

TESZTAUTOMATIZÁLÁS

1. forduló



A kategória támogatója: EPAM

RENDELKEZÉSRE ÁLLÓ IDŐ:

30:00

Ismertető a feladathoz

Fontos információk:

A kérdésekre **mindig van helyes válasz!** Ha csak egy helyes válasz van az adott kérdésre, radio button-os választási lehetőségeket fogsz látni.

Kérjük, hogy a feladatok szövegeit **ne másold** és a böngésződ fejlesztő eszközét/ konzolját se nyisd meg feladatmegoldás közben! Mindkettő kizárást vonhat maga után.

Minden forduló után a **megoldások csütörtök reggel 8 órakor** lesznek elérhetőek.

A megoldásokkal kapcsolatos esetleges **észrevételeket a megoldások megjelenését követő kedd éjfélig** várjuk.

A több válaszlehetőségű feleletválasztós kérdéseknél járnak **részpontszámok, ha egyik rossz választ sem jelölöd be.**

Ha kifutsz az adott feladatlap kitöltésére rendelkezésre álló időből, a felület **automatikusan megpróbálja beküldeni** az addig megadott válaszokat.

Minden feladatsornak van egy **becsült minimum megoldási ideje** (legalább a feladat elolvasási ideje). Aki ennél rövidebb idő alatt oldja meg, kizárható a versenyből.

Az első kategória után, amelynek a feladatlapját megoldod a fordulóban, kapni fogsz egy 2-3 perc alatt kitölthető **kérdőívet**. Az ezekből összeállított piackutatás legfontosabb eredményeit a díjátadót követően Veled is megosztjuk majd. Formáljuk közösen a piacot!

Felhasznált idő: 01:31/30:00

Elért pontszám: 0/18

1. feladat 0/2 pont

Melyik eszköz segítségével lehet böngészőn kívül JavaScript kódot futtatni?

Válasz

- ☐ DirectX 9.0c
- ☒ Node.js
- ☐ React.js
- ☐ Playwright.js

Magyarázat

Böngészőn kívüli JS futtatásra Node.js-t használunk.

A felsoroltak közül:

DirectX 9.0c - egy könyvtár multimédiás tartalmak, alkalmazások megjelenítésére.

React.js - egy frontend könyvtár JSben, ami a böngészőben fut.

Playwright.js - tesztelői framework.

2. feladat 0/2 pont

Válaszd ki a primitív TypeScript típust (típusokat)!

Válaszok

- ☒ string
- ☒ number
- ☐ Person
- ☐ EventEmitter

Magyarázat

EventEmitter - egy NodeJS osztály.

Person - egy saját típus, a TypeScript-ben nem beépített típus.

string, number - alap primitív típusok a TypeScriptben.

3. feladat 0/2 pont

Adott az alábbi Python kód:

```
for number in range(5, 55, 4):  
    print(number)
```

Mi lesz a számsorozat utolsó 3 kiírt eleme? Milyen feladatot lát el a 3. paraméter a range függvényben?

Válasz

- ☐ 1. 44,49,54
2. A kezdeti értéket határozza meg
- ☐ 1. 44,49,54
2. A növelési értéket határozza meg
- ☐ 1. 45,49,53
2. A kezdeti értéket határozza meg
- ☒ 1. 45,49,53
2. A növelési értéket határozza meg

Magyarázat

A range() esetén a paraméterek a következőképpen működnek: az első paraméter a sorozat első elemét, a második paraméter a sorozat utolsó elemét, a harmadik paraméter pedig az inkrementálás mértékét határozza meg, így a range-ben kiírt elemek:

- 1. 9
- 2. 13
- 3. 17
- 4. 21
- 5. 25
- 6. 29
- 7. 33
- 8. 37
- 9. 41
- 10. 45
- 11. 49
- 12. 53

4. feladat 0/2 pont

Adott az alábbi Python kód:

```
class Element:  
    def __init__():  
        self.selector_type = selector_type  
        self.selector = selector
```

Az alábbiak közül melyik/melyek helyes definíciók az `__init__` paraméter listájára, hogy sikeres legyen az objektum létrehozása?

Válaszok



```
def __init__(self, selector_type, selector):
```



```
def __init__(selector_type, selector):
```



```
def __init__(selector_type, selector, self):
```



```
def __init__(self, selector, selector_type):
```

Magyarázat

Python class létrehozásánál használhatjuk az '`__init__()`' függvényt, ami mindig lefut, amikor egy új objektum inicializálódik. A '`self`' paraméter egy referencia az aktuális objektumra, amivel hivatkozhatunk az aktuális objektum változóira, ennek mindig az első paraméternek kell lennie, az utána lévő paraméterek sorrendje tetszőleges ebben az esetben.

5. feladat 0/2 pont

Az alábbiak közül mely környezeti változókat kell beállítanunk a Windows számítógépünkön, hogy Java alkalmazásokat tudjunk futtatni?

Válaszok



PATH



JDKPATH



JAVA_HOME



WINDOWS_HOME

Magyarázat

A `JAVA_HOME` környezeti változót be kell állítanunk, hogy a számítógépre telepített JDK elérési útját tartalmazza.

6. feladat 0/2 pont

Az alábbiak közül melyek valós különbségek az interface és az absztrakt osztály között Javában?

1. Az absztrakt osztály nem támogatja a többszörös öröklődést, az interface igen.
2. Az absztrakt osztályt az "extends" kulcsszóval terjesztjük ki, míg az interface-t az "implements" kulcsszóval.
3. Az absztrakt osztályt nem lehet kiterjeszteni, az interface-t igen.
4. Az absztrakt osztály csak változókat tartalmazhat, az interface nem.

Válasz

- ☒ 1) és 2)
- ☐ 1), 2) és 3)
- ☐ 1), 2), 3) és 4)
- ☐ 3) és 4)

Magyarázat

Az absztrakt osztályt ki lehet terjeszteni, az "extends" kulcsszóval.

A többszörös öröklődést nem támogatja.

Az absztrakt osztály tartalmazhat változókat, absztrakt és nem absztrakt metódusokat is.

Az interface-t az "implements" kifejezéssel implementáljuk. Támogatja a többszörös leszármaztatást.

7. feladat 0/4 pont

Mely reguláris kifejezések illeszkednek teljesen a következő e-mail címre? ITMegmerettetes@hungary.hu

Válaszok



`^[a-zA-Z]+@[a-z]+\.[a-z]{2}$`



`[a-z]+@[a-z]+\.[a-z]{2}`



`^\w+@\w+\.\w{2}$`



`^\w+@\w+\.\w+?$`

Magyarázat

1. A második reguláris kifejezésre csak kisbetűvel kezdődő email címek illeszkednek, ezért ez nem lesz jó.
2. A negyedik reguláris kifejezés végén a pont karakter után 0 vagy 1 betűt vár, ezért a hu nem fog illeszkedni rá.
3. Az első reguláris kifejezésre illeszkedni fog az email cím, mivel az 1 vagy több kis és nagybetűvel kezdődő email címet vár majd egy @ karaktert ezután pedig 1 vagy több kisbetűvel folytatódik és egy . karakterrel, majd a végén 2 kisbetűt tartalmaz.
4. A harmadik válasz is helyes. Az 1 vagy több betűt vár kezdésnek majd @ karaktert és 1 vagy több betűt ismét utána pedig egy . karaktert. A végén pedig 2 betűt vár.

8. feladat 0/1 pont

Az alábbiak közül melyik **nem** a tesztelés célja?

Válasz

- ☐ Annak igazolása, hogy a szoftver az elvárásoknak megfelelően működik.
- ☐ Kód lefedettség növelése.
- ☐ Működés során előforduló hibák esélyének csökkentése.
- ☒ A szoftver kitelepítése.

Magyarázat

A megadott lehetőségek közül a szoftver kitelepítése nem lehet a tesztelés célja, mivel ez operációs feladat.

9. feladat 0/1 pont

Az alábbi tevékenységek közül melyik nem a teszt végrehajtási folyamat része?

Válasz

- ☐ Az aktuális és elvárt eredmények összehasonlítása.
- ☐ A tesztelés során talált hibák jelentése.
- ☒ A tesztek végrehajtásához szükséges tesztadatok előkészítése.

☐ A tesztek manuális vagy automatizált végrehajtása.

Magyarázat

A helyes válasz a tesztek végrehajtásához szükséges tesztadatok előkészítése, mivel az a teszt megvalósítási folyamat része.

[Legfontosabb tudnivalók](#)

[Kapcsolat](#)

[Versenyszabályzat](#)

[Adatvédelem](#)

© 2022 Human Priority Kft.

KÉSZÍTETTE

Megjelenés

 Világos 