

B. Tech CSE Second Year ▾

B. Tech CSE Third Year ▾

B. Tech CSE Fourth Year ▾

Interview Preparation ▾

Database Management System

Unit 1: Introduction to DBMS

Unit 2 : Relational data Model and Language & SQL

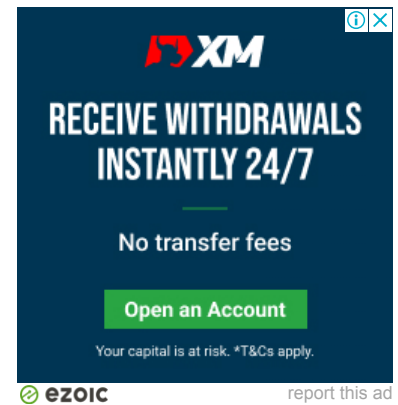
Unit 3 : Data Base Design & Normalization

Unit 4 : Transaction Processing Concept

Unit 5 : Concurrency Control Techniques

Problem faced when concurrent transactions are executing with example

The concurrency control technique is a process of managing the simultaneous execution of transactions in a shared database. However, when the concurrent transaction techniques are executed in an uncontrolled manner then it leads to several problems.



Latest Updates

Mostly Asked Java Interview

Questions For 2 Yrs Experience

Find All Pairs in Array whose

Sum is Equal to given number

The five leading problems that occur due to uncontrolled execution of concurrent transactions are

1. Lost Update Problem

Using Java

Java Program to find GCD of two
Numbers using Recursion

Python Program to Separate
Characters in a Given String

Python Program to add two
number using Recursion

Python Program to Find Highest
Frequency Element in Array

Python Program to Merge two
Arrays

Perform left rotation by two
positions in Array Using Python

Python Program to Delete
Element at Given Index in Array

Python Program to Delete
element at End of Array

Python Program to Copy one
String to Another String

Python Program to Remove
Repeated Character from String

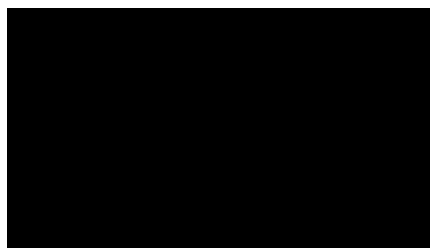
Print highest frequency
Character in String in Python

Python Program to Convert
lowercase vowel to uppercase

Convert Celsius to Fahrenheit in
Python

2. Dirty Read Problem

3. Unrepeated Read



4. Incorrect summary

5. Phantom Read


1. Lost Update Problem

A lost update problem appears when at least two transactions access the same transaction data and performs their operation. As we know, the data of transaction if the first read by the system and operation is performed and then write operation comes in the action which permanently saves the change in the actual database. Similarly, two or more transaction read the data from the database and performs their own operations but during the write operation sometimes the updated data from one transaction is overwritten by another transaction due to which the new data from the first transaction got lost and this situation is called lost update problem.

Lost Update Problem example

T1	T2
Read(X)	






RECEIVE
WITHDRAWALS
INSTANTLY
24/7

No transfer fees

Easy, fast &
secure

Via local Thai
banks



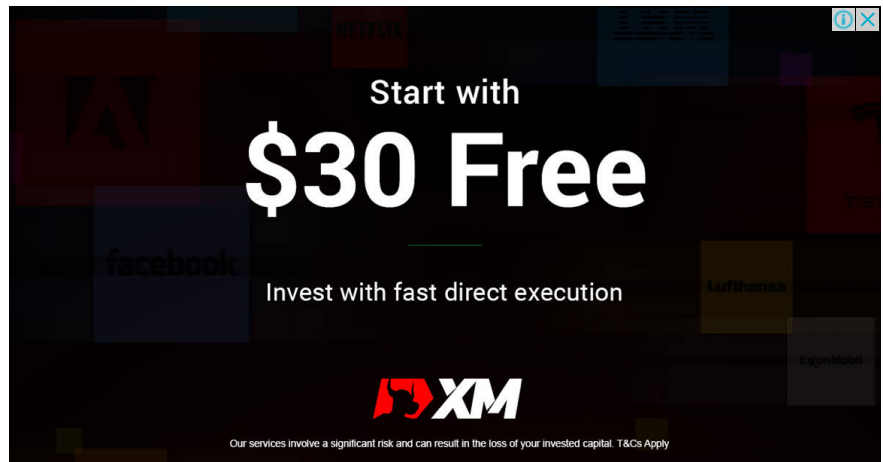
Open an Account

Your capital is at risk.
*T&Cs apply.

Read(X)	
	Read(X)
Update $X = X + 1000$	
	Update $X = X - 500$
Write(X)	
	Write(X)

Let two transactions T1 and T2.

The transaction first T1 reads the data X from the database, and then transaction T2 reads the same data from the database. After that T1 perform their operation to add 1000 to X, and then transaction T2 performs the operation to subtract 500 from the data which is read by T2 i.e. X, that T1 performs write operation which stores the value of X according to the changes made by T1 and then T2 runs write operation which again updates the value of X in database. This situation causes the loss of change of X which is performed by T1 as the T2 overwrites the X again after it gets updated by T1. We can also say that the update made by T1 is lost.



Start with
\$30 Free

Invest with fast direct execution

XM

Our services involve a significant risk and can result in the loss of your invested capital. T&Cs Apply

Open Your XM Account Now

Trade The Markets with Fast Direct Execution at Licensed and Regulated Broker XM

XM

[Sign up >](#)

RECEIVE WITHDRAWALS INSTANTLY 24/7

No transfer fees

Your capital is at risk. *T&Cs apply.

Open a

2. Dirty read problem

This is one of the problems that occur due to concurrent execution in an uncontrolled manner. In this problem, when two or more transactions run and one transaction reads the data value from the database and updates it with some operation but not completed, another transaction reads that updated value of data and completes its execution by the 'commit' statement. After completing of other transaction, the first transaction reads the same data or another data or performs another operation and get failed. According to the atomicity property of the transaction, the transaction rolled back and started from the beginning i.e. the old or raw value of that data. But the other transaction

which is already get completed by reading the old updated value of data is not considered an incorrect transaction as its data is considered as wrong data or dirty read. This whole scenario is called the dirty read problem.

IT MATTERS
WHERE YOU STAY™

SINGAPORE
BOOK NOW

H
FOR

Get up to
\$10500 Deposit Bonus*

Your capital is at risk. *T&Cs apply.

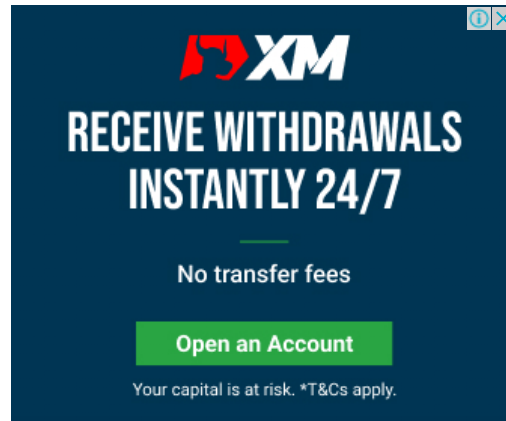

Open an Acc

Dirty read problem example

T1	T2
Read(Y)	
Update Y=Y+1000	
Write(Y)	
	Read(Y)
	Update Y=Y+500
	Write(Y)
	commit
Read(Z) // Failed	
Write(Z)	

Let two transactions T1 is execution and read a data Y from the database and then update the data by adding 1000 to data and then write statement is executed (but the transaction is not

finished). Then another transaction T2 comes in the action and it read the updated value of Y and performs subtracting 500 from Y and then executes the 'write' statement and with commit command, it completes its execution.



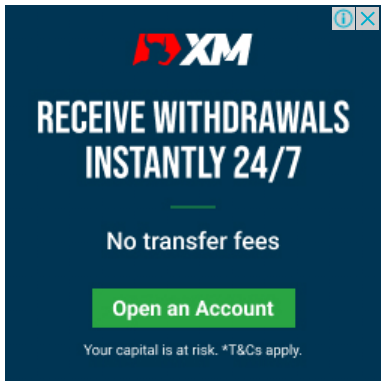
Then transaction T1 again performs read operation on some data Z and get failed. This failure will abort the transaction T1 and then the transaction T2 which has already been completed by reading the value of updated X which was the part of an aborted transaction, is now not a stable database transaction. Hence the data returned or written by transaction T2 is considered to be wrong.

3. Unrepeated Read problem

An unrepeated read appears when more than two transactions are executing the transaction data. In this problem, when a transaction starts executing data from the

database and performs a read operation, another transaction also reads the same data from the database. Then after the previous transaction, updated the value of that data and performed the write operation and then another operation again to read the updated value of data.

This time the data is changed as the first transaction performed some operations on that value, but the other transaction is unaware of it as for that transaction, both data values must be the same, but the data is found to be different when read operation is performed the second time. This causes the inconsistency of data.



Let’s understand this problem with the help of an example

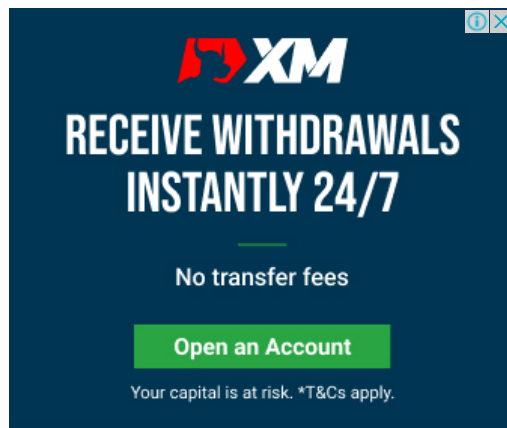
T1	T2
Read(X)	
	Read(X)
Update X=X+100	
Write(X)	
	Read(x)

Read(Y)	
Write (Y)	

Let two transactions be T1 and T2 where T1 starts executing and performing read operations on data item X from the database. Then T2 also reads the same value X from the database. Then T1 starts its operation to add 100 to X and performs a write operation on X. then again T2 read the value of X but this time the value id changes to X+100, which makes the transaction inconsistency as for T2 the value of X must be the same when read operation is performed on X second time.

4. Incorrect Summary problem

The incorrect summary problem occurs when more than two transactions execute their operations. One transaction calculates their aggregate function on data from the database while the other uploads the data value to the database. This leads to inconsistency. The result of one transaction is used as the input data value of another transaction.



Let us understand the situation
with the help of an example

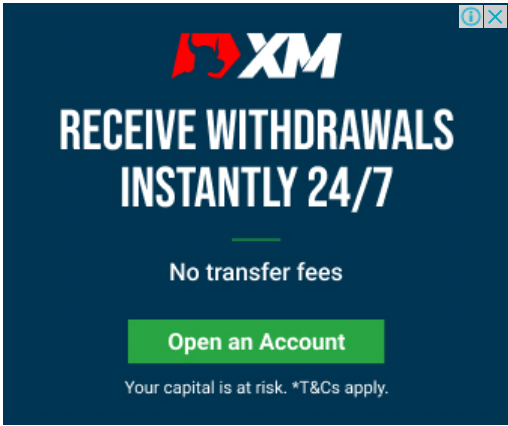
T1	T2
Read(X)	
Update $X=X+1000$	
Write(X)	
	Read(X)
	Update $X=X-500$
	Write(x)

Let two transactions T1 and T2.
The transaction T1 reads X from
the database, performs the
addition operation as $X+1000$ and
then performs the write operation
on X. Now transaction T2 reads the
value of X, which is updated to
 $X+1000$ and subtracts 500 from
the value, making the new value $X-500$
but transaction T2 may
calculate some values before they
have been updated by transaction
T2.

5. Phantom Read

problem

The phantom Read problem occurs when a transaction reads data from the database, and then another transaction reads the same database data and then the data is deleted as an operation of the first transaction. Now, when another transaction tries to read that data again, it does not find the data to read. It is a special case of an unrepeated read problem.

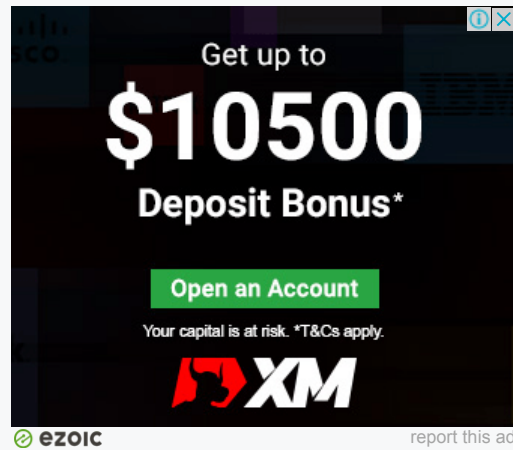


Let’s understand more with the help of an example

T1	T2
Read(X)	
	Read(X)
Delete(X)	
	Read(X)

Let transaction T1 read the data X from the database, and then transaction T2 read the same data X from the database. Now the transaction T1 deletes X from the database, and then T2 tries to read X again, which is already deleted

by T1, so the transaction will not be able to read the X.



Subjects

Data Structure
DBMS
Algorithm
Web Technology
Software Project
Management
Computer Network

f2f Interview Preparation

C Interview Questions
Java Interview Questions
Python Interview
Questions
Spring Boot Interview
Questions
Java 8 Interview Questions

Coding Interview Question

C Coding Questions
Java Coding Questions
Python Coding Questions
Java 8 Coding Questions






แผงเซลล์แสงอาทิตย์ราคาไม่แพง (ดูราคาโดย [คลิกที่นี่](#))

แผงเซลล์แสงอาทิตย์



If you own a mouse, you have to play this game. No Install. Play for free.

Panzer.Quest strategy game



ไม่เชื่อว่างานชิ้นไหนจะไม่มีปัญหา
สร้างผลงานเหนือจินตนาการด้วยพีเจอาร์ AI
โฉมใหม่ใน Creative Cloud

ซื้อเลยตอนนี้

