Transactions and Isolation Levels in SQL Server – Demos

This file contains the source code for the demonstrations.

Create Database Schema

```sql
USE master;
GO

CREATE DATABASE Sales;
GO

USE Sales;
GO

CREATE TABLE Employees(
  EmployeeID int primary key identity,
  Name nvarchar(50),
  Salary money
);
GO

INSERT INTO Employees(Name, Salary) VALUES ('Baj Ivan', 600);
INSERT INTO Employees(Name, Salary) VALUES ('Kaka Nadka', 1000);
INSERT INTO Employees(Name, Salary) VALUES ('Dyado Mraz', 2000);
INSERT INTO Employees(Name, Salary) VALUES ('Lelya Spaska', 1300);

CREATE TABLE Accounts(
  AccountID int primary key identity,
  Name nvarchar(50),
  Balance money
);
GO

INSERT INTO Accounts(Name, Balance) VALUES ('Bate Goiko', 500);
INSERT INTO Accounts(Name, Balance) VALUES ('Kaka Penka', 500);
```

Dirty Reads

First Transaction	Second Transaction

SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED
BEGIN TRAN
SELECT * FROM Employees
WAITFOR DELAY '00:00:10'
SELECT * FROM Employees
ROLLBACK
	BEGIN TRAN
	UPDATE Employees
	SET Salary = Salary * 1.2
	WAITFOR DELAY '00:00:10'
	ROLLBACK

Unrepeatable Reads

First Transaction	Second Transaction

SET TRANSACTION ISOLATION LEVEL READ COMMITTED
 	
BEGIN TRAN
 	
SELECT * FROM Employees
 	
WAITFOR DELAY '00:00:10'
 	
SELECT * FROM Employees
 	
ROLLBACK
	BEGIN TRAN
	
	UPDATE Employees
	SET Salary = Salary * 1.2
	
	COMMIT

Lost Updates

First Transaction	Second Transaction

SET TRANSACTION ISOLATION LEVEL READ COMMITTED
 	
BEGIN TRAN
 	
DECLARE @OldBalance money
 	
SELECT @OldBalance = Balance FROM Accounts
WHERE Name = 'Kaka Penka'
 	
WAITFOR DELAY '00:00:10'
 	
UPDATE Accounts
SET Balance = @OldBalance - 100
WHERE Name = 'Kaka Penka'
 	
UPDATE Accounts
SET Balance = @OldBalance + 100
WHERE Name = 'Bate Goiko'
 	
COMMIT TRAN
 	
SELECT * FROM Accounts
	BEGIN TRAN
	
	DECLARE @OldBalance money
	
	SELECT @OldBalance = Balance FROM Accounts
	WHERE Name = 'Kaka Penka'
	
	UPDATE Accounts
	SET Balance = @OldBalance - 100
	WHERE Name = 'Kaka Penka'
	
	UPDATE Accounts
	SET Balance = @OldBalance + 100
	WHERE Name = 'Bate Goiko'
	
	COMMIT TRAN
	
	SELECT * FROM Accounts

What will happen if we change the first transaction's isolation level to **REPEATABLE READ**?

Phantom Reads

First Transaction	Second Transaction

SET TRANSACTION ISOLATION LEVEL REPEATABLE READ
 	
BEGIN TRAN
 	
SELECT * FROM Employees
	BEGIN TRAN
	
	INSERT INTO Employees(Name, Salary)
	VALUES ('New Employee', 500);
	
	COMMIT

WAITFOR DELAY '00:00:10' SELECT * FROM Employees ROLLBACK	

Serializable Isolation

First Transaction	Second Transaction

SET TRANSACTION ISOLATION LEVEL SERIALIZABLE
 	
BEGIN TRAN
 	
SELECT * FROM Employees
 	
WAITFOR DELAY '00:00:10'
 	
SELECT * FROM Employees
 	
ROLLBACK
	BEGIN TRAN
	
	UPDATE Employees
	SET Salary = Salary * 1.2
	
	INSERT INTO Employees(Name, Salary)
	VALUES ('New Employee Again', 800);
	
	COMMIT

Snapshot Isolation

First Transaction	Second Transaction

ALTER DATABASE Sales
SET ALLOW_SNAPSHOT_ISOLATION ON
 	
SET TRANSACTION ISOLATION LEVEL SNAPSHOT
 	
BEGIN TRAN
 	
SELECT * FROM Employees
 	
WAITFOR DELAY '00:00:10'
 	
SELECT * FROM Employees
 	
ROLLBACK
 	
SELECT * FROM Employees
	BEGIN TRAN
	
	UPDATE Employees
	SET Salary = Salary * 1.2
	
	INSERT INTO Employees(Name, Salary)
	VALUES ('New Employee Snapshot', 800);
	
	COMMIT