

.NET FEJLESZTÉS C# NYELVEN

4. forduló

RENDELKEZÉSRE ÁLLÓ IDŐ:

30:00

Ismertető a feladathoz

Fontos információk

A forduló után a megoldások publikálásával együtt iránymutatásként elérhetőek lesznek a **helyezéssel kapcsolatos információk**, látni fogod, hogy a kategóriában a játékosok 20%, 40% vagy 60%-a közé tartozol-e épp.

Felhívjuk figyelmedet, hogy a következő, **5. fordulótól az egyes kategóriák csak a kijelölt napokon lesznek megoldhatóak 7-22 óra között**, érdemes letöltened a naptárat a [Kategóriáim](#) menüpontban.

Negyedik forduló

Sikeresen elkerültétek a meteorzáport. A következő lépés az alap rendszerek működésre bírásra.

Ez nehezebb, mint elsőre tűnt, mivel az adatvesztés miatt több kódrészletet is pótolnod kell, ebben segítségére lesz Viki, viszont mint minden Juniornak, neki is segítened kell az összetettebb problémáknál.

Felhasznált idő: 00:17/30:00

Elért pontszám: 0/11

1. feladat 0/2 pont

Viki teljesítménykritikus kódot szeretne futtatni és fél, hogy a GarbageCollector tisztítása pont a kód futása során fog lefutni. Melyik metódust kell meghívnia az kritikus kód előtt?

Válasz

- ☐ GC.RemoveMemoryPressure()
- ☐ GC.SuppressFinalize()
- ☒ GC.Collect()
- ☐ GC.WaitForPendingFinalizers()

Magyarázat

A GC.Collect() meghívásával az összes lehetséges memória felszabadításra kerül. Ez nem akadályozza meg, hogy a GarbageCollector tisztítása pont a kód futása során fog lefutni, viszont nagyban csökkenti azt.

2. feladat 0/3 pont

Melyiket használnád érzékeny adatok szerializálásához bináris formátumba? (Többet is megjelölhetsz)

Válaszok

- ☐ XmlSerializer
- ☒ ISerializable
- ☐ DataContractSerializer
- ☒ BinaryFormatter

Magyarázat

Az XmlSerializer kiemelte XML szöveg, nem bináris formátum.

A DataContractSerializer XML szerializáláskor a WCF-et használja.

3. feladat 0/6 pont

Az EnumExercise meghívásakor melyik két változónak lesz azonos értékre?



```
namespace App
{
    [Flags]
    public enum Days1 : ulong
    {
        M = 0x1, Tu = 0x2, W = 0x4, Th = 0x8,
        F = 10, Sa = 20, Su = 40, Wnd = Sa | F
    }

    [Flags]
    public enum Days2 : ulong
    {
        M = 0x1, Tu = 0x2, W = 0x4, Th = 0x8,
        F = 0x10, Sa = 0x20, Su = 0x40, Wnd = Sa | F
    }

    [Flags]
    public enum Days3 : ulong
    {
        M = 1, Tu = 2, W = 4, Th = 8,
        F = 10, Sa = 20, Su = 40, Wnd = Sa | F
    }

    public enum Days4 : ulong
    {
        M = 1, Tu = 2, W = 4, Th = 8,
        F = 10, Sa = 20, Su = 40, Wnd = Sa | F
    }

    public class Enum
    {
        public void EnumExcercise()
        {
            var t1 =(Days1.M | Days1.W | Days1.F | Days1.Wnd).ToString();
            var t2 = (Days2.M | Days2.W | Days2.F | Days2.Wnd).ToString();
            var t3 = (Days3.M | Days3.W | Days3.F | Days3.Wnd).ToString();
            var t4 = (Days4.M | Days4.W | Days4.F | Days4.Wnd).ToString();
        }
    }
}
```

Válasz

- ☐ t1 és t2
- ☒ t1 és t3
- ☐ t3 és t4
- ☐ t2 és t3
- ☐ t2 és t4
- ☐ t3 és t4

Magyarázat

A t1 értéke "M, Wnd"

A t2 értéke "M, W, Wnd"

A t3 értéke "M, Wnd"

A t4 értéke "31"

[Legfontosabb tudnivalók](#)

[Kapcsolat](#)

[Versenyszabályzat](#)

[Adatvédelem](#)

© 2022 Human Priority Kft.

KÉSZÍTETTE

Megjelenés

 Világos 