A logo on a black background

Description automatically generated

Lab 4

# Concurrency

# Advanced Database Topics

### Prepared By:

### Niket Bhavesh Bhatt

### Nandha Kumar Reddy Kaipa

### Devendra Bahadur Shahi

1. **Setup (10 marks)**

Created a database named `pos` and a `Products` table with columns `Id`, `Name`, and `Quantity`. Inserted three records: Mobile (Id=1001, Quantity=10), Tablet (Id=1002, Quantity=20), and Laptop (Id=1003, Quantity=30). The screenshot below shows the table creation and data insertion results.

Solution:

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1 :pos db SETUP

1. **Dirty Read Concurrency Problem**

**Problem Explanation:**

A Dirty Read occurs when Transaction 2 reads uncommitted data (Quantity = 5) from Transaction 1, which is later rolled back to Quantity = 10, causing inconsistency. The screenshot `Q2\_DirtyRead\_Problem\_T1` and ‘Q2\_DirtyRead\_Problem\_T2’ shows Transaction 2 reading Quantity = 5 during Transaction 1’s uncommitted update.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 2: Q2\_DIrtyread\_PROBLEM\_T1

A screenshot of a computer

AI-generated content may be incorrect.

Figure 3: Q2\_DIrtyread\_PROBLEM\_T2

**Solution Explanation:**

Using the `READ COMMITTED` isolation level ensures Transaction 2 only reads committed data, preventing the Dirty Read. The screenshot `Q2\_DirtyRead\_Solution\_T1` shows Transaction 2 reading Quantity = 10 after Transaction 1’s rollback.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 4: Q2\_DIrtyread\_Solution\_T1

1. **Lost Update Concurrency Problem**

**Problem Explanation:**

A Lost Update occurs when Transaction 1 and Transaction 2 both read Quantity = 10 and update independently. Transaction 2 sets Quantity = 8, but Transaction 1 overwrites it to Quantity = 9, losing the -2 update. The final quantity should be 7 (10 - 1 - 2). The screenshot `Q3\_LostUpdate\_Problem\_T1` and ‘Q3\_LostUpdate\_Problem\_T2’ shows the incorrect Quantity = 9.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 5: Q3\_LostUpdate\_problem\_t1

A screenshot of a computer

AI-generated content may be incorrect.

Figure 6: Q3\_LOSTUPDATE\_problem\_t2

**Solution Explanation:**

Using `REPEATABLE READ` with `UPDLOCK` locks the row during the read, ensuring Transaction 2 waits for Transaction 1 to commit. This results in the correct final quantity of 7 (10 - 1 - 2). The screenshot `Q3\_LostUpdate\_Solution\_t2` confirms Quantity = 7.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 7: Q3\_LOSTUPDATE\_Solution\_t2

**4. Phantom Read Concurrency Problem**

**Problem Explanation:**

A Phantom Read occurs when Transaction 1 reads male employees twice, and Transaction 2 inserts a new male employee (Sam) between the reads, causing the second read to show an additional row (3 rows instead of 2). The screenshot `Q4\_PhantomRead\_Problem\_T1` shows the first read with 2 rows and the second with 3 rows.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 8: Q4\_Phantomread\_problem\_TABLE\_CREATION

A screenshot of a computer

AI-generated content may be incorrect.

Figure 9: Q4\_Phantomread\_problem\_T1

**Solution Explanation:**

Using `SERIALIZABLE` isolation level locks the range of male employees, preventing Transaction 2 from inserting new rows until Transaction 1 completes. Both `SELECT` statements show 2 rows. The screenshot `Q4\_PhantomRead\_Solution\_T2` confirms this consistency.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 10: Q4\_Phantomread\_PROBLEM\_T2

A screenshot of a computer

AI-generated content may be incorrect.

Figure 11: Q4\_Phantomread\_SOLUTION\_T2

A screenshot of a computer

AI-generated content may be incorrect.

Figure 12: Q4\_Phantomread\_SOLUTION\_T1

# CONFIDENTIALITY AGREEMENT & STATEMENT OF HONESTY

I, **Devendra Bahadur Shahi** verify that the submitted work is my own, original work, and that I did not use Generative AI tools (e.g., ChatGPT, Bard) to produce this lab report. I confirm knowing that a mark of 0 may be assigned for sharing or copying this work.

**Devendra Bahadur Shahi Devendra Bahadur Shahi 110187102**

**Student Signature Student Name Student I.D.**

I, **Nandha Kumar Reddy Kaipa** verify that the submitted work is my own, original work, and that I did not use Generative AI tools (e.g., ChatGPT, Bard) to produce this lab report. I confirm knowing that a mark of 0 may be assigned for sharing or copying this work.

**Nandha Kumar Reddy Nandha Kumar Reddy 110186083**

**Student Signature Student Name Student I.D.**

I, **Niket Bhavesh Bhatt** verify that the submitted work is my own, original work, and that I did not use Generative AI tools (e.g., ChatGPT, Bard) to produce this lab report. I confirm knowing that a mark of 0 may be assigned for sharing or copying this work.

**Niket Bhavesh Bhatt Niket Bhavesh Bhatt 110181232**

**Student Signature Student Name Student I.D.**