

Adatvezérelt rendszerek - BMEVIAUAC01 2020/21/1

Quiz navigation



Show one page at a time

Finish review

Started on	Tuesday, 3 November 2020, 8:42 AM
State	Finished
Completed on	Tuesday, 3 November 2020, 9:19 AM
Time taken	37 mins 26 secs
Grade	18.00 out of 20.00 (90%)

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 másától 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.

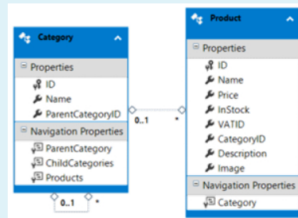
Question 1

Complete

Mark 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, amely nincs raktáron (*InStock=0*). Ezek után törölj minden *Category* rekordot is, amely üresen maradt. A DbContext létrehozása és kezelése a kód felelőssége. Használd a navigation property-ket, ahol lehetséges!



```
using (var context = new AdatvezEntities())
{
    var products = context.Product
        .Where( p => p.InStock == 0)
        .ToList();

    foreach ( var p in products)
    {
        context.Product.Remove(p);
    }

    context.SaveChanges();

    var categories = context.Category
        .Union( c => c.Product.Select( p => p.Category))
        .Where( c => c.Product.Count() == 0)
        .ToList();

    foreach ( var c in categories)
    {
        context.Category.Remove(c);
    }

    context.SaveChanges();
}
```

Comment:

Question 2

Complete

Mark 6.00 out of 8.00

Flag question

Adott a *CustomerSite* adatbázis tábla Microsoft SQL Server platformon *Country*, *City*, *Street*, *ZIP* oszlopokkal. Készíts trigger az irányítószám (ZIP) ellenőrzésére! Új sorok beszúrásakor és meglevő sorok módosításakor ellenőrizd, hogy a *ZIP* érték érvényes-e: ha a *Country* értéke "Hungary", akkor 4 számjegyűnek kell lennie (szám típusú oszlop!); bármely más ország esetén tetszőleges értékű lehet. Ha bármely beszúrás/módosítás nem felel meg a követelményeknek, szakítsd meg a műveletet!

```
CREATE TRIGGER ZipCheck
ON CustomerSite
FOR insert, update
AS
IF(UPDATE(ZIP))
BEGIN
    SELECT ZIP
    FROM CustomerSite c join inserted i ON c.ZIP = i.ZIP
```

```
WHERE CustomerSite.County = "Hungary"

IF(LEN(ZIP) != 4)

BEGIN

    ROLLBACK

    RETURN

END

END
```

Comment:
if nem az igazi

Question **3**

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.

Finish review