

Házi Feladat

Beadási határidő:
2025.10.10. 23:59:59

0. Feladat:

Regisztrálj be a következő oldalra és slack-en vagy mail-ben (vince.dulicz@gmail.com) küldd el a frissen elkészült repository-d linkjét, ahova az órai és házi feladatok kerülnek feltöltésre a továbbiakban.

New repo: <https://docs.github.com/en/repositories/creating-and-managing-repositories/creating-a-new-repository>

<https://github.com/>

I. Feladat

Adott két lista:

fogalmak = ['majorság', 'hűbéres', 'jobbággy', 'nemes', 'tized', 'kilenced', 'robot', 'szügyhám', 'vetésforgó', 'ugar', 'lovag']

meghatározások = ['Egy-egy nagybirtok vagy valamely részének igazgatási központja.', 'Aki örökletes használatra megkapja a földet.', 'Telkes paraszt, aki a földesúrtól kapott földön gazdálkodik.', 'Kiváltságos réteg.', 'Egyházi adó.', 'Földesúrnak beszolgáltatott adó.', 'Ötvenkét igás, vagy 104 kézimunka nap kötelezettség.', 'Igavonási találmány, melynek köszönhetően nem az állat nyakában van a húzó eszköz.', 'A termőföld használata évszakonként más és más.', 'Művelés alá nem vont terület.', 'Vagyonos katonai szolgálattelvő lóval, páncéllal.']

A két lista indexei megfelelnek egymásnak: az egyik listában szereplő fogalmakhoz a másik listában ugyanazon indexű *meghatározások* tartoznak, így mindegyik *fogalomhoz* a megfelelő párosítás található.

fogalmak[0] -> meghatározások[0]

fogalmak[1] -> meghatározások[1]

...

fogalmak[n] -> meghatározások[n]

1. Dictionary comprehension-el oldd meg, hogy a fogalmak lista elemei legyen a dict. kulcsa és a meghatározások lista elemei pedig az érték.
2. Kiindulásnak használd a fenti listát amely tartalmazza a fogalmak és meghatározások listáját.
3. Véletlenszerűen válassz ki egy meghatározást, azt írd ki.
4. Kérdezz rá a felhasználótól, hogy a meghatározáshoz milyen fogalom társul.
5. Ha eltalálta a fogalmat, akkor írd ki, hogy "Helyes válasz!". Ha nem találta el, akkor írd ki, hogy "Rossz válasz. Helyesen: " és ide írd ki a helyes választ.

6. A válasz ellenőrzésekor a kis-nagybetű eltérések és a kezdő-záró szóközők ne számítsanak. Azaz akkor is tekintsük helyesnek a választ, ha azt a felhasználó nagybetűkkel írta és az elejére, végére gépelt több szóközt is.

7. Addig ismételd a kikérdezést, ameddig mindegyik fogalomra rá nem kérdeztél. Mindegyikre csak egyszer kérdezz rá.

8. A program kommunikációját a mintának megfelelően szövegezd!

9. A program legyen képes fájlba kiírni a user által megadott válaszokat a hozzá tartozó kérdés és válasz párossal.

+ bónusz: adatok beolvasása fájlból flag beállításával

+ A program a fogalom bekérésre ha END, Q, vagy QUIT (ne legyen érzékeny a betűre, lehet kilépés az EnD és qUiT is) válasz érkezik akkor a program írja ki, hány helyes és hány helytelen választ adott, százalékosan is tegye ezt meg.

Középkori történelem fogalmakat kérdezzük ki.

Milyen fogalmat ír le az alábbi meghatározás:

Kiváltságos réteg.
Mi a fogalom: **nemes**
Helyes válasz!

A termőföld használata évszakonként más és más.
Mi a fogalom: **VETÉsforgó**
Helyes válasz!

Aki örökletes használatra megkapja a földet.
Mi a fogalom: **jobbágy**
Rossz válasz. Helyesen: hűbéres

Középkori történelem fogalmakat kérdezzük ki.

Milyen fogalmat ír le az alábbi meghatározás:

Földesúrnak beszolgáltatott adó.
Mi a fogalom: **tized**
Rossz válasz. Helyesen: kilenced

Egy-egy nagybirtok vagy valamely részének igazgatási központja.
Mi a fogalom: **Majorság**
Helyes válasz!

II. Feladat

Adott az alábbi három lista:

```
ranges = [class_a_range, class_b_range, class_c_range, class_d_range, class_e_range]
```

```
descriptions = [class_a_description, class_b_description, class_c_description,  
class_d_description, class_e_description]
```

```
class_labels = ["A", "B", "C", "D", "E"]
```

A listák indexei megegyeznek a hozzájuk tartozó leírással és címkével.

```
ranges[0] -> descriptions[0] -> class_labels[0]  
ranges[1] -> descriptions[1] -> class_labels[1]  
...  
ranges[n] -> descriptions[n] -> class_labels[n]
```

1. Dictionary comprehension-el oldd meg, hogy egy alábbi módon meghatározott dict.-et kapj:

```
ip_classes = {  
    "A": {"range": class_a_range, "description": class_a_description},  
    "B": {"range": class_b_range, "description": class_b_description},  
    "C": {"range": class_c_range, "description": class_c_description},  
    "D": {"range": class_d_range, "description": class_d_description},  
    "E": {"range": class_e_range, "description": class_e_description}  
}
```

2. Olvassunk be egy teljes IP címet és írjuk ki, hogy az milyen osztályú cím
3. A felhasználó biztosan jól adja meg, azt nem kell ellenőrizni.
4. A program kommunikációját a mintának megfelelően alakítsd.
Egy IP cím osztályát állapítjuk meg.

Adj meg egy teljes IP címet (x.x.x.x/x): [113.255.255.255/24](#)
A megadott IP cím osztálya: A

Egy IP cím osztályát állapítjuk meg.

Adj meg egy teljes IP címet (x.x.x.x/x): [196.255.255.255/24](#)
A megadott IP cím osztálya: C

III. Feladat

Adott az alábbi string-es formájú adathalmaz. Alakítsuk át listává list comp.-el és írjuk ki.

Saját függvényel oldjuk meg a kiírást, aminek három paraméterét lehet állítani

(akár **args*-al megoldva, akár több paraméterrel)

a *nev*, *kor*, *varos* ahol ha *False* értéket kap, az adott oszlop nem jelenik meg.

Alap esetben ez az érték legyen *True*, tehát mindent kiírunk.

```
data_list_string_format = „Név;Életkor;Város Németh Kamilla;19;Debrecen Fekete  
Géza;18;Pécs Kovács Péter;27;Budapest Kiss Tibor;20;Debrecen Szabó  
Erzsébet;21;Budapest Szilágyi Ede;18;Pécs Agárdi Pál;26;Budapest Pálosi  
Richárd;23;Budapest Budai Máté;19;Debrecen Karácsony Antal;20;Budapest Aradi  
Márta;27;Pécs Piros Adél;29;Debrecen Bíró Zsolt;16;Budapest Szabados  
Attila;25;Debrecen Román Sarolta;24;Budapest Virág Bertalan;22;Pécs Varga  
Imre;18;Budapest Tóth Sándor;22;Debrecen Nagy Ibolya;23;Pécs Horváth  
Ferenc;17;Budapest Balogh Edina;26;Budapest”
```

Majd a listát dict. comp.-el alakítsuk dict.-é az alábbi módon:

```
[  
{'nev': 'Németh Kamilla', 'age': 19, 'city': 'Debrecen', 'other': []},  
{'nev': 'Fekete Géza', 'age': 18, 'city': 'Pécs', 'other': []},  
{'nev': 'Kovács Péter', 'age': 27, 'city': 'Budapest', 'other': []},  
{'nev': 'Kiss Tibor', 'age': 20, 'city': 'Debrecen', 'other': []},  
{...}  
]
```

Írj függvényt ami rendszerezi a 12-20, 21-25, 26-32, 33+ éveseket külön-külön,
Írjuk ki azt is melyik a leggyakoribb város, mi az átlagéletkor.

Tájékoztassuk a felhasználót, kik élnek azonos városban, de itt a nevük csak
Vezetéknév K. módon jelenjen meg.