

# MOBILFEJLESZTÉS

4. forduló



A kategória támogatója: AutSoft Zrt.

## Ismertető a feladathoz

**A 4. forduló után elérhetőek lesznek a helyezések %-os formában: azaz kiderül, hogy a kategóriában a versenyzők TOP 20% - 40% -60% -ához tartozol-e!**

Szeretnénk rá felhívni figyelmedet, hogy a játék nem Forma-1-es verseny! Ha a gyorsaságod miatt kilököd a rendesen haladó versenyzőket, kizárást vonhat maga után!

Felhasznált idő: 15:00/15:00

Elért pontszám: 0/12

## 1. feladat 0/3 pont

Mi igaz a JobScheduler-re?

### Válaszok

- ☒ Lehetővé teszi az alkalmazás számára az adott feladat végrehajtását  
**Ez a válasz helyes, de nem jelölted meg.**
- ☐ Figyelembe veszi az eszköz memória állományát, az időzítés vezérlése árán
- ☒ Túléli az alkalmazások újraindítását  
**Ez a válasz helyes, de nem jelölted meg.**
- ☐ Ez az osztály hozzáférést biztosít a rendszer riasztási szolgáltatásaihoz.

### Magyarázat

A JobScheduler API műveletet hajt végre az alkalmazás számára, ha egy sor előre meghatározott feltétel teljesül. Lehetővé teszi az alkalmazás számára az adott feladat végrehajtását, miközben figyelembe veszi az eszköz akkumulátorát, az időzítés vezérlése árán. A JobManager megkönnyíti a feltöltések kezelését, mivel automatikusan kezeli a hálózat megbízhatatlanságát. Az alkalmazások újraindítását is túléli. A rendszer riasztási szolgáltatásaihoz hozzáférést az AlarmManager biztosítja.

## 2. feladat 0/1 pont

Kotlin Flow: Melyik **NEM** termináló operátor?

### Válasz

☐ reduce()

☐ toList()

☒ take()

Ez a válasz helyes, de nem jelölted meg.

☐ lastOrNull()

Ez a válasz helytelen, de megjelölted.

### Magyarázat

take(): Olyan flow-t ad vissza, amely tartalmazza az első count (a take() kötelező paramétere, pozitív egész) elemeket.

## 3. feladat 0/1 pont

Mit jelent, hogy a SwiftUI Reactive?

### Válasz

☐ Adott állapotra, a renderelt UI mindig ugyanolyan

☒ Amikor az állapot változik, a SwiftUI automatikusan frissíti a UI-t

Ez a válasz helyes, de nem jelölted meg.

☐ Nem implementáljuk, hanem deklaráljuk a UI-t

☐ Amikor csak Reactive értékek változnak, semmi nem változik

### Magyarázat

A Reactive azt jelenti, hogy a változásra reagálva a keretrendszer frissíti a UI-t.

## 4. feladat 0/3 pont

Milyen két protokollra alias a Swift Codable?

A dokumentációból vagy Swift API interfész leírásban szereplő formátumban várjuk a választ. A megoldáshoz csak a csillag karaktereket kell kiegészíteni. A sorrend nem számít, mindkét esetet elfogadjuk. A formázás a következő: ProtocolA & ProtocolB

Tehát az & jel előtt és után is pontosan 1 space van, és mindegyik nagybetűvel kezdődik.

```
public typealias Codable = ****
```

### Válaszok

A helyes válasz:

Decodable & Encodable

Encodable & Decodable

## Magyarázat

A Swift Codable egy type alias az Encodable és a Decodable protokollokra.

## 5. feladat 0/2 pont

Hogyan működik a Swift copy-on-write mechanizmusa?

### Válaszok

- ☐ Egy Class másolásakor a nyelv addig késlelteti az objektum tényleges másolását, ameddig nem történik módosítás, ezzel memóriát spórolva.
- ☒ Egy Struct másolásakor a nyelv addig késlelteti az objektum tényleges másolását, ameddig nem történik módosítás, ezzel memóriát spórolva.  
**Ez a válasz helyes, de nem jelölted meg.**
- ☒ Egy Array másolásakor a nyelv addig késlelteti az objektum tényleges másolását, ameddig nem történik módosítás, ezzel memóriát spórolva.  
**Ez a válasz helyes, de nem jelölted meg.**

## Magyarázat

Ha egyik változóból a másikba másolunk egy adatot, akkor a Swift első körben csak a referenciát másolja le. Amint az adaton változtatunk a második változóban, akkor másolja le újabb memóriaterületre az adatot. Ez azonban csak structokon működik, classoknál referencia átadás történik alpból. Az Array egy Struct, viszont nem csak az Array-re vonatkozik a copy-on-write mechanizmus, ezért a 3. pont nem jó válasz.

## 6. feladat 0/2 pont

Mi a baseline Android linterek használata esetén?


### Válasz

- ☐ Egy határérték, amelyet nem szabad átlépnie a lint által talált hibáknak
- ☐ Egy snapshot, amelyhez képest ellenőrzi a lint a változtatásokat
- ☐ Egy config, amelyben a felsorolt hibákat a lint figyelmen kívül hagyja
- ☒ Egy snapshot, amelyben a felsorolt hibákat a lint figyelmen kívül hagyja  
**Ez a válasz helyes, de nem jelölted meg.**

## Magyarázat

Az Android Lint egy statikus kódelemző eszköz, amely ellenőrzi az Android projekt forrás fájljait az esetleges hibákért, valamint a helyesség, a biztonság, a teljesítmény, a használhatóság és a hozzáférhetőség optimalizálási fejlesztéseit. A baseline egy snapshot, amely tartalmazza a projekt jelenlegi warning hibákat, amelyet egy baselineként lehet használni a jövőbeli ellenőrzésekhez, így csak az új problémák jelennek meg.



[Legfontosabb tudnivalók](#)  [Kapcsolat](#)  [Versenyszabályzat](#)  [Adatvédelem](#) 

© 2023 Human Priority Kft.

KÉSZÍTETTE  **cone**

Megjelenés

 Világos 