Laboratory Work №10 — In-Class Task

SQL Transactions and Isolation Levels

Initial Setup

Use the following pre-created tables:

```
-- tickets table
| id | event        | available | price  |
|----|--------------|-----------|--------|
| 1  | Concert      | 100       | 50.00  |
| 2  | Theater      | 50        | 75.00  |
| 3  | Sports Game  | 200       | 35.00  |

-- wallets table
| id | user_name | balance  |
|----|-----------|----------|
| 1  | Emma      | 500.00   |
| 2  | James     | 300.00   |
| 3  | Sophie    | 150.00   |
```

Task 1: What Happens Next?

For each scenario, write what the final state will be:

Scenario A:

```
BEGIN;
UPDATE wallets SET balance = balance + 100.00 WHERE user_name
= 'Emma';
UPDATE wallets SET balance = balance - 50.00 WHERE user_name
= 'James';
ROLLBACK;
```

Emma's final balance: _____ James's final balance: _____

Scenario B:

```
BEGIN;
UPDATE tickets SET available = available - 5 WHERE event =
'Concert';
COMMIT;
```

```
UPDATE tickets SET available = available - 3 WHERE event =
'Concert';
ROLLBACK;
```
Concert tickets available: _____

(Hint: What happens to statements after COMMIT?)

Scenario C:

```
BEGIN;
DELETE FROM wallets WHERE user_name = 'Sophie';
SAVEPOINT before_delete;
INSERT INTO wallets (user_name, balance) VALUES ('Sophie',
200.00);
ROLLBACK TO before_delete;
COMMIT;
```
Does Sophie exist in the table? □ Yes □ No

If yes, what is her balance? _____

Task 2: Concurrent Transactions Timeline

Two users are buying concert tickets simultaneously. Follow the timeline:

Time	User A (READ COMMITTED)	User B
T1	`BEGIN;`	
T2	`SELECT available FROM tickets WHERE event='Concert';`	
T3		`BEGIN;`
T4		`UPDATE tickets SET available = available - 10 WHERE event='Concert';`
T5		`COMMIT;`
T6	`SELECT available FROM tickets WHERE event='Concert';`	
T7	`COMMIT;`	

Initial available tickets: 100

2.1 What value does User A see at T2? _____ (1 point)

2.2 What value does User A see at T6? _____ (1 point)

2.3 If User A used SERIALIZABLE instead of READ COMMITTED, what would User A see at T6? _____ (1 point)

2.4 What type of read phenomenon occurred in the READ COMMITTED scenario? (1 point)

□ Dirty Read
□ Non-Repeatable Read
□ Phantom Read

Task 3: ACID Properties

Match each scenario with the ACID property it demonstrates:

Property
A. Atomicity
B. Consistency
C. Isolation
D. Durability

3.1 After COMMIT, even if the power goes out, the data changes are preserved when the system restarts.

Answer: ___

3.2 A transaction transferring money ensures the total amount in all accounts remains the same (no money created or lost).

Answer: ___

3.3 If a transaction fails halfway through, none of its changes are applied to the database.

Answer: ___

3.4 Two transactions running at the same time don't interfere with each other's operations.

Answer: ___

Task 4: Write the Transaction

Write a transaction that:

1. Deducts $75 from Emma's wallet
2. Decreases Theater tickets by 1
3. Uses a savepoint after step 1
4. Commits the changes

```
_____________;

UPDATE wallets SET balance = balance - 75.00
    WHERE user_name = 'Emma';

_____________ after_payment;

UPDATE tickets SET available = available - 1
    WHERE event = 'Theater';

_____________;
```

Quick Reference

Command	Description
BEGIN	Start a transaction
COMMIT	Save all changes permanently
ROLLBACK	Undo all changes
SAVEPOINT name	Create a checkpoint
ROLLBACK TO name	Return to checkpoint
RELEASE SAVEPOINT name	Remove checkpoint

Isolation Level	Dirty	Non-Repeatable	Phantom
READ UNCOMMITTED	Yes	Yes	Yes
READ COMMITTED	No	Yes	Yes
REPEATABLE READ	No	No	Yes
SERIALIZABLE	No	No	No