

PROGRAMOZÁS

A programozás rész három, egymástól független, egyre nehezedő feladatból áll. A megoldásokat Python nyelven kérjük elkészíteni. A programok megírásakor a felhasználó által megadott adatok helyességét, érvényességét nem kell ellenőrizni, feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek. Ügyeljen a mintának megfelelő kiíratásra – ha a felhasználótól kér be adatot, jelenítse meg a képernyőn, hogy milyen értéket vár! Az ékezetmentes kiíratás is elfogadott.

1. Feladat

A vasút kényelmes és környezetbarát megoldás a távoli helyek felkereséséhez. Az alábbi feladatok a vasútközlekedéshez kapcsolódnak. A feladat megoldásához kövesse az alábbi utasításokat, a programját **vonatA.py** néven mentse!

1. Kérje be a felhasználótól és tárolja el, hogy mekkora távolságra utazik.
2. Kérje be a felhasználótól és tárolja el, hogy mennyibe került az utazás.
3. Számolja ki, hogy egy kilométerre mennyibe került az utazás. Az eredményt kerekítse 10 Ft-ra.
4. Egy kilométer megtétele autóval 36 Ft-ba kerül. Adjon választ a felhasználónak teljes mondatban, hogy az utazása autóval vagy vonattal olcsóbb!

Minták a szöveges kimenet kialakításához:

```
1. feladat:  
Mekkora távolságra utazik (Km)? 223  
Mennyibe került az utazás? 1350  
Az út egy kilométerre 10 Ft-ba került.  
Az út vonattal olcsóbb.
```

```
1. feladat:  
Mekkora távolságra utazik (Km)? 135  
Mennyibe került az utazás? 4950  
Az út egy kilométerre 40 Ft-ba került.  
Az út autóval olcsóbb.
```

8 pont

2. Feladat

Az ország egyik legforgalmasabb vasútvonala a 30A. A 30a.txt nevű fájl ennek a vasútvonalnak az állomásait és megállóhelyeit tartalmazza.

A feladat megoldásához kövesse az alábbi utasításokat, a programját **vonatB.py** néven mentse!

1. Írjon metódust, feltoltes néven, melyben beolvassa a „30a.txt” nevű fájl, és tárolja egy tetszőleges adatszerkezetben. Amennyiben ez nem sikerül, tetszőleges szöveges adatok sorozatával dolgozhat: adjon meg néhány csokoládét és dolgozzon azzal!

2. Írjon metódust `megalloKereses` néven, mely egy paraméterként kapott megállóhelyről meghatározza, hogy az adott megállóhely szerepel-e a listában.
 1. A keresésnél ügyeljen arra, hogy a kis-nagy betűk használata ne okozzon eltérő eredményt. Figyeljen arra, hogy a megállóhelyek neve végén a „mh.” szó található.
 2. A keresést ne folytassa, ha meg tudja adni a választ!
 3. A függvény visszatérési értéke logikai típus legyen!
3. Hozzon létre egy metódust `megallo` néven, melyben bekér a felhasználótól egy megállóhelyet!
4. Az imént bekért megállóhelyről a `megalloKereses` függvény segítségével határozza meg, hogy szerepel-e a listában! Ha szerepel, írja ki a mintának megfelelően a sorszámát!

Minták a szöveges kimenet kialakításához:

2. feladat:

```
A 30A útvonal beolvasása sikeresen megtörtént
Kérek, adj meg egy megállóhelyet: Tárnok
Van ilyen nevű megálló az útvonalon, a 6. helyen
```

2. feladat:

```
A 30A útvonal beolvasása sikeresen megtörtént
Kérek, adj meg egy megállóhelyet: Tárnok
Nincs ilyen megállóhely a listában.
```

14 pont

3. Feladat

A vasút számára csak 5 ellenség van: az utasok és a négy évszak. Az alábbi feladatban az időjárási okok miatti késésekkel fogunk foglalkozni.

A „kesesek.csv” fájlban az i. A fájl szerkezete a következő:

```
Következmény megnevezése; Időjárási ok ;darab ;min; max; átlag
Állomástávolságú közlekedés;Vihar;6;2;8;5,67
Áramellátási zavar;Vihar, erős szél;9;3;63;14,44 ...
```

Darab: a késések száma

Min, max, átlag: a késések ideje, percben

Minden feladatot egy-egy külön metódusban hajtson végre! A feladat megoldásához kövesse az alábbi utasításokat, a programját **vonatC.py** néven mentse!

1. Hozzon létre egy osztályt a kesesek.csv fájl tartalmának a tárolására!
2. Az osztály felhasználásával olvassa be a fájlban lévő adatokat egy listába, és a további feladatokban ezzel a listával dolgozzon. Az első sor fejlécet tartalmaz, amelyet nem kell tárolni!

3. Írja ki a listában lévő adatok alapján hány vonat késett, mennyi volt a legkisebb és a legnagyobb késés!
4. Listázza ki egy sorba a beolvasott adatok közül az „Vihar” időjárási ok miatti következményeket! A metódus új állomány esetén is helyes értéket adjon eredményül.
5. Készítsen tájékoztatót, egy véletlenül kiválasztott következményről! A tájékoztatóban adja meg:
 1. a következmény nevét, időjárási okát
 2. az késő vonatok számát, az átlagos késést másodpercben, 10-re kerekítve úgy, hogy ne legyen átváltási veszteség (kerekítés után ne legyen alacsonyabb, mint kerekítés nélkül)
 3. jelentős probléma-e (a probléma jelentős, ha 100 vonatnál több késett emiatt)

Minták a szöveges kimenet kialakításához:

3. feladat:

Beolvasás

Az adatok alapján 487 vonat késett, minimum 1 percet, maximum 290 percet

Vihar miatti következmények: Állomástávolságú közlekedés, Áramellátási zavar, Biztosítóberendezési meghibásodás, Fabledőlés, Hamisfoglaltság, Mindenhol megálló vonat, Használhatatlan sorompó, Szolgálatképtelen mozdony (áramszedő törés), Vonatpótló autóbuszos közlekedés

Tájékoztató:

Következmény: Leeresztett áramszedős közlekedés

Időjárási ok: Erős szél

Késő vonatok száma: 4 db; átlagos késés: 120 másodperc

Nem jelentős probléma.

18 pont