

#### Tuntitehtävät vko 4

Puskurit. Transaktioidenhallinta. Suorituskyky. Monitorointi ja jäljitys. Hälytykset. IBM DB2:een tutustuminen:

- a) Asenna AdventureWorks tietokanta SQL Server:iin. AdventureWorks on Microsoftin harjoittelutietokanta SQL Server:iin. AdventureWorks tietokannasta on olemassa eri versiot melkein jokaiselle eri SQL Server versiolle (2005, 2008, 2012 versiot jne). Myös SQL Server Express:lle löytyy oma lite versionsa AdventureWorksLT. Tässä käytetään täys versiota SQL Server:stä, joten asenna täys versio myös AdventureWorks:sta. Tätä tietokantaa tarvitaan myöhemmissä harjoituksissa. Esitä mitä vaiheita tarvitaan kyseisen tietokannan asennuksessa. AdventureWorks tietokanta löytyy osoitteesta:  
<http://msftdbprodsamples.codeplex.com/>.

Vastaus:

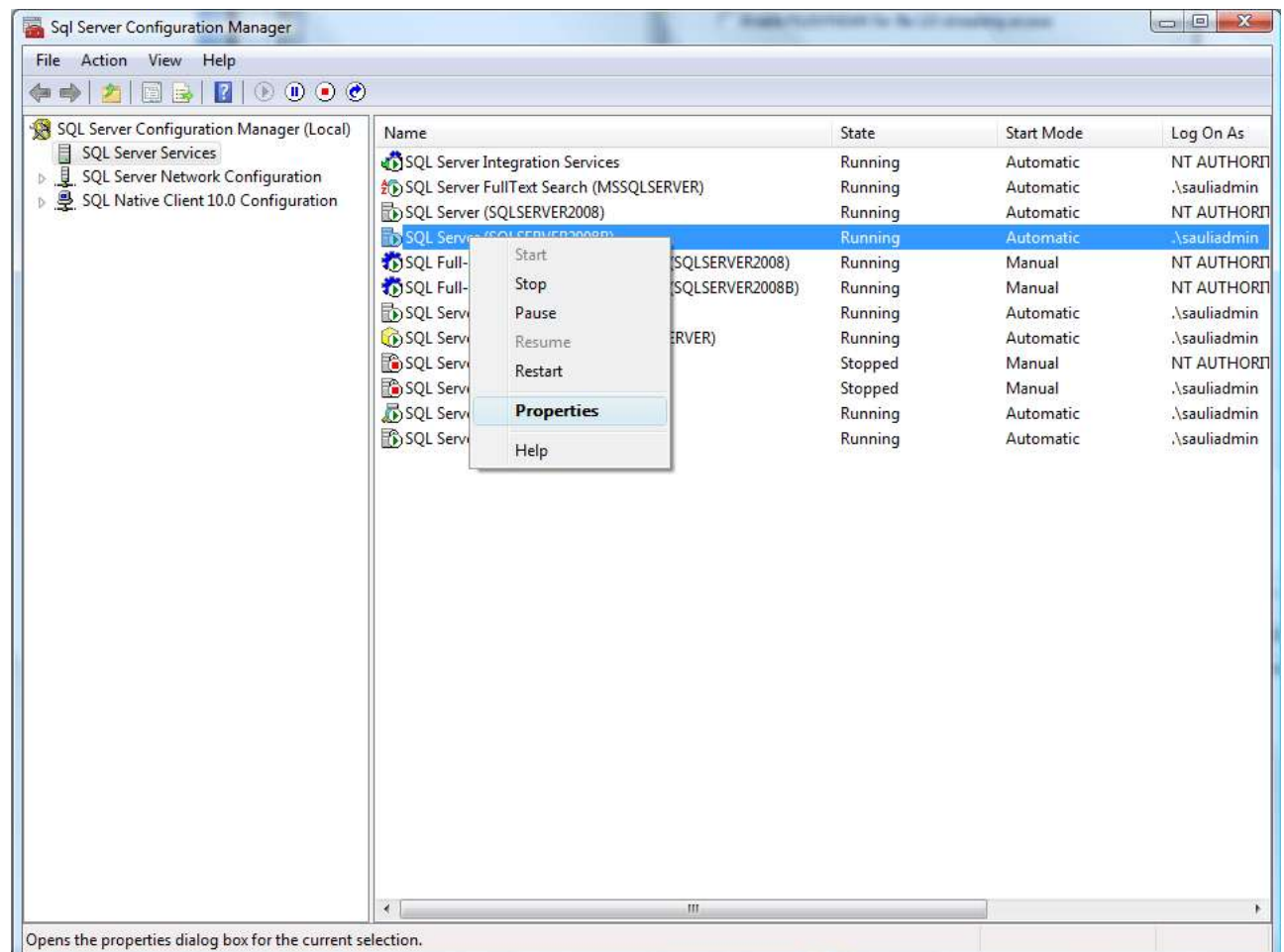
- AdventureWorks tietokanta on ladattava osoitteesta <http://msftdbprodsamples.codeplex.com/>. Sieltä valitaan AdventureWorks OLTP versio SQL Server 2005:lle. Vanhempi versio on mahdollista asentaa uudempaan versioon SQL Server:stä. Toisin päin se ei onnistu ilman SQL Server:in päivittämistä.
- Kopioi AdventureWorks\_Data.mdf ja AdventureWorks\_Log.ldf tiedostot SQL Server instanssiin käyttämään kansioon. Koulun koneilla esimerkiksi kansioon:

C:\Program Files\Microsoft SQL Server\MSSQL10.MSSQLSERVER\MSSQL\DATA

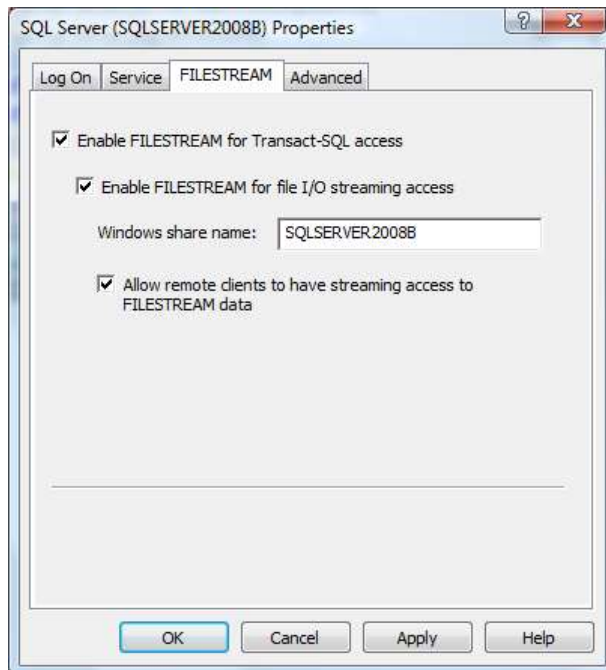
- AdventureWorks käyttää streamattuja tiedostoja tietokannassaan. Sen vuoksi SQL Server on konfiguroitava käyttämään niitä. SQL Server:in streamatuista tiedostoista on kerrottu lisää osoitteessa: <http://msdn.microsoft.com/en-us/library/bb895234.aspx> (huomaa valita se versio SQL Server:stä jota käytät sillä ohjeet eroavat toisistaan päivitysten vuoksi).

- Valitse SQL Server:in Configuration Manager:ista se SQL Server:in instanssi, jonka konfiguraatiota haluat muuttaa. Valitse siis:  
Start → All Programs → Microsoft SQL Server 2008 → Configuration Tools → SQL Server Configuration Manager

Ja klikkaa alla olevan kuvan mukaisesti instanssia hiiren oikealla ja valitse Properties:

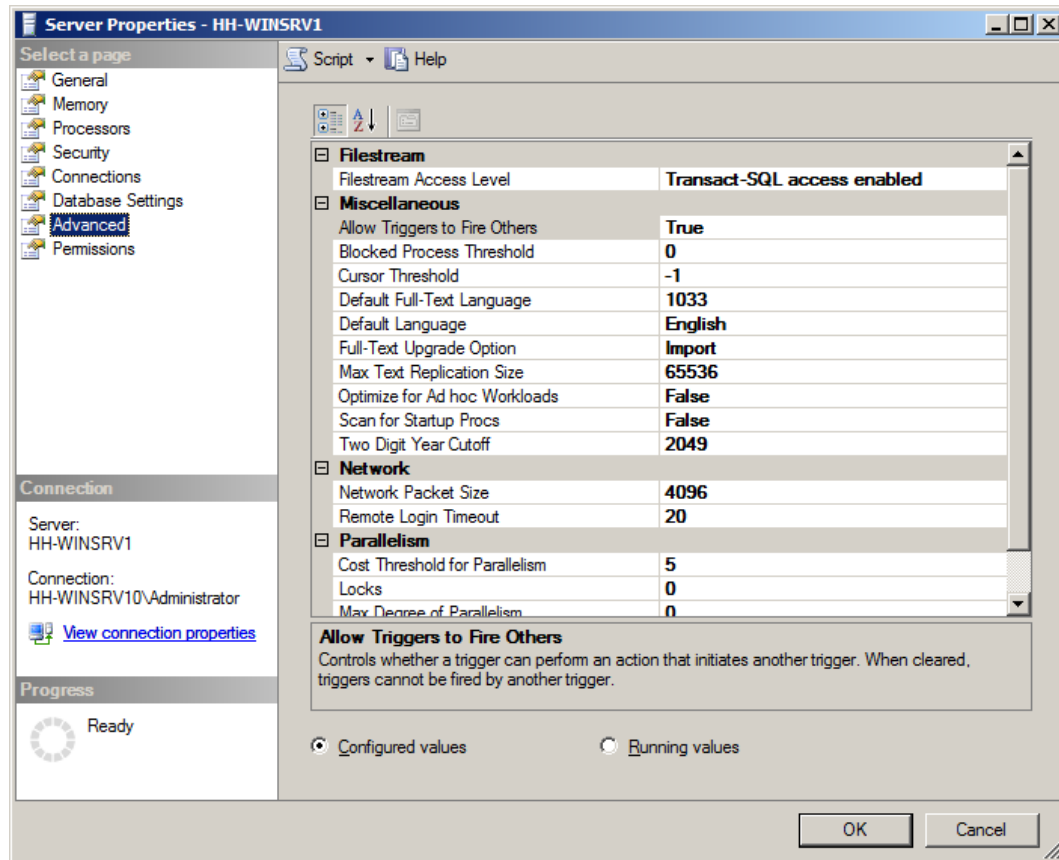


- Siirry Filesteam välilehdelle ja laita Filestream päälle (ks. kuva alla):

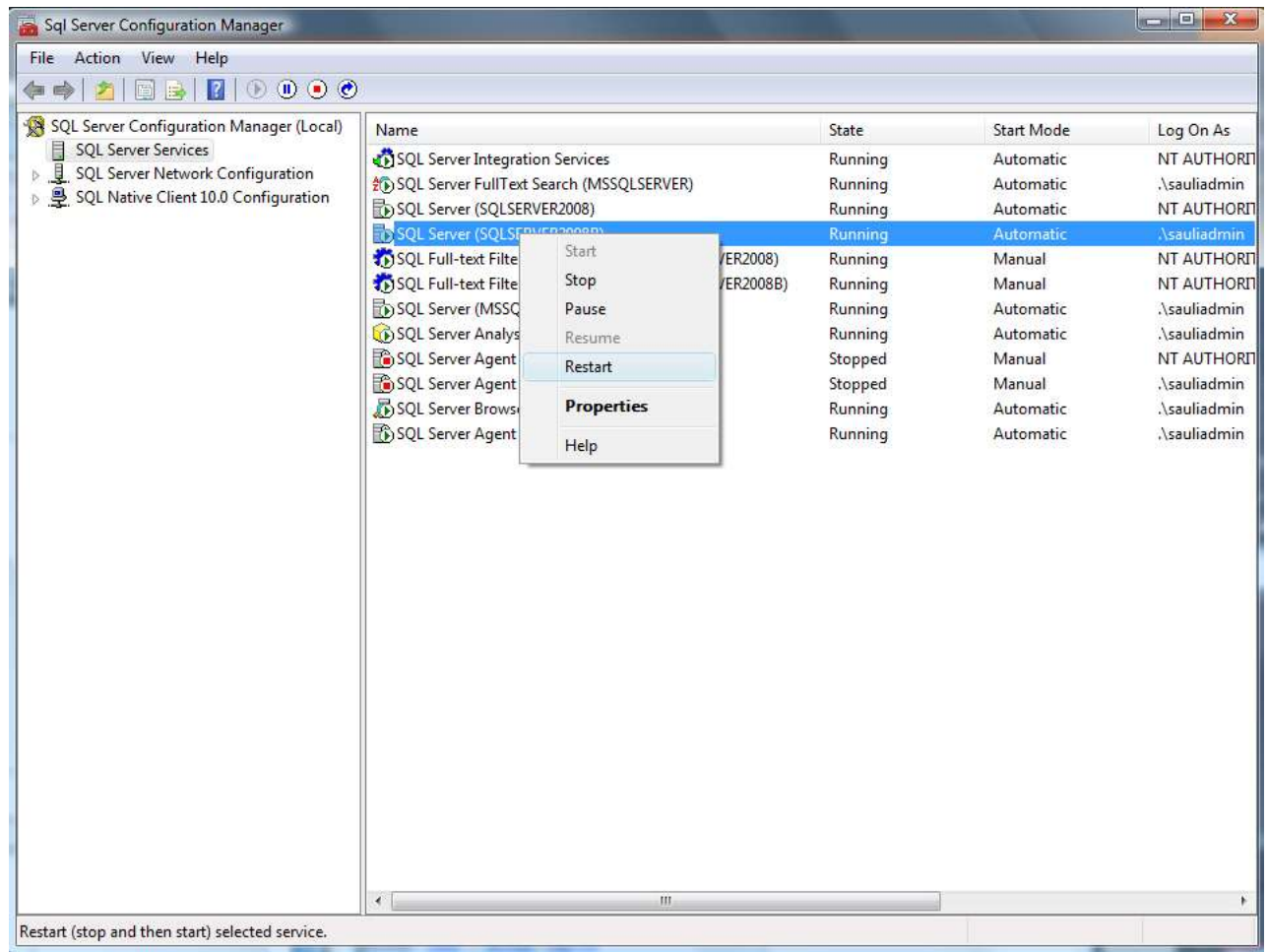


Hyväksy muutokset OK:lla.

- Aseta SQL Server Management Studiossa konfiguroitavalle instanssille Filestream Access Level alla olevan kuvan mukaiseksi, eli Transact-SQL access enabled:



- Käynnistä instanssi uudelleen SQL Server Configuration Managerin avulla.



- SQL Server Management Studiossa ajetaan seuraavaksi alla oleva T-SQL koodi (Query ikkunassa):

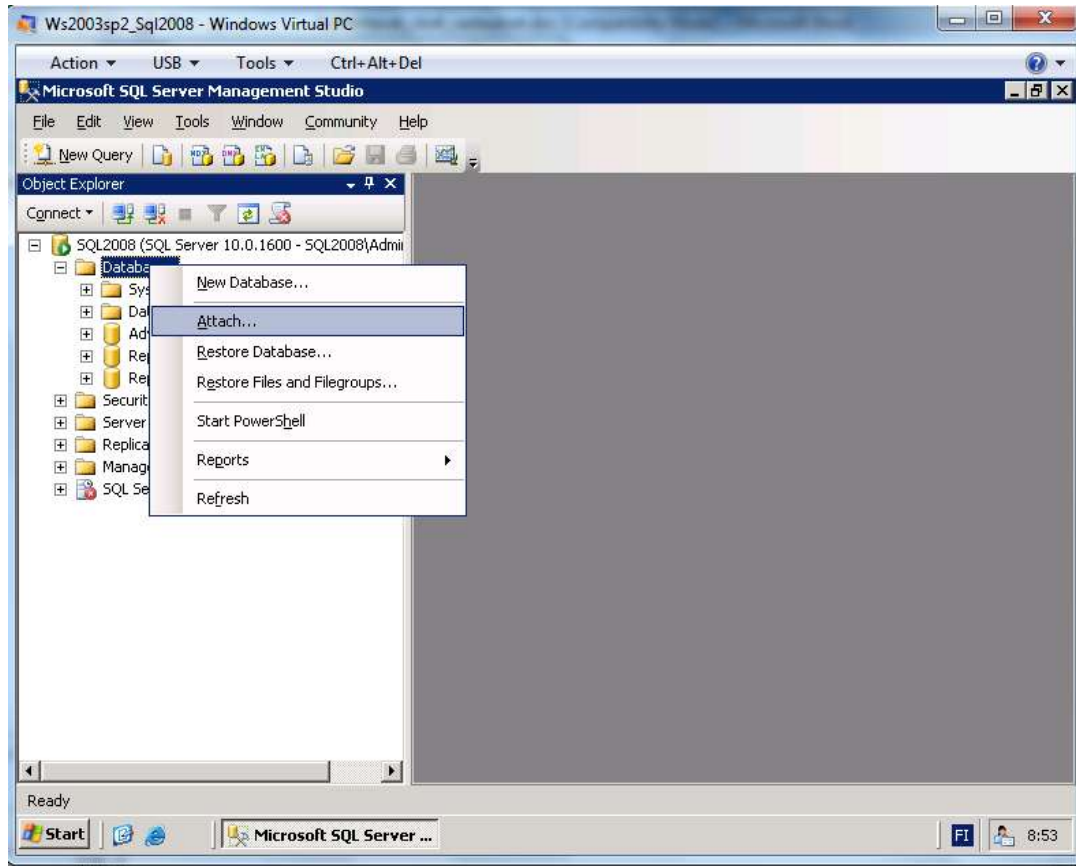
```
EXEC sp_configure filestream_access_level, 2  
RECONFIGURE
```

Onnistuneesta konfiguroinnin muutoksesta tulee alla oleva viesti:

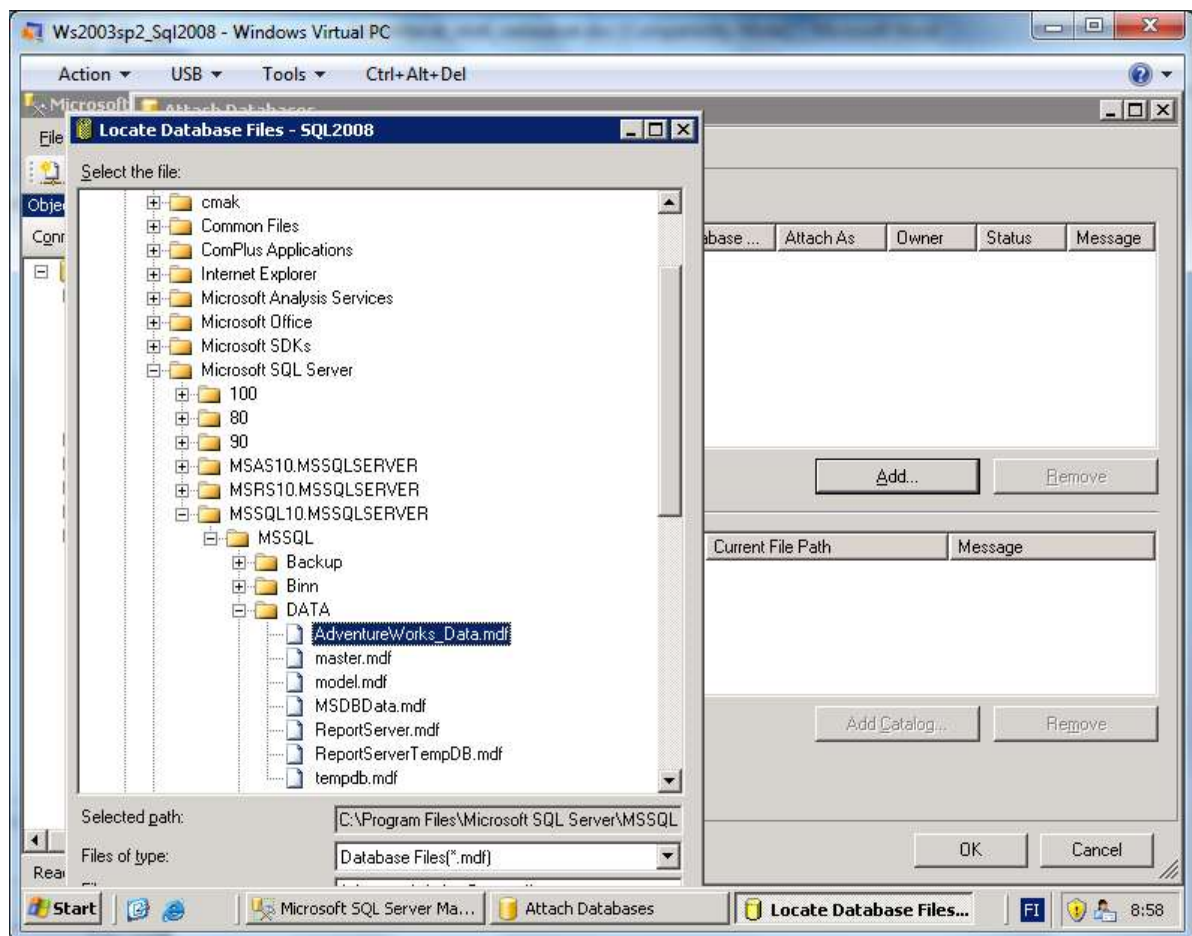
Configuration option 'filestream access level' changed from 0 to 2. Run the RECONFIGURE statement to install.

- Seuraavaksi SQL Server:in instanssi on käynnistettävä uudelleen. Se onnistuu suoraan SQL Server Configuration Manager:sta:

- Nyt päästään vihdoinkin liittämään eli attach:aamaan ladattu AdventureWorks\_Data.mdf ja AdventureWorks\_Log.ldf tiedostot. Näistä mdf on varsinainen tietokannan datatiedosto ja ldf on transaktiologin tiedosto. Siirry siis SQL Server Management Studioon ja klikkaa hiiren oikealla Databases:ia Object Explorer:ssa ja valitse Attach:



- Klikkaa Add painonappulaa ja valitse haluamasi tietokanta, jonka haluat liittää SQL Server:in instansiin.



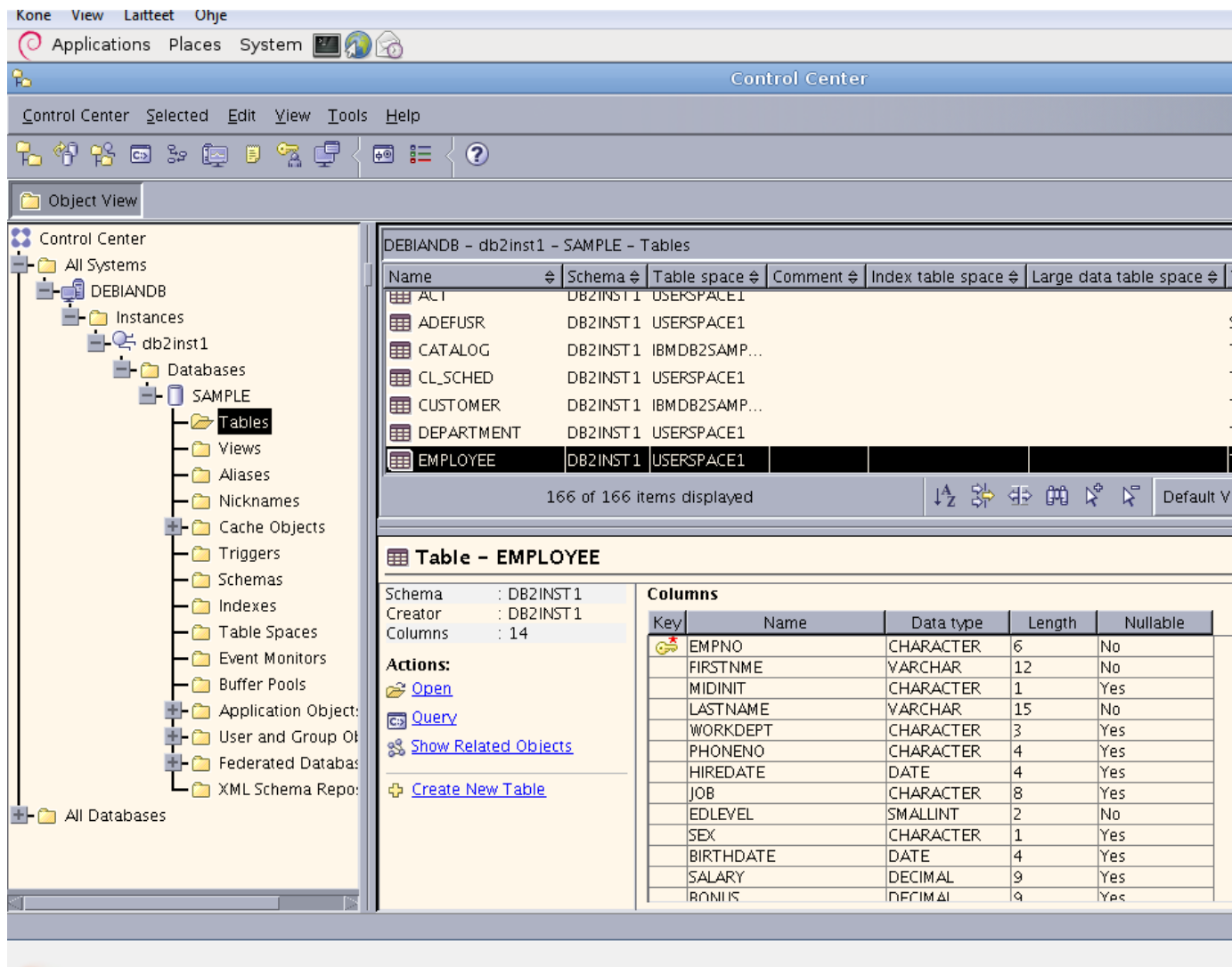
SQL Server kysyy Full Text Search Catalogia. Sitä ei välttämättä tarvita. AdventureWorks tietokannan voi liittää ilman sitä.

- b) Tutustu IBM:n DB2:een ja sen työkaluihin, tutkimalla esimerkiksi DB2:sta löytyvää SAMPLE tietokantaa. Tutki millainen tietokanta on taulujen, kenttien ja tietotyyppien, sekä indeksien osalta käyttäen IBM DB2 Control Center:iin. DB2 löytyy edellä jo tutuksi tulleesta VirtualBox:sta, johon Debian Linux on virtualisoitu.

Vastaus:

IBM DB2 Control Center:

- Tutkitaan ensin DB2:n Control Center:iä. IBM DB2 Control Center löytyy Debianista valitsemalla: Applications -> IBM DB2 -> Control Center



- Control Center:stä löytyy työkaluja tietokannan system catalogin tarkasteluun, taulujen luomiseen ja muuttamiseen. Myös DB2 SQL kielisiä lauseita voi ajaa Command Editorissa, valitsemalla Control Center:ssä:  
Tools -> Command Editor

Tähän työkaluun tutustutaan seuraavaksi, koska suurin osa ylläpidosta tehdään DB2 SQL kieltä käyttäen.

- Command Editorissa täytyy ensin kirjautua haluamaansa tietokantaan DB2:ssä DB2 SQL kieltä käyttäen. Kielen syntaksista löytyy tietoa osoitteesta:

<http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/index.jsp?topic=%2Fcom.ibm.db2z9.doc.sqlref>



[%2Fsrc%2Ftpc%2Fdb2z\\_sql\\_connect.htm.](#)

Ensin on siis kirjaututtava DB2:sta löytyvään tietokantaan. Esimerkiksi alla kirjaututaan SAMPLE nimiseen tietokantaan, koska sellainen löytyy sieltä. Käyttäjätunnus on db2inst1 ja perässä seuraa sen salasana.

```
connect to sample user db2inst1 using linstdb2;
```

- Kun tietokantaan on päästy onnistuneesti kirjautumaan, voidaan kokeilla mitä tahansa DB2 SQL:n lausetta. Alla on listattu sample tietokannasta löytyvän employee taulun kaikki data rivit:

```
select * from employee;
```

Alla olevassa kuvassa on esitetty millaiselta Command Editor näyttää:

The screenshot shows the IBM Command Editor 2 window. The title bar reads 'Command Editor 2'. The menu bar includes 'Command Editor', 'Selected', 'Edit', 'View', 'Tools', and 'Help'. The toolbar contains various icons for file operations and execution. The 'Commands' tab is active, showing the following text:

```
connect to sample user db2inst1 using linstdb2;
select * from employee;
```

Below the command input area, the results of the query are displayed in a table with 11 columns. The first row of data is:

EMPNO	NAME	SEX	EMPID	DEPT	EMPNO	NAME	SEX	EMPID	DEPT	EMPNO
000300	PHILIP	X	SMITH	E11	2095	08/19/2002	OPERATOR	14	M	10/27/1978
000310	MAUDE	F	SETRIGHT	E11	3332	09/12/1994	OPERATOR	12	F	04/21/1961
000320	RAMLAL	V	MEHTA	E21	9990	07/07/1995	FIELDREP	16	M	08/11/1962
000330	WING		LEE	E21	2103	02/23/2006	FIELDREP	14	M	07/18/1971
000340	JASON	R	GOUNOT	E21	5698	05/05/1977	FIELDREP	16	M	05/17/1956
200010	DIAN	J	HEMMINGER	A00	3978	01/01/1995	SALESREP	18	F	08/14/1973
200120	GREG		ORLANDO	A00	2167	05/05/2002	CLERK	14	M	10/18/1972
200140	KIM	N	NATZ	C01	1793	12/15/2006	ANALYST	18	F	01/19/1976
200170	KIYOSHI		YAMAMOTO	D11	2890	09/15/2005	DESIGNER	16	M	01/05/1981
200220	REBA	K	JOHN	D11	0672	08/29/2005	DESIGNER	18	F	03/19/1978
200240	ROBERT	M	MONTEVERDE	D21	3780	12/05/2004	CLERK	17	M	03/31/1984
200280	EILEEN	R	SCHWARTZ	E11	8997	03/24/1997	OPERATOR	17	F	03/28/1966
200310	MICHELLE	F	SPRINGER	E11	3332	09/12/1994	OPERATOR	12	F	04/21/1961
200330	HELENA		WONG	E21	2103	02/23/2006	FIELDREP	14	F	07/18/1971
200340	ROY	R	ALONZO	E21	5698	07/05/1997	FIELDREP	16	M	05/17/1956

At the bottom of the results, it says '42 record(s) selected.' The status bar at the bottom of the window shows 'Control Center' and 'Command Editor 2'.

- c) Seuraavaksi tutustutaan transaktioiden hallintaan tietokannoissa sekä lukkoihin ja niiden merkitykseen. Alla on esitetty komennot, joiden avulla aiheeseen tutustuminen tehdään SQL Server:ssä. Aluksi luodaan tietokanta pankkiA ja sinne luodaan taulu tilit. Tämän jälkeen lisätään tilit tauluun muutaman tilin saldot (summa sarakkeeseen). Nyt PankkiA tietokanta on valmis transaktioihin ja lukkoihin tutustumiseen.

Lukkojen tutkiminen

=====

käytä tietokantaa pankkiA

Ikkuna 1:

```
drop table tilit;
```

```
create table tilit  
(  
id int primary key,  
summa money);
```

```
insert into tilit (id, summa) VALUES (1,100);  
insert into tilit (id, summa) VALUES (2,100);
```

Ikkuna 2

```
select * from sys.dm_tran_locks
```

Vastaus:



	resource_type	resource_subtype	resource_database_id	resource_description	resource_associated_entity_id	resource_lock_partition	request_mode	request_type	request_status	request_reference_count	request_lifetime	request_session_id
1	DATABASE		21		0	0	S	LOCK	GRANT	1	0	54
2	DATABASE		21		0	0	S	LOCK	GRANT	1	0	52

Books Online:sta löytyy selitykset: <http://msdn2.microsoft.com/en-us/library/ms190345.aspx>

=====  
Transaktiot:

käytä tietokantaa pankkiA

```
drop table tilit
```

```
create table tilit  
(  
id int primary key,  
summa money)
```

```
insert into tilit (id, summa) VALUES (1,100);  
insert into tilit (id, summa) VALUES (2,100);
```

=====  
TESTI1:

Ikkuna 1

```
select * from tilit;
```

Ikkuna 2

```
begin transaction  
update tilit set summa = summa - 40 where id=1  
update tilit set summa = summa + 40 where id=2
```

```
select * from tilit;
```

Ikkuna 1

```
select * from tilit;
```

--> Miksi ei mitään näy?  
(muuta kuin Executing query )

Ikkuna 2  
commit;

Ikkuna 1  
Suoritti kyselyn loppuun

--> Mitä tapahtui

=====

TESTI 2

Ikkunal:  
delete from tilit;  
insert into tilit (id, summa) VALUES (1,100);  
insert into tilit (id, summa) VALUES (2,100);

Toista TESTI1, mutta tee lopuksi commit- käskyn sijaan rollback

Tehtävä 1: selitä käyttäytyminen kohdassa TESTI1 ja TESTI2

=====

TESTI 3

Ikkuna 1:  
  
drop table tilit  
create table tilit  
(  
id int primary key,  
summa money)

insert into tilit (id, summa) VALUES (1,100);  
insert into tilit (id, summa) VALUES (2,100);

Ikkuna 2

begin transaction  
update tilit set summa = summa - 40 where id=1  
update tilit set summa = summa + 40 where id=2  
  
select \* from tilit;

Ikkuna 1

```
SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED  
select * from tilit;  
COMMIT
```

Ikkuna 2  
rollback

Ikkuna 1

```
select * from tilit
```

Komento on selitetty Books Online:ssa osoitteessa: <http://msdn2.microsoft.com/en-us/library/ms173763.aspx>

d) Selitä käyttäytyminen verrattuna tehtävään c:

=====

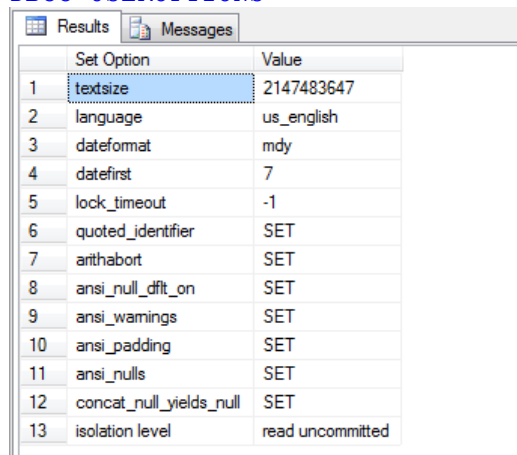
Mikä on käyttäjän Isolation Level?

- käytä asian selvittämiseen komentoa komentoa DBCC USEROPTIONS  
Mikä on isalotaion level ikkunoissa 1 ja 2

Vastaus:

Ikkuna 1:

DBCC USEROPTIONS



	Set Option	Value
1	textsize	2147483647
2	language	us_english
3	dateformat	mdy
4	datefirst	7
5	lock_timeout	-1
6	quoted_identifier	SET
7	arithabort	SET
8	ansi_null_dflt_on	SET
9	ansi_warnings	SET
10	ansi_padding	SET
11	ansi_nulls	SET
12	concat_null_yields_null	SET
13	isolation level	read uncommitted

Ikkuna 2:

DBCC USEROPTIONS

Results		Messages
	Set Option	Value
1	textsize	2147483647
2	language	us_english
3	dateformat	mdy
4	datefirst	7
5	lock_timeout	-1
6	quoted_identifier	SET
7	arithabort	SET
8	ansi_null_dflt_on	SET
9	ansi_warnings	SET
10	ansi_padding	SET
11	ansi_nulls	SET
12	concat_null_yields_null	SET
13	isolation level	read committed

Komennosta on kerrottu Books Online:ssa osoitteessa: <http://msdn2.microsoft.com/en-us/library/ms180065.aspx>

Sulje ikkunat ja avaa uudestaan, jotta Isolation level palautuu oletukseksi  
Mikä on oletusarvoinen Isolation Level

Vastaus:

Ikkuna 1:

DBCC USEROPTIONS

	Set Option	Value
1	textsize	2147483647
2	language	us_english
3	dateformat	mdy
4	datefirst	7
5	lock_timeout	-1
6	quoted_identifier	SET
7	arithabort	SET
8	ansi_null_dflt_on	SET
9	ansi_warnings	SET
10	ansi_padding	SET
11	ansi_nulls	SET
12	concat_null_yields_null	SET
13	isolation level	read committed

Ikkuna 2:

DBCC USEROPTIONS

	Set Option	Value
1	textsize	2147483647
2	language	us_english
3	dateformat	mdy
4	datefirst	7
5	lock_timeout	-1
6	quoted_identifier	SET
7	arithabort	SET
8	ansi_null_dflt_on	SET
9	ansi_warnings	SET
10	ansi_padding	SET
11	ansi_nulls	SET
12	concat_null_yields_null	SET
13	isolation level	read committed

=====

SNAPSHOT ISOLATION

käytä tietokantaa foo

```
drop table tr1
create table tr1
(
id int primary key,
riviSumma money)
```

```
insert into tr1 (id, riviSumma) VALUES (1,100);
insert into tr1 (id, riviSumma) VALUES (2,100);
```

```
USE foo;
GO
BEGIN TRAN
UPDATE tr1
SET riviSumma = 300
```

```
Ikuna 2:
use foo;
SELECT SUM(riviSumma) AS Yhteensa
FROM tr1
```

--> Kysely jää odottamaan

Ikkuna 1:

Rollback

Ikkuna 1:  
Sulje Ikkuna 2 (Avoin istunto estää seuraavan käskyn toiminnon)

```
ALTER DATABASE foo
SET READ_COMMITTED_SNAPSHOT ON;
```

Tee sama koe, eli

Ikkuna 1

```
USE foo;
GO
BEGIN TRAN
UPDATE tr1
SET riviSumma = 300
```

Ikkuna 2



```
use foo;  
SELECT SUM(riviSumma) AS Yhteensa  
FROM tr1
```

Tulos saadaan (tilanteesta ennen päivitystä)

=====

(sulje kyselyikkunat) ja avaa ne uudestaan, Muista Use foo

Jatketaan erityyvyystasolla

READ\_COMMITTED\_SNAPSHOT

Ikkuna 1:

```
select * from tr1
```

Rivit ovat alkuperäiset (kussakin summana 100)

Ikkuna 1:

```
Begin transaction tran1  
SELECT SUM(riviSumma) AS Yhteensa  
FROM tr1
```

Summa = 200

Ikkuna 2:

```
Begin transaction tran2
```

```
UPDATE tr1  
SET riviSumma = 300
```

```
commit
```

Ikkuna 1

```
SELECT SUM(riviSumma) AS Yhteensa  
FROM tr1
```

!! Sama transaktio, kuin aiemmin, mutta nyt eri tulos (Transactio 2 kommitoitu ja käytössä read committed erityyvyystaso

=====

Muutetaan tietokanta siten, että annetaan komento, jolla SNAPSHOT käyttäytyminen muutetaan

Muista käyttää tietokantaa foo

Sulje ikkuna 2

Anna ikkunassa 1 komennot, joilla lähtötilanne palautetaan

```
drop table tr1
create table tr1
(
id int primary key,
riviSumma money)
```

```
insert into tr1 (id, riviSumma) VALUES (1,100);
insert into tr1 (id, riviSumma) VALUES (2,100);
```

Sulje myös ikkuna 1, (istunto lopetetaan)

Muutetaan SNAPSHOT-käyttäytyminen

```
ALTER DATABASE foo
SET ALLOW_SNAPSHOT_ISOLATION ON
```

Ikkuna 1:

```
SET TRANSACTION ISOLATION LEVEL SNAPSHOT
Begin transaction tran1
SELECT SUM(riviSumma) AS Yhteensa
FROM tr1
```

Yhteensa = 200

Ikkuna 2:

Begin transaction tran2

```
UPDATE tr1
SET riviSumma = 300
```

commit

```
Ikkuna 1
SELECT SUM(riviSumma) AS Yhteensa
FROM tr1
```

Yhteensa = 200

Nyt transaktio käsittelee sen käynnistymisen hetken tilannetta, vaikka muutoksia on kommitoitu kantaan johtuen sen ISOLATION LEVEL asetuksesta

=====

