



Module 7: Transactions and Isolation Levels



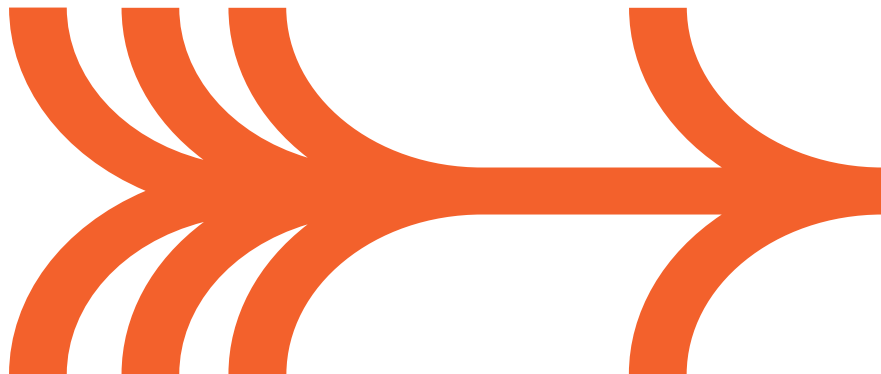
Transaction and isolation levels

- **Transactions**
 - Overview
 - Implementing transactions





Transactions



- A set of tasks that must all be completed successfully or aborted.
- A subset of tasks cannot be completed if some fail.
- Individual data modifications are committed automatically.
- **Cashpoint example:**
 1. Request £50
 2. Authorised
 3. £50 deducted from account
 4. £50 transaction added to transactions list
 5. Machine breaks and no money paid
 - What do you want to happen now?

QA TRANSACTIONS

Assuming the account has £1000 at the start of the transaction:

- Transaction works fine

Insert transaction	✓
Update balance	✓
Pay cash at machine	✓

Transactions		
ID	CustomerID	Amount
1234	12323456	(£50.00)

Accounts	
CustomerID	Balance
12323456	£950.00

- Cash machine fails to provide cash (not in a transaction)

Insert transaction	✓
Update balance	✓
Pay cash at machine	✗

Transactions		
ID	CustomerID	Amount
1234	12323456	(£50.00)

Accounts	
CustomerID	Balance
12323456	£950.00

Table updates committed

- Cash machine fails to provide cash (in a transaction)

Insert transaction	✓
Update balance	✓
Pay cash at machine	✗

Transactions		
ID	CustomerID	Amount

Accounts	
CustomerID	Balance
12323456	£1,000.00

Table updates rolled-back

QA BEGIN / COMMIT / ROLLBACK TRANSACTION

- BEGIN TRANSACTION
 - Start the transaction
- COMMIT TRANSACTION
 - Completes the transaction
 - All modifications are enforced
- ROLLBACK TRANSACTION
 - Completes the transaction
 - All modifications are undone
- SQL Server can automatically rollback the current transaction on error if XACT_ABORT setting is enabled.

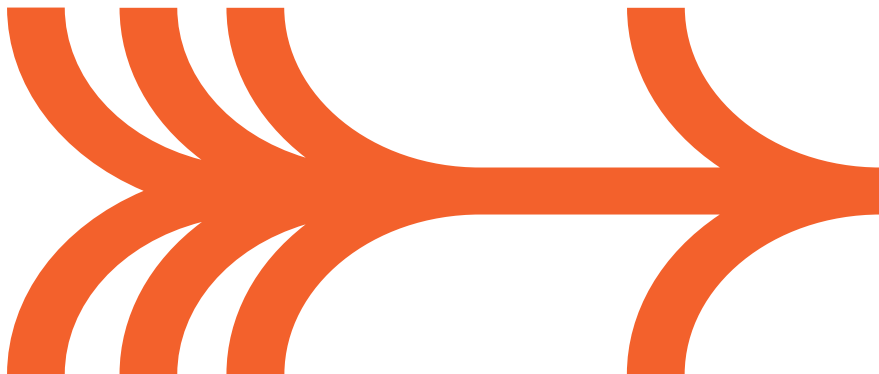
SET XACT_ABORT ON

- Can be combined with TRY / CATCH to gain more control.

Transaction example

Command outline:

```
BEGIN TRY
  BEGIN TRAN
    <some data modifications>
  COMMIT TRAN
END TRY
BEGIN CATCH
  ROLLBACK TRAN
  <code for the error>
END CATCH
```



QA TRANSACTION EXAMPLE (cont.)

Demonstration:

```
DECLARE @AccountNo INT = 12343422
DECLARE @Machine INT = 23321
DECLARE @Amount MONEY = 50.00

BEGIN TRY
    BEGIN TRAN
        INSERT INTO dbo.CashpointTrans(AcctNo, TranDT, MachineID, TranValue)
            VALUES (@AccountNo,getdate()),@Machine, @Amount)
        UPDATE dbo.CurrentAccount
            SET Balance = Balance – @Amount
            WHERE AcctNo = @AccountNo
    COMMIT TRAN
END TRY
BEGIN CATCH
    PRINT 'Error was caught';
    THROW
END CATCH
```



Exercise



- **Transactions**
 - Overview
 - Implementing transactions

Review

