

# TESZTAUTOMATIZÁLÁS

5. forduló



A kategória támogatója: EPAM

RENDELKEZÉSRE ÁLLÓ IDŐ:

35:00

## Ismertető a feladathoz

Felhasznált idő: 01:32/35:00

Elért pontszám: 0/40

### 1. feladat 0/10 pont

Adott a következő DOM struktúra.

Mely XPATH selectorokkal tudjuk kiválasztani csak azokat a "dossier"-eket amelyekben egy valid hiperlink (nem helyfoglaló / placeholder, egy tényleges URL-re mutat) van ami petrol típusú.

```
<ul>

  <li element = "dossier">
    <a class = "label" type = "diesel car" href = "http://www.test.hu/toyota.jpg">Yaris</a>
  </li>
  <li element = "dossier">
    <a class = "label" type = "petrol car" href = "http://www.test.hu/BMW.jpg">Z4</a>
  </li>
  <li element = "dossier">
    <a class = "label" type = "diesel car">Astra</a>
  </li>
  <li element = "dossier">
    <a class = "label" type = "petrol car" href = "http://www.test.hu/ford.jpg">Focus</a>
  </li>
  <li element = "dossier">
    <a class = "label" type = "petrol car">Yaris</a>
  </li>
  <li element = "dossier">
```

```
<a class = "label" type = "diesel car" href = "http://www.test.hu/Renault.jpg">Clio</a>
</li>
</ul>
```

## Válasz

☐

//li[@element="dossier"]//a[contains(@type, "petrol")][contains(@href, "")]

☒

//li[@element="dossier"][./a[contains(@type, "petrol")][contains(@href, "http")]]

☐

//li[@element="dossier"]//a[contains(@type="petrol")][contains(@href="")]

☐

//li[@element="dossier"][./a[(@type="petrol")][(@href="http://")]]

## Magyarázat

Az első válaszban lévő selector az span elemet fog meg, ezért nem jó.

A harmadik válaszban lévő selectornak helytelen a szintaktikája, a contains funkció vesszővel elválasztva várja az attribútumot és tartalmát.

A negyedik válaszban levő @type="petrol" túl szigorú vizsgálat, ez csak a konkrétan petrol-t tartalmazó type attribútumokat fogja meg.

A második válasz a helyes, mivel az olyan li elemeket fogja meg, amelyeknek van dossier értékkel rendelkező element attribútumuk, és van egy span gyerekük, amelynek a type attribútuma tartalmazza a petrol szót és van href attribútuma.

## 2. feladat 0/10 pont

Tegyük fel, hogy egy webshop fejlesztésén dolgozik egy csapat. A kódbázisban van belső logika az áru és az akciók kezelésére, van egy adatbázis, és van egy webinterfész. A webinterfész a logikával egy REST API-n keresztül kommunikál.

Egy konvencionális tesztelői piramis szerint, relatív hány API/Service tesztre van szükség Unit/Komponens és E2E/UI tesztekhez képest?

## Válasz

☐

Ugyanannyi API teszt, mint Unit és E2E teszt

☒

Több API teszt, mint E2E teszt, de kevesebb API teszt, mint Unit teszt

☐

API tesztek közül kell legyen a legtöbb

☐ API tesztekből kell legyen a legkevesebb

## Magyarázat

A tradicionális tesztelői piramis alapján a legtöbb kell, hogy legyen a unit/komponens tesztek közül, és legkevesebb az end-to-end/ui tesztek közül. Az API/Service tesztek pedig a piramis közepén helyezkednek el.

### 3. feladat 0/10 pont

Melyik reguláris kifejezés használható, ha azt szeretnénk, hogy az a következő kifejezések mindegyikére illeszkedjen?

Szoftvertesztelés

Manuális szoftvertesztelés

Automata szoftvertesztelés

Szoftver-Tesztelés

#### Válasz



`(Manuális |Automata )?(S|s)zoftver(-)?(T|t)esztelés`



`[Manuális |Automata ]?[S|s]zoftver[-]?[|t]esztelés`



`(?:Manuális |Automata)(S|s)zoftver(?:-)(T|t)esztelés`



`(?!Manuális |Automata)(S|s)zoftver(?:-)(T|t)esztelés`

## Magyarázat

Csak az első válasz helyes.

A második helytelen, mivel itt a karakter-osztályok nem megfelelőek.

A harmadik szintén helytelen, mivel ebben az esetben egyik részkifejezés sem opcionális.

A negyedik pedig azért helytelen, mivel itt a nem-illeszkedő csoportok miatt (?!), csak azokra fog illeszkedni a kifejezés, ahol ezek hiányoznak.

## 4. feladat 0/10 pont

Tegyük fel, hogy egy mozi online jegyeladási felületének tesztelésére kell automata teszteseteket írnod. Az alkalmazásnak az alábbi követelményeknek kell megfelelnie:

a felhasználó kiválaszthat filmet és egy hozzátartozó vetítési időpontot,  
ezt követően megadhatja, hogy hány jegyet szeretne vásárolni,  
majd a megadott darabszámmal megegyező mennyiségű helyet kell lefoglalnia,  
ezt követően jut el a vásárlási felületre, ahol kifizetheti a jegyeket.

Egyéb követelmények:

ha nem adott meg darabszámot, akkor hibaüzenetet kap,  
ha nem annyi ülőhelyet foglalt, amennyi az előzőleg megadott darabszám, akkor szintén hibaüzenetet kap,  
ha nem véglegesíti a fizetést 10 percen belül, akkor a felhasználó által előzőleg lefoglalt helyek automatikusan felszabadulnak.

Az alábbiak közül melyik teszteset automatizálásával ajánlott kezdeni abban az esetben, ha még nincsenek automata tesztek a projekten és korlátozott időtartam áll rendelkezésre a tesztelés végrehajtásához?

### Válasz

- ☐ Hat db hely lefoglalása egymás mellett, majd az egyik foglalás áthelyezése egy másik székre
- ☐ Fizetési véglegesítés határidejének túllépése, majd az ülőhelyek felszabadulásának ellenőrzése
- ☐ A nem megadott darabszám esetén megjelenő hibaüzenet szövegének ellenőrzése
- ☒ Két db ülőhely lefoglalása, majd fizetés

### Magyarázat

A helyes válasz:

Két db ülőhely lefoglalása, majd fizetés

Az automata tesztelésénél érdemes a legfontosabb, a felhasználó által leggyakrabban végrehajtott esetet automatizálni először, különösen, ha tesztelési időkorlát is adott a projekten.

[Legfontosabb tudnivalók](#)

[Kapcsolat](#)

[Versenyszabályzat](#)

[Adatvédelem](#)

© 2022 Human Priority Kft.

KÉSZÍTETTE

