Lukkojen tutkiminen

=====================

käytä tietokantaa foo

Ikkuna 1:

drop table tr1

create table tr1

(

id int primary key,

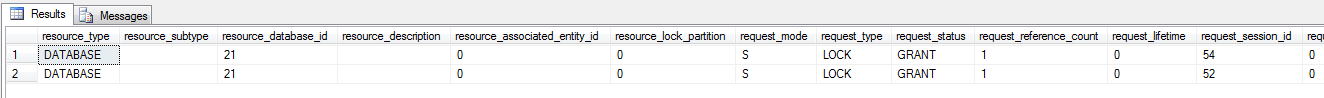
summa money)

insert into tr1 (id, summa) VALUES (1,100);

insert into tr1 (id, summa) VALUES (2,100);

Ikkuna 2

select \* from sys.dm\_tran\_locks

Vastaus:  


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

=====================================================

Transaktiot:

käytä tietokantaa foo

drop table tr1

create table tr1

(

id int primary key,

summa money)

insert into tr1 (id, summa) VALUES (1,100);

insert into tr1 (id, summa) VALUES (2,100);

=================================================

TESTI1:

Ikkuna 1

select \* from tr1;

Ikkuna 2

begin transaction

update tr1 set summa = summa - 40 where id=1

update tr1 set summa = summa + 40 where id=2

select \* from tr1;

Ikkuna 1

select \* from tr1;

--> Miksi ei mitään näy?

(muuta kuin Excecuting query )

Ikkuna 2

commit;

Ikkuna 1

Suoritti kyselyn loppuun

--> Mitä tapahtui

==============================================================

TESTI 2

Ikkuna1:

delete from tr1;

insert into tr1 (id, summa) VALUES (1,100);

insert into tr1 (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 tr1

create table tr1

(

id int primary key,

summa money)

insert into tr1 (id, summa) VALUES (1,100);

insert into tr1 (id, summa) VALUES (2,100);

Ikkuna 2

begin transaction

update tr1 set summa = summa - 40 where id=1

update tr1 set summa = summa + 40 where id=2

select \* from tr1;

Ikkuna 1

SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED

select \* from tr1;

COMMIT

Ikkuna 2

rollback

Ikkuna 1

select \* from tr1

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

Tehtävä 2: selitä käyttäytyminen verrattuna tehtävään 1

================================================================

Mikä on käyttäjän Isolation Level?

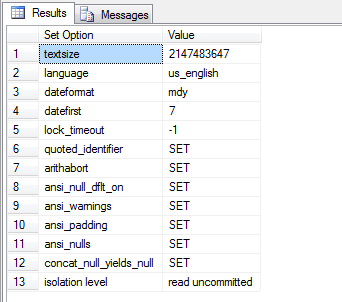
- käytä asian selvittämiseen komentoa komentoa DBCC USEROPTIONS

Mikä on isalotaion level ikkunoissa 1 ja 2

Vastaus:

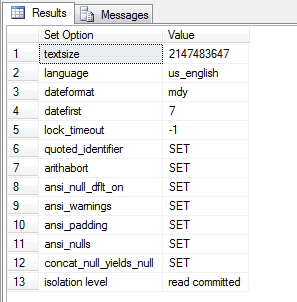
Ikkuna 1:

DBCC USEROPTIONS



Ikkuna 2:

DBCC USEROPTIONS



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

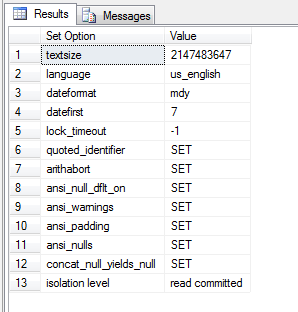
Sulje ikkunat ja avaa uudestaan, jotta Isolation level palautuu oletukseksi

Mikä on oletusarvoinen Isolation Level

Vastaus:

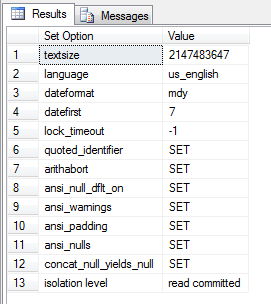
Ikkuna 1:

DBCC USEROPTIONS



Ikkuna 2:

DBCC USEROPTIONS



==================================================================

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 erityvyystasolla

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 erityvyystaso

====================================================

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 lopetteaan)

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

Yhtensa = 200

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

===================================================================