

MOBILFEJLESZTÉS - IOS

4. forduló



A kategória támogatója: zenitech - Powered by
AutSoft

Ismertető a feladatlaphoz

Közeleg az 5. forduló, figyelj az időpontokra!

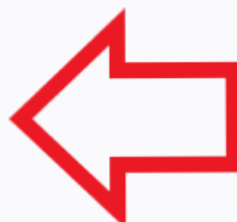
Használd a naptárat:

KATEGÓRIÁIM

Összesen 10 kategóriára jelentkezted



Versenynaptár letöltése



Vagy figyeld kategóriánként az időpontokat (íme egy MINTA, hol találod):

● 3. FORDULÓ

A lezárt fordulókban eddig megszerzett pontok:

0/100 pont

**BOSCH**

Invented for life

Fordulók

Forduló	Pontok, időtartam		Feladat megoldható	Státusz
7. forduló	23 pont 25:00		2023.11.28. 20:00-tól 2023.11.28. 20:35-ig	Feladatlap
6. forduló	23 pont 30:00		2023.11.21. 20:00-tól 2023.11.21. 20:40-ig	Feladatlap
5. forduló	28 pont 25:00		2023.11.14. 20:00-tól 2023.11.14. 20:35-ig	Feladatlap

Amennyiben olyan kategóriában játszol, ahol van csatolmány, de hibába ütközel a letöltésnél, ott valószínűleg a vírusirtó korlátoz, annak ideiglenes kikapcsolása megoldhatja a problémát. (Körülbelül minden 3000. letöltésnél fordul ez elő.)

Jó versenyzést kívánunk!

1. feladat 4 pont

Mi annak a libdispatch structnak a neve Swiftben, amely lehetővé teszi, hogy bizonyos kódrészleteket egyszerre ne lehessen több szálon elérni?

Válaszok

2. feladat 10 pont

Az alábbiak közül mely(ek) nem érvényes UIBezierPath metódus(ok) UIKit-ben?

Válasz

- ☐ `addLine(to:)`
- ☐ `addCurve(to:controlPoint1:controlPoint2:)`
- ☐ `addArc(withCenter:radius:startAngle:endAngle:clockwise:)`
- ☐ `addTriangle(withBase:height:)`

3. feladat 8 pont

SwiftUI-ben mi annak a property wrappernek a neve, amelyet arra használhatunk, hogy egy tárolt változót hozzunk létre, ami automatikusan érvényteleníti (invalidates) a nézetét, ha az érték megváltozik? (A válasznál fontos a @ karakter is, nem csak a property wrapper neve.)

Válasz

4. feladat 15 pont

Az alábbiak közül mely(ek) érvényes URLRequest cache policy(-k)?

Válaszok

- ☐ `.useProtocolCachePolicy`
- ☐ `.reloadIgnoringLocalCacheData`
- ☐ `.onlyLoadIfCached`
- ☐ `.returnCacheDataElseLoad`

5. feladat 10 pont

Mit csinál az alábbi kód?

```
let queue = DispatchQueue(label: "com.example.queue")
queue.async {
    print("Hello, World!")
}
```

Válasz

- ☐ A "Hello, World!" szöveget a fő szálon (main queue) írja ki.
- ☐ A "Hello, World!" szöveget egy soros szálon (serial queue) írja ki.
- ☐ A "Hello, World!" szöveget egy párhuzamos szálon (concurrent queue) írja ki.
- ☐ A "Hello, World!" szöveget egy háttérszálon (background queue) írja ki.

6. feladat 20 pont

Mi a különbség a map, flatMap, és compactMap között Swiftben?

Válasz

- ☐ map transzformálja a collection összes elemét majd eltávolítja a nil értékeket, flatMap átalakítja és aztán "kisimítja" (flattens) az eredményt, compactMap pedig csak átalakít minden elemet
- ☐ map átalakítja egy collection minden elemét, flatMap átalakítja és aztán "kisimítja" (flattens) az eredményt, compactMap pedig az átalakítás után eltávolítja a nil értékeket
- ☐ map átalakítja egy collection minden elemét és eltávolítja a nil értékeket, flatMap átalakítja a collection minden elemét, compactMap átalakítja és aztán "kisimítja" (flattens) az eredményt
- ☐ map átalakítja egy collection minden elemét és aztán "kisimítja" (flattens) az eredményt, flatMap az átalakítás után eltávolítja a nil értékeket, és a compactMap pedig átalakítja a collection minden elemét.

7. feladat 20 pont

Mi a különbség a == és a === között Swiftben?

Válasz

- ☐ == azt ellenőrzi, hogy két példány egyenlő-e, és === azt ellenőrzi, hogy két példány azonos-e
- ☐ == azt ellenőrzi, hogy két példány azonos-e, és === azt ellenőrzi, hogy két példány egyenlő-e

- ☐ == és === mindkettő azt ellenőrzi, hogy két példány egyenlő-e
- ☐ == és === mindkettő azt ellenőrzi, hogy két példány azonos-e

Megoldások beküldése