

4. önellenőrző teszt

1 kérdés

A tárolt eljárás, függvény és trigger közös vonása, hogy mindegyiket a CREATE utasítással hozzuk létre, és alkalmasak az üzleti logika megvalósítására. Létrehozáskor a kód nem fut le, hanem egy végrehajtási terv generálódik hozzá., ami szerveren tárolódik, mint az adatbázis egy objektuma.

- ☐ Igaz
- ☐ Hamis

2 kérdés

Mely állítások igazak az alábbi függvény esetén?

```
create function teszt1 (@x int) returns char(7) as
begin
    if @x > 0 return 'pozitív'
    else if @x < 0 return 'negatív'
    else return 'nulla'
end
```

- ☐ Mivel a függvény szintaktikailag hibás, azt a szerver nem hozza létre. (x)
- ☐ Mivel a függvény szintaktikailag hibátlan, ezért létrejön. Ezt követően meghívni például a select dbo.teszt1(56) utasítással lehet.
- ☐ Mivel a függvény szintaktikailag hibátlan, ezért létrejön. Ezt követően meghívni például az exec dbo.teszt1 56 utasítással lehet.
- ☐ Mivel a függvény szintaktikailag hibátlan, ezért létrejön. Ezt követően meghívni például a select * from dbo.teszt1(56) utasítással lehet.

3 kérdés

Melyik zártípusra jellemző az alábbi állítás?

1. ideiglenes zár, melyet az adatmódosításra készülő tranzakció a szűrőfeltétel kiértékelésének idejére szerez meg:
2. a SELECT-hez használt olvasási zár:
3. adatmódosító SQL utasításokhoz használt zár:

4 kérdés

Debug céljára a [triggererek](#) tartalmazhatnak PRINT utasításokat, azonban visszatérési értékük nincs, ahogy bemeneti paramétereik sincsenek.

4 kérdés Válasz

- Igaz
- Hamis

5 kérdés

Melyik izolációs szintre jellemző az állítás?

- az S zárok az egész tartományra a tranzakció végéig megmaradnak, ezért más tranzakciók nem tudnak a tartományba rekordot beszúrni/törölni - véd a fantomolvasás ellen
- az S zár a tranzakció végéig megmarad, és blokkolja más tranzakciók módosító utasításait

6 kérdés

Tegyük fel, hogy fut az alábbi tranzakció:

```
set transaction isolation level read committed
begin tran
update Employees set Salary = 1000 where EmployeeID = 1
--rollback tran
```

Egy másik tranzakcióban pedig az alábbi select utasítás kerül kiadásra:

```
select * from Employees where FirstName = 'Andrew'
```

Mi lesz az eredmény?

- Az 1. tranzakció X zárat helyez el a tábla első rekordjára, és amíg az commit-tal vagy rollback-kel véget nem ér, addig a zár megmarad a rekordon, így a másik tranzakció várakozni kényszerül, mert minden rekordot meg kellene vizsgálnia.
- Az 1. tranzakció nem befolyásolja a másik működését, így a select lefut, és visszaadja Andrew eredeti fizetését (100\$-t).

- Az 1. tranzakció nem befolyásolja a másik működését, így a select lefut, és visszaadja Andrew módosított fizetését (1000\$-t).
- Az 1. tranzakció S zárat helyez el a tábla első rekordjára, és amíg az commit-tal vagy rollback-kel véget nem ér, addig a zár megmarad a rekordon, így a másik tranzakció várakozni kényszerül, mert minden rekordot meg kellene vizsgálnia.

7 kérdés

Az update utasítás ... tranzakciónak tekinthető.

- explicit
- implicit

8 kérdés

Mely utasítások használhatóak egy explicit tranzakció létrehozásához?

- BEGIN TRANSACTION
- COMMIT TRANSACTION
- ROLLBACK TRANSACTION
- SAVE TRANSACTION
- END TRANSACTION
- START TRANSACTION
- STOP TRANSACTION

9 kérdés

Melyik állítás igaz?

A tranzakció futtatása során alapvető elvárás, hogy ...

- ... az adatbázis folyamatosan konzisztens állapotban maradjon.
- ... az egymással párhuzamosan futó tranzakciók a lehető legkevésebbé legyenek hatással egymásra.
- ... a sikeresen lefutott tranzakciók hatása hardver hiba esetén is megőrződjön.
- ... a programozási vagy egyéb hibák inkonzisztens állapotot idézzenek elő.
- ... a tranzakció minden lépése még hiba esetén is sikeresen lefusson.
- ... az egymással párhuzamosan futó tranzakciók 100%-ban legyenek hatással egymásra.

10 kérdés

Magasabb izolációs szinten

- a plusz adminisztráció miatt, és mert a hosszú élettartamú zárokra többet várnak a tranzakciók, gyengül a rendszer teljesítménye.
- csökken a logikai hibák előfordulásának veszélye.
- a kevesebb adminisztráció miatt, és mert a hosszú élettartamú zárokra többet várnak a tranzakciók, javul a rendszer teljesítménye.
- a plusz adminisztráció miatt, bár a hosszú élettartamú zárokra is kevesebbet várnak a tranzakciók, gyengül a rendszer teljesítménye.
- nő a logikai hibák előfordulásának veszélye.

11 kérdés

Válassza ki az állításnak megfelelő fogalmat!

(lehetseges valaszok: felhasznalo által definialt függvény, tarolt eljárás)

- Tartalmazhat olyan kódot, amely módosítja a tábla tartalmát. _____
- Meghívhat egy tárolt eljárást, vagy függvényt. _____
- Nem tartalmazhat TRY/CATCH blokkot. _____
- Típusai a skaláris és táblaértékű. _____
- Futtatásához az EXECUTE parancsot használjuk. _____
- Nem tartalmazhat olyan függvényhívást, amely nem determinisztikus, pl. RAND, GETDATE. _____
- Ott használható, ahol a szerver valamilyen kifejezést vár pl. SELECT-ben, CONSTRAINT-ben. _____

12 kérdés

Adja meg a függvények kategóriáját!

(lehetseges valaszokstring kezelo fgv, datum kezelo, analitikus, matematikai, logikai, egyeb)

LEN:

ISDATE:

@@IDENTITY:

GETDATE:

LAG:

RAND:

CAST:

IIF:

ERROR_PROCEDURE:

MAX() OVER:

13 kérdés

Adott az alábbi függvény:

```
create function teszt1(@x money) returns table
as return (select employeeid, lastname, salary from Employees where salary <= @x)
```

Melyik utasítás helyes?

- ☐ select * from dbo.teszt1(100) (x)
- ☐ select dbo.teszt1(100)
- ☐ select * from employees where dbo.teszt1(100)
- ☐ select teszt1(100) from employees
- ☐ select * from teszt1(150) where salary > 100 (x)
- ☐ A válaszok között nem szerepel helyes megoldás.

14 kérdés

Melyik igaz?

- ☐ A trigger meghívhat más tárolt eljárásokat és függvényeket, illetve az általuk végzett műveleteket kiválthatják más [triggerek](#) is, sőt rekurzívan a saját maga működését is.
- ☐ Egy táblán több trigger is lehet, akár azonos típusú is.
- ☐ Egy táblán csak egyféle trigger lehet.
- ☐ Egy táblán több trigger is lehet, de csak különböző típusúak.

15 kérdés

Mely tulajdonságokat kell a tranzakciónak teljesítenie?

- ☐ kohézió
- ☐ explicitás
- ☐ atomicitás

- elszigetelés (x)

16 kérdés

Mely állítások igazak az alábbi eljárás kapcsán?

```
create proc teszteljaras (@s money) as
begin try
    select *
    from employees
    where salary > @s
    order by lastname
end try
begin catch
    select error_number() as 'E_Number', error_message() as 'E_Message'
end catch
```

- Az eljárás szintaktikailag hibátlan, így létrejön.
- Meghívása: *exec teszteljaras 1000* utasítással lehetséges.
- Meghívása: *teszteljaras 1000* utasítással lehetséges.
- Az eljárás szintaktikailag hibás, így nem jön létre. Try/catch nem használható az eljárásban.
- Az eljárás nem tartalmazhat önmagában select utasítást, így az nem jön létre.
- Meghívása: *select teszteljaras 1000* utasítással lehetséges.
- Meghívása: *exec teszteljaras (1000)* utasítással lehetséges.

17 kérdés

Mely állítások igazak?

- A [triggererek](#) olyan speciális tárolt eljárások, amelyeket egy előre megadott feltétel bekövetkezése esetén a rendszer automatikusa végrehajt.
- Egy trigger akár a saját maga működését is kiválthatja.
- Ha egy triggerben hiba történik, akkor a trigger kiváltó SQL utasítás lefut.
- A [triggererek](#) által az aktív adatbázis passzív válik.
- A DDL-trigger egy táblához van csatolva, és valamely adatmódosító utasítás (UPDATE, DELETE, INSERT) hatására fut le.
- A [triggererek](#) egyik tipikus alkalmazási területe az automatizált üzleti workflow megvalósításának támogatása.

- Az SQL serveren a trigger futás közben két speciális virtuális táblában fér hozzá a működését kiváltó utasítás által módosított adatokhoz, mely virtuális táblák szerkezete eltér az eredeti táblától.

18 kérdés

Állítsa össze azt a függvényt, amely paraméterként kapja egy téglalap két oldalának a hosszát, és kiszámolja a téglalap területét.

CREATE	FUNCTION	terulet	(@a FLOAT	,	@b FLOAT)	RETURNS
Float								
AS	BEGIN							
	RETURN	@a*@b						
END								

Adja meg azt az utasítást, amellyel a függvényt meg lehet hívni az alábbi paraméterekkel.

```
declare @a float, @b float
set @a = 43
set @b = 79
```

	@a as 'Szélesség', @b as 'Magasság',					
	as 'Terület'					

terulet	dbo.terulet	,	a FLOAT	RETURNS	END TRY
RUN	(PROCEDURE	@b	SELECT)
@a	BEGIN TRY	b FLOAT	FUNCTIONS	RETURN	