

Quiz navigation



Show one page at a time

Finish review

Started on	Tuesday, 3 November 2020, 8:32 AM
State	Finished
Completed on	Tuesday, 3 November 2020, 9:12 AM
Time taken	39 mins 46 secs
Grade	15.00 out of 20.00 (75%)

Information
Flag question

A teszt megkezdésével elfogadom a következőket:

- A kérdéseket saját tudásom alapján válaszolom meg, nem használom az előadás anyagokat, saját jegyzeteimet, videókat, internetes forrásokat, és semmilyen meg nem engedett segédeszközt.
- A kérdéseket nem beszélem meg hallgató társaimmal a teszt ideje alatt, függetlenül attól, hogy mikor fejezem be a tesztet.
- A hallgató társaimtól és senki mástól nem kérek segítséget a teszt ideje alatt és nem nyújtok segítséget senkinek.
- A teszt kérdéseket és az értékelés során kapott válaszokat nem mentem el semmilyen formában.
- Betartom a BME etikai kódexében foglaltakat.

Question 1
Complete
Mark 3.00 out of 8.00
Flag question

Adott az *Address* adatbázis tábla Microsoft SQL Server platformon *City*, *Street*, *ZIP* oszlopokkal. Készíts triggerrel az irányítószám (ZIP) ellenőrzésére! Új sorok beszúrásakor és meglévő sorok módosításakor ellenőrizd, hogy a *ZIP* érték érvényes-e: 4 számjegyűnek kell lennie (szám típusú oszlop!) és 1-essel csak Budapest város (*City*) esetén kezdődhet. Ha bármely beszúrás/módosítás nem felel meg a követelményeknek, szakítsd meg a műveletet!

```
create trigger ValidZIP on Address
for insert, update as
Select zip
from Address
if(len(zip) != 4)
    rollback
return
else
    if(city == Budapest And zip[0] != 1)
        rollback
return
```

Comment:

Nem az insert táblán dolgozik.

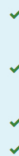
A módosítás több rekordot is érinthet, ezt egyáltalán nem veszi figyelembe.

Question 2
Correct
Mark 4.00 out of 4.00
Flag question

Az alábbi JSON dokumentumban szintaktikai hibák vannak. Jelöld az összes sort, amelyben van bármilyen szintaktikai hiba.

```
1 "version": 1.0
2 {
3   "name": Adam,
4   "age": 19,
5   # all hobbies as a list
6   "hobbies": { "football", "card games" },
7   "address" : {
8     "city": "Budapest"
9 }
```

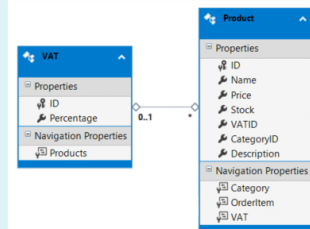
- ☒ 1
☐ 2
☒ 3
☐ 4
☒ 5
☒ 6
☐ 7
☐ 8
☐ 9



Your answer is correct.

Question 3
Complete
Mark 8.00 out of 8.00
Flag question

Adott az alábbi adatbázis modell és annak Entity Framework leképzése az *AdatvezEntities* nevű DbContext osztályban. Írj C# kódot Linq to Entities-t használva, amely töröl minden *Product* rekordot, amelyből kevesebb, mint 3 darab van raktáron (*InStock*). Ezek után törölj minden *VAT* rekordot is, amelyhez ezek után nem tartozik termék (*Product*). A DbContext létrehozása és kezelése a kód felelőssége. Használd a navigation property-ket, ahol lehetséges!



```
using(var db = new entitis)
{
    var result = db.products.where(p => p.InStock < 3).toList();
    foreach(p in result)
```

```
{
    db.Products.Remove(p);
}

var result = db.Vat.Join(product, p => p.VatId, v => v.Id).Where(v => v.Product.Count == 0).ToList();
foreach(p in result)
{
    db.Vat.Remove(p);
}

db.SaveChanges();
}
```

Comment:

Finish review