

# .NET FEJLESZTÉS C# NYELVEN

2. forduló



A kategória támogatója: DXC Technology

## Ismertető a feladathoz

### Útmutató:

- A **radio button-os kérdésekre** egy helyes válasz van.
- Ha lejár a feladatlap ideje, a rendszer **AUTOMATIKUSAN** beküldi azt az addig megjelölt válaszokkal.
- Az **adatbekérős feladatokra NEM jár részpontszám**, csak a feleletválasztósakra.
- **Badge-eket** a 4.forduló után kapsz majd először.
- Az **adatbekérős kérdéseknél** igyekeztünk minden variációt megadni (kisbetű, nagybetű, szóköz), de ha mégis eltérést tapasztalsz a megoldásokban, kérjük, jelezd felénk!

+1: Azért szólunk, hogy senkit ne a végén érjen meglepetés: a játék nem tipp-mix és csapatkategória sincs! Természetesen akinek nem inge...

Jó versenyzést kívánunk!

Felhasznált idő: 00:00/10:00

Elért pontszám: 0/15

## 1. feladat 0/5 pont

Mely sorban / sorokban található hiba?

Több sor esetén növekvő sorrendben, szóköz nélkül, vesszővel elválasztva add meg (pl.: 1,3)!

```
1 var name = "John";
2 var isJohn = name === "John";
3 var isJohn2 = name == "John";
4 var isJohn3 = name is "John";
5 var isJohn4 = name is string;
6
7 var lambda = (bool b) => b ? 1 : 2;
8 object lambda2 = (bool b) => b ? 1 : 2;
9 var lambda3 = (bool b) => b ? 1 : "2";
```

### Válaszok

A helyes válasz:

2,9

2,7,8,9

2,4,9

### Magyarázat

C# 8.0 verziónál a 2,4,9, C# 9.0 verziónál a 2,7,8,9, C# 10.0-nál viszont 2,9, a helyes válasz, így mind a három választ elfogadjuk.

## 2. feladat 0/10 pont

Válaszd ki az **igaz** állításokat!

### Válaszok

- ☐ A using kulcsszót csak IDisposable erőforrásra érdemes használni.
- ☒ A System.Collections.IEnumerator interfész azt jelöli egy adatszerkezetnél, hogy az iterator tervezési mintát megvalósítja.  
**Ez a válasz helyes, de nem jelölted meg.**
- ☐ Overriding-nak hívjuk, mikor azonos scope-ban több azonos nevű, de eltérő szignatúrájú metódus van.
- ☐ Overloading-nak hívjuk, mikor egy leszármazott osztály módosítja az ősz osztály egy metódusának viselkedését.
- ☒ A ref és az out függvény paraméter között a különbség, hogy a változó a függvény előtt vagy közben inicializálódik.  
**Ez a válasz helyes, de nem jelölted meg.**
- ☐ A System.Array.Clone() a prototype tervezési mintát valósítja meg.
- ☒ Reflection-el egy object private metódusa is meghívható.  
**Ez a válasz helyes, de nem jelölted meg.**
- ☐ dynamic változónak private metódusa is meghívható.
- ☐ C# 8.0 egyik újdonsága a default interface volt, amely segítségével már megoldható a többszörös öröklés is.

### Magyarázat

**Kedves Versenyzők!**

Nagyon sok észrevételt kaptunk a feladattal kapcsolatosan, ezeket a feladatírókkal egyeztetve az alábbi módosítások történtek:

A "using kulcsszót csak IDisposable erőforrásra érdemes használni" és az "A System.Array.Clone() a prototype tervezési mintát valósítja meg": helytelen válaszlehetőségek a korábbi helyesnek jelöltekkel ellentétben.

"A const változók implicit static változók" válaszlehetőség pedig törlésre került.

*A using kulcsszót csak IDisposable erőforrásra érdemes használni.*

Ez egy syntax sugar egy try-finally szerkezetre, ami meghívja a disposed.

A **const** változók *implicit static* változók.

Compile-time eldől az értéke, ezért úgy viselkedik, mint egy readonly static.

**Overriding**-nak hívjuk, mikor azonos scope-ban több azonos nevű, de eltérő szignatúrájú metódus van.

Ez az orverload.

**Overloading**-nak hívjuk, mikor egy leszármazott osztály módosítja az őssosztály egy metódusának viselkedését.

Ez az orverride.



[Legfontosabb tudnivalók](#)

[Kapcsolat](#)

[Versenyszabályzat](#)

[Adatvédelem](#)

© 2023 Human Priority Kft.

KÉSZÍTETTE  cone

Megjelenés



Világos

