

Szöveges állományok kezelése

Készítette: Vastag Atila

2017

Kezdjük egy file megnyitásával és tartalmának a képernyőre írásával, és e file tartalma legyen a következő:

```
# -*- coding: ISO-8859-2 -*-  
from typing import *  
from io import *
```

```
file: TextIOWrapper = open ("./data/ip.txt", encoding='utf-  
8', mode='r')
```

```
oneLine: str = file.readline()  
allLines: List[str] = file.readlines()
```

```
file.close()
```

```
print(allLines)
```

```
project  
|  
-- data  
    |  
    --ip.txt  
|  
--main.py
```

```
10.0.0.1  
10.0.0.2  
10.0.0.3  
10.0.0.4  
10.0.0.5  
10.0.0.6
```

Egész file beolvasása egyetlen karakterláncba az alábbiak szerint történik.

Először is az **open()** függvény segítségével megnyitjuk a beolvasni kívánt file-t. Az **open()** függvény egy előkészítő függvény, amely arra készíti fel a rendszert, hogy az adott merevelemezen (vagy más adattárolón) fellelhető file-lal dolgozni tudjunk. Legegyszerűbb esetben ez a file nevének a megadásával történik.

```
file: TextIOWrapper = open ("../data/ip.txt", encoding='utf-8')
```

A fenti kódsor hatására a **file** változón keresztül férhetünk hozzá a file tartalmához.

Meghívható függvények a **file**-on:

- **read(n)** → függvény a megadott **n** számú karaktert olvassa ki az aktuális filefolyamból
- **tell()** → függvényt alkalmazva megkaphatjuk az aktuális folyampozíciót (épp hanyadik karakter kiolvasásánál tart).
- **seek()** → függvény segítségével állíthatjuk be egy megnyitott filefolyam aktuális pozícióját.
- **readline()** → függvény segítségével a megnyitott filefolyam egy sorát olvashatjuk ki mint string
- **readlines()** → függvény segítségével a megnyitott filefolyam több (összes) sorát olvashatjuk ki mint string lista

A nyitott fájlok erőforrásokat fogyasztanak, és a fájl módjától függően más programok esetleg nem tudják elérni azokat. Fontos a fájlokat azonnal bezárni, amint végeztél a feldolgozásukkal.

```
file.close()
```

Ha már minden olvasás és írás befejeződött, akkor a **close()** utasítással zárhatjuk be a folyamatot, azaz ez a parancs szakítja meg a kommunikációt az adattárolóval.

Hogy véletlenül se felejtjük el bezárni a file-t pythonban van egy kényelmes/hasznos konstrukció a **with** . Ez meghívja az adott objektum `__enter__()` es `__exit__()` tagfüggvényeit. A with block után a változó megmarad, de a file már zárva van.

A **with** előnyei:

- biztos nem felejtjük el bezárni a filet
- ha valami (i/o) hiba történik, a fájl akkor is bezáródik
- nyertünk egy sor kódot
- szinte biztos, hogy nem keverjük össze a különböző fájlokat,amibe írni akarunk file1,file2, ...
- kizárólag ebben a blokkban tudunk a fájlba írni
- rövid ideig van nyitva egy fájl, kevesebb az esélye, hogy több helyről van írva olvasva egyszerre
- érthető a kódban, hogy mi hol történik

```
# -*- coding: ISO-8859-2 -*-
```

```
from typing import *  
from io import *
```

```
with open ("./data/ip.txt", encoding='utf-8,mode="r")) as file:  
    oneLine: str = file.readline()  
    allLines: List[str] = file.readlines()
```

```
print(oneLine)  
print(allLines)
```

Most úgy ahogy az a nagykönyvben is meg van írva

```
# -*- coding: ISO-8859-2 -*-  
  
from typing import *  
from io import *  
  
oneLine: str = None  
allLines: List[str] = None  
  
try:  
    with open ("../data/ip.txt", encoding='utf-8, mode="r") as file:  
        oneLine = file.readline()  
        allLines = file.readlines()  
except FileNotFoundError as ex:  
    print(f"{ex.filename} nem található!")  
else:  
    print(oneLine)  
    print(allLines)
```


File megnyitási módok:

| Character | Meaning |
|-----------|---|
| 'r' | open for reading (default) |
| 'w' | open for writing, truncating the file first |
| 'x' | open for exclusive creation, failing if the file already exists |
| 'a' | open for writing, appending to the end of the file if it exists |
| 'b' | binary mode |
| 't' | text mode (default) |
| '+' | open a disk file for updating (reading and writing) |

Írás szövegfájlokba

A fájlokba nagyjából ugyanúgy írhatasz, ahogy olvasod azokat. Először megnyitasz egy fájlt, és kapsz egy adatfolyam-objektumot, aztán metódusokat használsz az adatfolyam-objektumon, hogy adatokat írsz a fájlba, majd bezárod a fájlt.

Egy fájl írásra való megnyitásához használd az **open()** függvényt, és add meg az írás módot. Két fájl mód használható az íráshoz:

Az „*írás*” mód felülírja a fájlt. Add át a **mode='w'** paramétert az **open()** függvénynek. A „*hozzáfűzés*” mód a fájl végéhez fog adatokat adni. Add át a **mode='a'** paramétert az **open()** függvénynek.

Mindkét mód automatikusan létrehozza a fájlt, ha még nem létezik, így nincs szükség semmi szöszmötölős „ha a fájl még nem létezik, akkor hozz létre egy új fájlt csak hogy először megnyithasd” függvényre. Csak nyisd meg a fájlt, és kezd írn.

Mindig zárd be a fájlokat, amint végeztél az írásukkal, hogy felszabadítsd a fájlhivatkozást, és biztosítsd, hogy az adatok valójában kiírásra kerüljenek a háttértárra. Ahogyan az adatok fájlból olvasásakor, itt is meghívhatod az adatfolyam-objektum **close()** metódusát, vagy használhatod a with utasítást, és rábízhatsz a fájl bezárását a Pythonra.

Fogadni mernék, hogy kitalálsz, melyik módszert javaslom.

```
# -*- coding: ISO-8859-2 -*-  
  
from typing import *  
  
from io import *  
  
try:  
    with open ("./data/ip.txt", encoding='utf-  
8', mode="a") as file:  
        file.write("123.123.123.123")  
except FileNotFoundError as ex:  
    print(f"{ex.filename} nem található!")
```

```
10.0.0.1  
10.0.0.2  
10.0.0.3  
10.0.0.4  
10.0.0.5  
10.0.0.6  
123.123.123.123
```

1 – Egy szöveges állományban, **eredmeny.txt**, az érettségizők pontjai vannak elmentve a következő módon, soronként és a sorokban tabulátorral elválasztva:

Virág 9,28

Jázmin 6,26

Feladatunk, hogy kikeressük a legtöbb és legkevesebb pontot elért érettségizőt és a **max.txt** illetve a **min.txt** állományokba írjuk bele őket. Egyes logikai egészeket alkotó műveleteket függvényekkel oldjuk meg.

2 – A programunk feladata, hogy hőmérsékletet mérjen reggel délben és este, majd ezeket a hőmérsékleteket a **meresek.txt** állományokba mentse a hét minden napján (napi három mérés).

A hőmérsékleteket **Random** számmal adjuk meg 0 és 40 közt!

Keressük ki a hét végén, miután megtörtént az ősz mérés, hogy mekkora volt az átlag hőmérséklet reggel, délben és este és az **atlag.txt** állományba mentjük el.

Egyes logikai egészeket alkotó műveleteket függvényekkel oldjuk meg.

3 – Egy **forrás.txt** állományban számoljuk meg a magán és mássalhangzókat, számokat és egyéb szimbólumokat, majd az eredményt írjuk az **eredmeny.txt** állományba.

Egyes logikai egészeket alkotó műveleteket függvényekkel oldjuk meg.

forrás.txt:

“Microsoft suspends the Intel Kaby Lake and Ryzen AMD processors for Windows 7 and 8. Unfortunately, the tech mogul has no plans in bringing the next-gen chips to the old operating systems in the future. In fact, just last week, Microsoft released the KB 4012982 error entitled "Your PC uses a processor that isn't supported on this version of Widows.”

<http://www.universityherald.com/articles/69674/20170317/microsoft-blocks-windows-7-8-updates-ryzen-amd-intel-kaby.htm>

4 – Egy szöveges dokumentumban e-mail címek találhatóak és a hozzájuk tartozó jelszó. Kérjük meg a felhasználót hogy adja meg az e-mail címét és a jelszót és ha ez megtalálható a szöveges dokumentumban léptessük be a rendszerbe és egy log.txt állományba írjuk be, hogy mikor és ki lépett be. Az e-mail cím bekérésénél ügyeljünk arra, hogy megfeleljen az e-mail cím formátumának.

A feladatot az OOP-al oldjuk megoldjuk meg.

5 – A **konyvek.txt** állományban az adatok a következő módon vannak tárolva:

- Vezetéknév (íróé),
 - Keresztnév (íróé),
 - SzületésiDátum,
 - Cím,
 - ISBN,
 - Kiadó,
 - KiadvásiÉv,
 - ár,
 - Téma,
 - Oldalszám,
 - Honorárium (amit a könyvért kapott az író)
- a) Írjuk ki a képernyőre az össz adatot
- b) Keressük ki az informatika témájú könyveket és mentjük el őket az **informatika.txt** állományba
- c) Az 1900.txt állományba mentjük el azokat a könyveket amelyek az 1900-as években íródtak
- d) Rendezzük az adatokat a könyvek oldalainak száma szerint csökkenő sorrendbe és a **sorbarakott.txt** állományba mentjük el.
- e) „kategoriak.txt” állományba mentse el a könyveket téma szerint. Például:
- Thriller:
- könyv1
 - könyv2
- Krimi:
- könyv1
 - könyv2

6 – A **roplabda.txt** állományban az adatok a következő módon vannak tárolva:

Név,
Magasság,
Poszt,
Nemzetiség,
Csapat,
Ország (ahol a csapat játszik)

- a) Írjuk ki a képernyőre az össz adatot
- b) Keressük ki az ütő játékosokat az **utok.txt** állományba
- c) A **csapattagok.txt** állományba mentjük a csapatokat és a hozzájuk tartozó játékosokat a következő formában:
Telekom Baku: Yelizaveta Mammadova, Yekaterina Gamova,
- d) Rendezzük a játékosokat magasság szerint növekvő sorrendbe és a **magaslatok.txt** állományba mentjük el.
- e) Mutassuk be a **nemzetisegek.txt** állományba, hogy mely nemzetiségek képviseltetik magukat a röplabdavilágban mint játékosok és milyen számban.
- f) **atlagnalmagasabbak.txt** állományba keressük azon játékosok nevét és magasságát akik magasabbak mint az „adatbázisban” szereplő játékosok átlagos magasságánál.
- g) állítsa növekvő sorrendbe a posztok szerint a játékosok össz magasságát
- h) egy szöveges állományba, „*alacsonyak.txt*” keresse ki a játékosok átlagmagasságától alacsonyabb játékosokat. Az állomány tartalmazza a játékosok nevét, magasságát és hogy mennyivel alacsonyabbak az átlagnál, 2 tizedes pontossággal.

7 – A **magyarvarosok.txt** állományban az adatok a következő módon vannak tárolva:

Nev (város neve),
Város típusa,
Megye név,
Járas,
Kistérség,
Népesség,
Terület

- a) Írjuk ki a képernyőre az össz adatot
- b) Keressük ki a megyeszékhely megyei jogú városokat és mentjük el a **megyeloguvarosok.txt** állományba
- c) Az **nepesseg.txt** állományba mentjük el azokat a településeket és a hozzájuk tartozó adatokat, ahol a népesség 50.000 és 100.000 közt van
- d) Keressük ki azokat a településeket, melyek területei meghaladják az 200-at és a **nagyteruletek.txt** állományba mentjük el.
- e) Keressük ki Békés megye össz települését és a **bekes.txt** állományba mentjük el.
- f) **megyeteruletek.txt** állományba mentjük el a megye nevét és területének nagyságát.

8 – A **lotto.txt** állományban az adatok a következő módon vannak tárolva:

Név,

tippek

- a) Írjuk ki a képernyőre az össz adatot
- b) Random számok segítségével generáljuk le a napi 7 nyerő számot és írjuk őket egy szüveges állományba melynek az aktuális nap lesz a neve
- c) Keressük ki, van(ak)-e 7 találatos szelvény(ek), ha igen írjuk ki a nyertesek nevét a **nyertesek-{mai dátum}.txt** állományba.
- d) Keressük ki, hogy a befizetett játékosok hány találatot értek el, és mentjük el a **talalatok-{mai dátum}.txt** állományba a játékos nevét és a találatainak számát

9 – A **vezetok.txt** állományban az adatok a következő módon vannak tárolva:

Vezetéknév,Keresztnév,Anya vezetéke, Anya keresztnéve

Születés időpont,Születés helye,Megye,Ország

Utca,Házszám,Írányítószám,Város,Megye,Ország

Kategóriák

Az egyes egészek tabulátorral vannak elválasztva, az egészek adatai pedig vesszővel.

Írjunk programot, mely menü segítségével lehetővé teszi a következő adatok kérését:

- a) Írjuk ki a képernyőre az össz adatot a vezetőkről
- b) A felhasználó által megadott megyére a **megye-vezetoi.txt** állományba elmenti a megadott megyében lakó vezetőket.
- c) A felhasználó által megadott kategóriával rendelkező vezetőket a **{kategoria nev}-kategoria.txt** állományba menti el.
- d) A **fiatalok.txt** állományba kikeresi azokat a vezetőket akik 18 és 21 év között vannak.
- e) **kulfoldi.txt** állományba azokat a vezetőket keresi ki, akik nem Magyarországon születtek.

10 – Az **nb1.txt** állományban az adatok a következő módon vannak tárolva:

A labdarúgó mezére írt szám (szám)

A labdarúgó utóneve (szöveg); előfordul, hogy valaki felvett nevet használ, ilyenkor üres is lehet

A labdarúgó vezetékneve (szöveg)

A labdarúgó születési dátuma (dátum)

Értéke igaz, ha magyar állampolgár (is) a labdarúgó (logikai) [-1 igaz, 0 hamis]

Értéke igaz, ha külföldi állampolgár (is) a labdarúgó (logikai) [-1 igaz, 0 hamis]

A labdarúgó euró ezrekben kifejezett értéke (szám)

A klub neve (szöveg)

A poszt neve (szöveg), például kapus, bal oldali védő, bal szélső

Írjunk programot, mely menü segítségével lehetővé teszi a következő adatok kérését:

a) A kapusokon kívül mindenkit mezőnyjátékosnak tekintünk. Keresse ki a legidősebb mezőnyjátékos vezeték- és utónevét, valamint születési dátumát! (Feltételezheti, hogy csak egy ilyen játékos van.)

- b)** Határozza meg hány magyar, külföldi és kettős állampolgárságú játékos van!
- c)** Határozza meg játékosok összértékét csapatonként és írja ki a képernyőre! A csapatok neve és a játékosainak összértéke jelenjen meg!
- d)** Keresse ki, hogy mely csapatoknál mely posztokon van csupán egy szerződött játékos! Írja ki a csapat nevet és a posztot amire csak egy játékost szerződtek!
- e)** Keressük ki azon játékosokat, akiknek az értékük nem haladja meg a játékosok értékének átlag értékét.
- f)** Írja ki azon játékosok nevét, születési dátumát és csapataik nevét, akik 18 és 21 év közt vannak és magyar állampolgárok. Ha nincs ilyen, akkor megfelelő üzenettel helyettesítse a kimenetet.
- g)** A „hazai.txt” illetve a „legios.txt” állományokba keresse ki a magyar, illetve a külföldi állampolgárságú játékosokat csapatonként. A szöveges állományoknak tartalmazniuk kell a csapat nevét majd alatta felsorolva a játékosok teljes nevét, poszt nevet és értéküket.

11 - Adva van az adatok.txt állományban a Magyar Női Röplabda Bajnokság csapatainak pontszámai a következő képpen:

Békéscsaba
1,2,1,3,3,3,3,3,3,1,2,2,1,3,3,1,3

ahol a csapat nevét tabulátorral elválasztva követik a az elért pontok mérkőzésenként (max 18 lejátszott mérkőzés).

- a) Hány csapat vett részt a bajnokságban?
- b) Ki nyerte a bajnokságot?
- c) Döntetlen mérkőzéskor a csapat 2 pontot szerez. Mutassa be csapatonként ki hány döntetlen mérkőzést játszott le!
- d) Ha egy mérkőzés 5 szetben dől el, akkor a vesztes csapat 1 pontot szerez. Mely csapatok játszottak 5 szettes mérkőzést és hányat?
- e) Ki a bajnokság első három helyezete. Mutassa be mintának megfelelően:

| helyezés | - csapat neve | pontszám |
|----------|---------------|----------|
|----------|---------------|----------|
- e) Az elért pontok alapján, az utolsó három csapat kiesik az első osztályból. Kik ők?
- f) Mutassa be csapatonként a győzelmi és verességi arányt csapatonként!
- g) Mely csapatok győzelmi aránya van az átlag alatt?