Karam Shbeb Group 6

Lab 10 – Steps Template

Ex 1

Step No.	Terminal 1	Step No.	Terminal 2
1	Start a transaction and display the accounts information. Begin; Select * from accounts;	2	Start a transaction and update the username for “Alice Jones” as “ajones” Begin; Update accounts set username = 'ajones' where username = 'jones';
3	Display again the accounts table	4	Display again the accounts table
Do both terminals show the same information? Explain the reason Accounts table are the same in both cases even when the transaction is not done yet. That happens because a query in the current transaction cannot read data modified by another transaction that has not yet committed, thus preventing **dirty reads**			
		5	Commit the changes and compare again both sessions.
		6	Start a new transaction
7	Update the balance for the Alice’s account by +10.	8	Update the balance for the Alice’s account by +20 **The terminal will get block, so just about the operation using ctrl + C**
Explain the output form the second terminal When we use Read committed the balance will be +20 because the first update in terminal 1 is lost and that is due to the inability of this isolation to protect data update When we use Repeatable Read the balance will be +30 because +10 and +20 will be added to the balance together and that is due to the ability for this isolation to protect data update			

9	Commit the changes.	10	Rollback

Ex2.

Step No.	Terminal 1	Step No.	Terminal 2

1	Read users with group_id=2
	\set AUTOCOMMIT off
	Begin;
	set transaction isolation level
	Repeatable read;
	Select * from accounts where
	group_id = 2;
		2	Move Bob to group 2
			\set AUTOCOMMIT off
			Begin;
			set transaction isolation level
			Repeatable read;
			Update accounts set group_id = 2
			where username = 'bbrown';

3	Read users with group_id=2		
4	Update selected users balances by +15		
5	Read users with group_id=2	6	Read users with group_id=2
7	Commit transaction	8	Commit transaction
Explain the result for this isolation levels. Read committed: The results are not the same on both terminals and the reason behind that is the updating +15 and the inability of this isolation to protect the data from non-repeatable read. Repeatable read: The results are the same on both terminals and that is because this isolation level protects the data from non-repeatable read.			

Step No.	Terminal 1	Step No.	Terminal 2

1	Read users with group_id=2
	\set AUTOCOMMIT off
	Begin;
	set transaction isolation level
	Read Committed;
	Select * from accounts where
	group_id = 2;
		2	Move Bob to group 2
			\set AUTOCOMMIT off
			Begin;
			set transaction isolation level
			Read Committed;
			Update accounts set group_id = 2
			where username = 'bbrown';

3	Read users with group_id=2		
4	Update selected users balances by +15		
5	Read users with group_id=2	6	Read users with group_id=2
7	Commit transaction	8	Commit transaction
Explain the result for this isolation levels.			

Ex 3

Step No.	Terminal 1	Step No.	Terminal 2
1	Start a transaction in both terminals Begin;	2	Start a transaction in both terminals Begin;
3	Set the same transaction isolation level set transaction isolation level **Repeatable read**;	4	Set the same transaction isolation level set transaction isolation level **Repeatable read**;
5	Read the sum of users balances with group_id=2	6	Move Bob to group 2 (T2).
7	Read accounts with group_id=2		
8	Update selected accounts' balances by +sum		
9	Read users with group_id=2	10	Read users with group_id=2
11	**Commit;**	12	Commit;
Explain the result for this isolation level. Both isolation levels (Repeatable read, Serializable) protect the data from non-repeatable read. Then the read in the first terminal will not include Bob in the second group because Bob is moved to the second group in the second terminal			

Step No.	Terminal 1	Step No.	Terminal 2
1	Start a transaction in both terminals	2	Start a transaction in both terminals
3	Set the same transaction isolation level set transaction isolation level **serializable**;	4	Set the same transaction isolation level set transaction isolation level **serializable**;
5	Read the sum of users balances with group_id=2	6	Move Bob to group 2 (T2).
7	Read accounts with group_id=2		
8	Update selected accounts' balances by +sum		
9	Read users with group_id=2	10	Read users with group_id=2
11	Commit;	12	Commit;
Explain the result for this isolation level.			