BUSINESS ACADEMY
AARHUS

Databases and XML (4) – 29.2.2016

Today's agenda

Time	Activity
08.30	ACID
08.45	Transaction puzzle exercise
09.50	Triggers
10.05	Break
10.35	Trigger exercise
11.20	SQL assignment
12.00	Lunch
12.30	Mandatory assignment introduction
12.45	Video group exercise
13.35	Questions and exercises

- **Atomicity**
- **Consistency**
- **Isolation**
- **Durability**

ACID (transactions)

- **Atomicity** – either all changes in the transaction take place or none do, atomic unit
- A transaction is an indivisible unit that is either performed in its entirety or is not performed at all.
- It is the responsibility of the recovery subsystem of the DBMS to ensure atomicity

Atomicity

- **Consistency** – the transactions transforms the database from one consistent state to another following all integrity rules and constraints
- The DBMS can ensure consistency by enforcing all the constraints that have been specified on the database schema, such as integrity and enterprise constraints. However, in itself this is insufficient to ensure consistency.
- Example: money transfer. We have a transaction that is intended to transfer money from one bank account to another and the programmer makes an error in the transaction logic and debits one account but credits the wrong account, then the database is in an inconsistent state. However, the DBMS would not have been responsible for introducing this inconsistency and would have had no ability to detect the error.

- **Isolation** – Transactions execute independently of one another. In other words, the partial effects of incomplete transactions should not be visible to other transactions.
- Isolation can also be seen as a mechanism used to control access to data and ensure that transactions access data only if the data is in the level of consistency that those transactions expect (locking or row versioning).

- **Durability** – The effects of a successfully completed (committed) transaction are permanently recorded in the database and must not be lost because of a subsequent failure.
- Data changes are always written to the database's transaction log before they are written to the data portion of the database on disk

Durability

BUSINESS ACADEMY
AARHUS

Transactions (exercise)

Group 1
Mikkel O
Kaloyan
Nikolaj VT
Jens Christian

Group 2
Michelle
Terkel
Mikkel VB
Jakob

Group 3
Andreas B
Steffen P
Nikolay R
Elias

Group 4
Mikkel H
Andreas W
Martynas
Mike

Group 5
Simeon
Frederik
David
Natalia

Group 6
Troels
Adrian
Thomas
Martin

Group 7
Steffen BP
Mohamad
Simon Holt
Rostislav

Group 8
Christian
Vilius
Alice
Mihail

Group 9
Malik
Marc
Neli
Jannie

Groups for transaction exercise

Review the code

In your groups you shall review the code and get a proper understanding of:

- What does this code part do?
- Where does it fit into the Create Order Stored Procedure?

Team up with the other groups

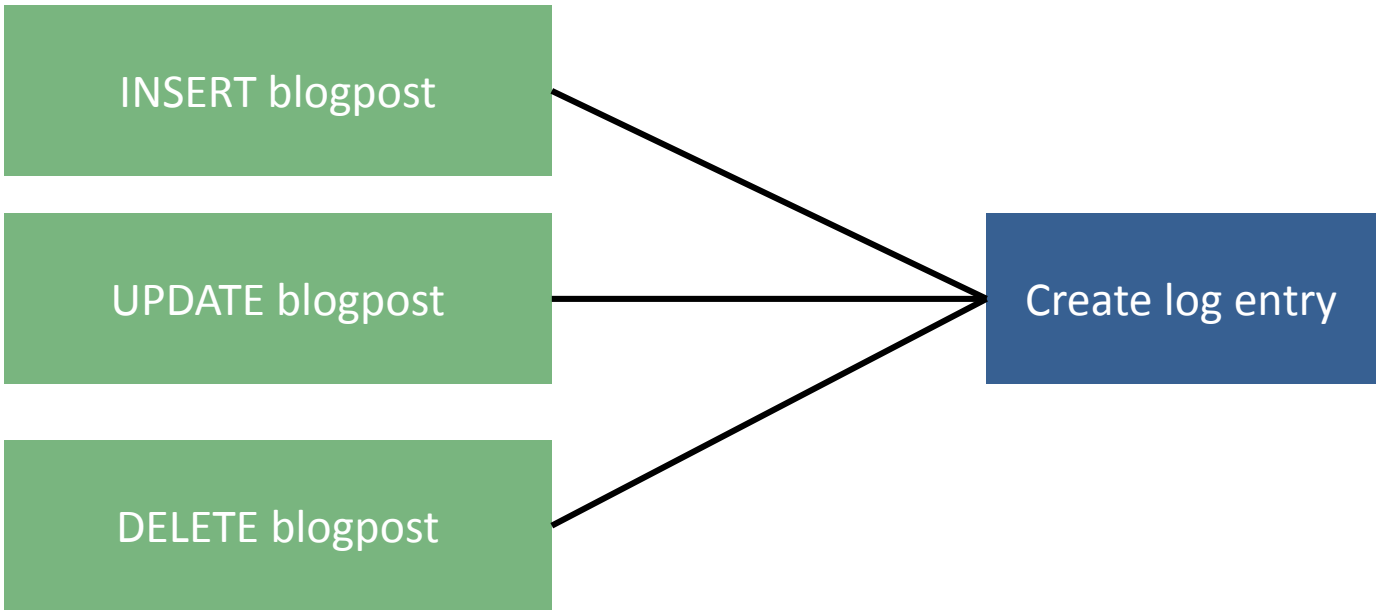
You shall now present your code part, get a presentation of a new code part and receive the code part in paper.

Assembling the parts

When you have a proper understanding of all the gathered code parts you shall assemble them to form a proper structure.

Triggers

BUSINESS ACADEMY
AARHUS


INSERT blogpost
UPDATE blogpost
DELETE blogpost
Create log entry

Event

```
CREATE TRIGGER <trigger_name>
ON <scope>
<trigger timing> <trigger condition>
AS
BEGIN
<trigger action>
END
```

Trigger syntax

```
CREATE TRIGGER inventory_minimum
ON stock
AFTER INSERT, UPDATE
AS
BEGIN
   EXEC msdb.dbo.sp_send_dbmail
     @profile_name = 'Inventory Mail',
     @recipients = 'supervisor@foo.com',
     @body = 'Someone changed the inventory.',
     @subject = 'Change Notification' ;
END
```

Name

Scope

Condition

Timing

Action

Trigger example

Develop a trigger that inserts a row in the ShopLog table each time a row is inserted into the ShopOrder table

Hint: There is a generic table named **INSERTED** which can be used for retrieving the inserted id in the ShopOrder table.

Extra! How is it possible to modify the trigger (or add multiple triggers) to handle both INSERT, UPDATE and DELETE and log these activities?

Hint: There are two tables; **INSERTED** and DELETED, but none **UPDATED** since it is a combination of the two others tables.

BUSINESS ACADEMY
AARHUS

Hints for mandatory assignment

- The three home assignments (Start of the blog, expansion of blog and automation of SQL in blog)
- In one PDF file – with explanations and screenshots
- No SQL code or database files included – SQL code only in screenshots

- Remember to take a screenshot of the *results* of the different queries
- Remember that a data model and a database diagram are not the same thing

- Database diagram to be included in the new automated database that you create in exercise 3

- If you have done your home work, this should be easy
- If you have not done your home work, you have *a lot* of work to do, so get started as soon as possible

- Help will be prioritized to assignment 1-3 – assignment 4 will only be given help is time allows

Mandatory assignment – the first part

Make a screencast (min. 2 mins - max. 7 mins) about one of these topics, which you get in a moment:

1. Stored Procedures
2. Views
3. Triggers
4. Transactions
5. Joins
6. Data modelling Crow's foot notation
7. Data modelling Chen's notation
8. Normalization
9. Functions

- Upload a link to your screencast on Fronter in the appropriate folder by next Friday (March 11th)

- The week after (March 14th), you will be giving each other feedback on the videos. It is therefore very important that you are all there for classes.

- If you cannot be there, you must send me an e-mail latest 8.00 the day of classes with the reason – and then you have to do the extra assignments of the written assignment in stead.

Are there difficult or even unbeatable assignments?

The art of selection – important and central topics to the presentation

Recording and editing video

1. Stored procedures
Mikkel O
Kaloyan
Nikolaj VT
Jens Christian

2. Views
Terkel
Jakob
Neli
Malik

3. Triggers
Andreas B
Steffen P
Nikolay R
Elias

4. Transactions
Mikkel H
Andreas W
Martynas
Mike

5. Joins
Simeon
Frederik
David
Natalia

6. Data modelling – Crow's foot
Troels
Adrian
Thomas
Martin

7. Data modelling – Chen's
Steffen BP
Mohamad
Simon Holt
Rostislav

8. Normalization
Christian
Vilius
Alice
Mihail

9. Functions
Marc
Jannie
Mikkel VB
Michelle

Mandatory assignment part 1 – in groups

Recap upon SQL and data modelling so far

Review the litterature that has been used so far

Work on the mandatory assignment (SQL 1-3)

Next week