SUB MAJOR	KASHMIR	80000

Post Lab Questions:

1. **Explain pitfalls of 2PL (Two Phase Locking) Protocol**

Ans.

A lock is a system object associated with a shared resource such as a data item of an elementary type, a row in a database, or a page of memory.

The protocol uses locks, applied by a transaction to data, which may block other transactions from accessing the same data during the transaction's life.

The 2PL protocol, locks are applied and removed in two phases:

1. Expanding phase: locks are acquired and no locks are released.
2. Shrinking phase: locks are released and no locks are acquired.

Two-phase locking may also limit the amount of concurrency that occurs in a schedule because a Transaction may not be able to release an item after it has used it. This may be because of the protocols and other restrictions we may put on the schedule to ensure serializability, deadlock freedom, and other factors

it is safely determined only when a transaction has completed processing and requested commit.

Conclusion:

Concurrency control is essential in DBMS for handling the simultaneous execution of transactions among various databases.

- In the Read lock, the data item can only read by the transaction.
- In the exclusive lock, the data item can be both reads as well as written by the transaction.
- In the row lock, the data item forces the locks to be taken only on rows.

Therefore We have understood the implementation of read locks in read and exclusive mode

Hence, we have successfully implemented read lock in read and exclusive mode in the expirements .