

JAVA SPRING MICROSERVICES

4. forduló



A kategória támogatója: Vodafone Magyarország
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!

A 151.-edik hiba, csak a mai listán... – Ray Arnold

Mr. Hammond egyszer már elsiette a park élesítését, viszont tudja, hogy a projekt jó kézben van. A beszélgetések alapján a vezetőség többször is kijelentette, hogy a minőség a legfontosabb számukra és tanultak a múlt hibáiból. Az elvárás adott, minél több és minél hatásosabb tesztekkel kell írnotok, hogy a minőség biztosított legyen.

Mire lesz szükséged a fordulóban?

- Java 11 telepítve
- Maven telepítve (vagy Maven Wrapper használata)
- Fejlesztői környezet (ajánlott)

Ha egy adott feladat egy annotációt vár válaszként, akkor oda csak az annotáció nevét add meg @ előjellel, nem szükséges teljes minősített nevet megadni, például: @NotNull.

Felhasznált idő: 00:00/30:00

Elért pontszám: 0/7

Indítás utáni csatlományok

1. feladat 0/1 pont

Rakd növekvő sorrendbe a tesztelési módszereket, mennyiség, költség és futási idő függvényében.

1. Integrációs tesztek
2. Komponens tesztek
3. Egység tesztek
4. End to end tesztek

Válasz

☐ 2,3,1,4

☒ 3,2,1,4

Ez a válasz helyes, de nem jelölted meg.

☐ 3,1,2,4

Magyarázat

A legtöbb fejlesztőnek rémlik a "teszt piramis" kifejezés amely a szoftverfejlesztések során előkerül, ebben a piramisban a következő sorrendet olvashatjuk (alulról felfelé): Egység tesztek, Komponens tesztek, Integrációs tesztek, End to end tesztek.

2. feladat 0/2 pont

Melyik annotációkkal állíthatjuk be a teszt futtatás során az alkalmazni kívánt profilokat Spring Boot használata esetén?

Válasz

☐ @ActivateProfiles

☒ @ActiveProfiles

Ez a válasz helyes, de nem jelölted meg.

☐ @ActivateProfile

☐ @ActiveProfile

Magyarázat

Tesztek futtatása során az @ActiveProfiles nevű annotációval tudjuk beállítani a kívánt profilt.

Hivatalos dokumentáció: <https://docs.spring.io/spring-framework/docs/current/reference/html/testing.html#spring-testing-annotation-activeprofiles>

3. feladat 0/0 pont

Szeretnénk egy bizonyos beant csak akkor a Springes kontextusba tölteni, ha a **test** profil aktív, milyen úton tehetjük ezt meg?

Válaszok

☒ A @Profile nevű annotációt kell az osztályra helyeznünk és a **test** | **it-test** értéket kell neki megadni.

Ez a válasz helyes, de nem jelölted meg.

☐ Fel kell annotálnunk a bean-t a @Component annotációval, és a **profile** nevű attribútumát be kell állítanunk **test**-re.

☒ A @Profile nevű annotációt kell az osztályra helyeznünk, és a **test** értéket kell neki megadni.

Ez a válasz helyes, de nem jelölted meg.

☐ Ebben az esetben az adott osztályt egy @Configuration osztályon belül kell példányosítani egy metóduson belül, amit annotálunk a @Bean annotációval, és a **value** attribútumába a **test** értéket írjuk.

Magyarázat

Profilokat a különböző beaneken a@Profilenevű annotációval tudunk tudjuk meghatározni, meg tudunk adni egyszerre többet is, de a kérdésben a "CSAK AKKOR" jelenléte miatt a harmadik válaszlehetőség helyes. Az első esetben a bean két profilban is elérhető lesz, de a feladat szerint, csak akkor kell ha a **test** aktív és semmi más.

4. feladat 0/1 pont

Csapatotok sikeresen bekötötte a CI folyamatba a szolgáltatások folyamatos építését és tesztelését. A tesztek többsége sikeresen lefut, viszont az utolsó commit óta valami nem klappol és közben kapjátok a hírt egy külső csapattól, hogy folyamatosan hibára futnak egy végpont hívása során. Sajnos a dokumentációk még nem érhetőek el a szolgáltatásról, a fejlesztési ciklus kicsit megcsúszott, de a külső csapat azonnali segítséget kér tőletek.

Az alkalmazás fordításához és a tesztek futtatásához Maven szükséges vagy fejlesztői környezet.

Maven esetén a kód főkönyvtárában a következő parancsot kell kiadni: **./mvnw test**

Melyik annotációt kell ráraknunk a `CheckinControllerTest` osztályra, hogy el tudjon indulni? (Kérlek a válaszodba írd be a @ jelet is!)

A 4.-6. feladatok megoldásához a letöltött állományt használd!

Válasz

A helyes válasz:

`@AutoConfigureMockMvc`

Magyarázat

Spring Boot egy beépített kontextus felismerési implementáció képes beaneket előre konfigurálni majd pedig azokat a fejlesztő számára elérhetővé tenni. Nem történik ez másképpen a teszteknel sem, a **@SpringBootTest** alkalmazása után a kontextus elérhető, már csak gondoskodni kell arról, hogy a használt beanek elérhetőek legyenek.

Nincs ez másképpen a `MockMvc` esetén, alpból nem érhető el, ahhoz, hogy elérhető legyen a teszt futtatása során a **@AutoConfigureMockMvc** annotációt kell ráraknunk.

5. feladat 0/1 pont

A végponton lévő POJO osztály valamiért a tesztelés során nem képes feloldani a kérésben bejövő mezőket. Milyen annotációval oldhatjuk fel ezt a problémát a kontroller metódus szignatúrájában? (Kérlek a válaszodba írd be a @ jelet is!)

Válasz

A helyes válasz:

`@RequestBody`

Magyarázat

A végponton metódusából hiányzik a **@RequestBody** nevű annotáció amely nélkül a **POST**-os kérések esetén a kérés törzse nem konvertálódik át a POJO objektumra.

6. feladat 0/2 pont

Hiába a validációs annotációk, mégsem működik a kérés ellenőrzése, mi lehet a probléma?

Válaszok

- ☐ Az application.properties fájból hiányzik egy konfigurációs paraméter, ami engedélyezi ezt a működést.
- ☒ Hiányzik egy függőség, amely aktiválja azt.
Ez a válasz helyes, de nem jelölted meg.
- ☐ Ilyesfajta tesztekkel nem tudjuk tesztelni a validációt, mert ebben az esetben egy olyan webes kontextus jön létre, amely nem tudja kezelni azokat.
- ☒ A kontroller metódus paraméterére rá kell tennünk a **@Valid** vagy **@Validated** annotációk egyikét.
Ez a válasz helyes, de nem jelölted meg.

Magyarázat

Az alkalmazásból hiányzik a **spring-boot-starter-validation** nevű függőség, és hiányzik a kontroller metódus definíciójában a **@Valid** vagy **@Validated** nevű annotációk egyike.



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

© 2023 Human Priority Kft.

KÉSZÍTETTE **cone**

Megjelenés

Világos