

Tema 14.

• Fișiere text în Python

Ce ne așteaptă?

1. Tipuri de fișiere și regimuri de lucru
2. Deschiderea și închiderea fișierelor
3. Înscrierea și citirea datelor din fișiere

1. Ce tipuri de fișiere se pot procesa în Python?
2. Care sunt regimurile de lucru cu fișierele text?
3. Care sunt particularitățile regimurilor de lucru cu fișierele?
4. Cum se deschide și se închide un fișier în Python?
5. Ce informații referitoare la fișier pot fi citite în Python?
6. Care este rolul cuvântului cheie *with*?
7. Care sunt metodele de înscrisere a datelor în fișiere text?
8. Care sunt metodele de citire a datelor din fișiere text?
9. Cum se determina poziția cursorului într-un fișier text?
10. Cum se schimbă poziția cursorului într-un fișier text?
11. Cum se realizează procedurile de citire și înscrisere în regim *r+*?
12. Cum se realizează procedurile de citire și înscrisere în regim *w+*?
13. Cum se realizează procedurile de citire și înscrisere în regim *a+*?

1. Tipuri de fișiere și regimuri de lucru

Tipuri de fișiere

- **Fișierul** – o resursă pentru stocarea datelor.
- **Fișiere text** – fișiere în care datele sunt stocate sub formă de caractere sau stringuri
ex: toate fișierele ce conțin text
- **Fișiere binare** – fișiere în care datele sunt stocate sub formă de baiți
ex: *fișiere audio, video sau chiar text*
- **Regim de lucru cu fișierele** specifică operația care se va realiza asupra fișierului (înscrisere, citire, completare)

Regimurile de lucru cu fișierele text (1)

- **Regimul r (regim implicit):**
 - Deschide un fișier pentru operația de citire
 - După deschidere cursorul se va poziționa la începutul fișierului
 - Dacă fișierul nu există atunci se generează excepția `FileNotFoundException`
- **Regimul w:**
 - Deschide un fișier pentru operația de înscrisere
 - Dacă fișierul conține date, acestea se vor suprascrie cu datele noi
 - Dacă fișierul nu există atunci acesta se creează
- **Regimul a:**
 - Deschide un fișier pentru operația de completarea a datelor
 - Dacă fișierul conține date acestea se vor păstra adăugându-se date noi
 - Dacă fișierul nu există atunci acesta se creează

Regimurile de lucru cu fișierele text (2)

- **Regimul r+:**
 - Deschide un fișier pentru operația de citire și apoi de înscrisere
 - După deschidere cursorul se va poziționa la începutul fișierului
 - Dacă fișierul conține date, acestea se vor suprascrie cu datele noi
- **Regimul w+:**
 - Deschide un fișier pentru operația de înscrisere și apoi de citire
 - Dacă fișierul conține date acestea se vor suprascrie cu datele noi
- **Regimul a+:**
 - Deschide un fișier pentru operația de completare și apoi de citire
 - După deschidere cursorul se va poziționa la sfârșitul fișierului
 - Dacă fișierul nu există atunci acesta se creează

Deosebiri dintre regimurile de lucru

	r	r+	w	w+	a	a
Citire	+	+		+		+
Înscriere		+	+	+	+	+
Creare fișier			+	+	+	+
Ștergere și suprapunere			+	+		
Poziție cursor la început	+	+	+	+		
Poziție cursor la sfârșit					+	+

2. Deschiderea și închiderea fișierelor

Metode de deschidere și de închidere a fișierului

- Metoda `open()` – deschide un fișier

nume_variabila = `open(nume_fisier, regim_de_lucru)`

- Exemplu de deschidere a unui fișier

```
f=open("Exemplu.txt", 'w')
```

- Metoda `close()` – închide fișierul

nume_variabila.`close()`

- Exemplu de închidere a unui fișier

```
f.close()
```


Citirea informațiilor despre fișier

- Atributul `name` – returnează numele fișierului

nume_variabila.name

- Exemplu de citirea a numelui unui fișier

```
nume = f.name  
print(f"Numele fisierului este {nume}")
```

- Atributul `mode` – returnează regimul de lucru cu fișierul

nume_variabila.mode

- Exemplu de citirea a regimului de lucru cu fișierul

```
regim = f.mode  
print(f"Fisierul este deschis în regim {regim}")
```

- Atributul `closed` – returnează `True` dacă fișierul este închis și invers

nume_variabila.closed

- Exemplu de verificare a starii fișierului

```
print(f"Fisierul este închis: {f.closed}")
```

Verificarea posibilităților de înscriere sau citire

- Metoda `readable()` – returnează `True` dacă fișierul e în regim de citire și invers

nume_variabila.readable()

- Exemplu de citirea a numelui unui fișier

```
citire = f.readable()  
print(f"Este fisierul in regim de citire: {citire}")
```

- Metoda `writable()` – returnează `True` dacă fișierul e în regim de înscriere și invers

nume_variabila.writable()

- Exemplu de citirea a numelui unui fișier

```
inscriere = f.writable()  
print(f"Este fisierul in regim de inscriere: {inscriere}")
```

Exemplu de citire a informațiilor despre fișier

- Exemplu de citirea a informației despre fișierul exemplu.txt aflat în folderul de program

```
f=open("Exemplu.txt", "w")

nume = f.name
regim = f.mode
citire = f.readable()
inscriere = f.writable()
inchis = f.closed

print(f"Numele fisierului: {nume}",)
print(f"Regimul de lucru al fisierului: {regim}")
print(f"Este fisierul in regim de citire: {citire}")
print(f"Este fisierul in regim de scriere: {inscriere}")
print(f"Este fisierul inchis: {inchis}")

f.close()

inchis = f.closed
print(f"Este fisierul inchis: {inchis}")
```

Cuvântul cheie *with*

- Cuvântul cheie *with* are grijă de închiderea automată a fișierului după utilizarea acestuia

- Sintaxa blocului cuvântului cheie *with*:

```
with open(ume_fisier, regim) as ume_variabila:  
    ume_variabilă.ume_metode  
...
```

- Exemplu de utilizare:

```
with open("Welcome.txt", "w") as f:  
  
    print(f"Numele fisierului: {f.name}",)  
    print(f"Regimul de lucru al fisierului: {f.mode }")  
    print(f"Fisierul este inchis: {f.closed}")  
  
print(f"Fisierul este inchis: {f.closed}")
```

3. Înscrierea și citirea datelor din fișiere

Înscrierea datelor într-un fișier text – metoda write()

- Sintaxa metodei write():

nume_variabila.write(textul in scris)

- Metoda write() în regim "w"

```
with open("Welcome.txt", 'w') as f:  
    f.write("Bun venit ")  
    f.write("la cursul de Python")  
    print("Înscrierea s-a realizat cu succes")
```

- Metoda write() în regim "a"

```
with open("Welcome.txt", 'a') as f:  
    f.write(" de la ciclul I, licenta - ")  
    f.write("Universitatea Tehnica a Moldovei")  
    print("Completarea s-a realizat cu succes")
```

Înscrierea datelor într-un fișier text – metoda writelines()

- Sintaxa metodei writelines():

nume_variabila.writelines(secvență cu text)

- Metoda writelines() în regim "w"

```
with open("Welcome.txt", "w") as f:  
    f.writelines(["Bun venit ", "la cursul de Python\n"])  
    print("Înscrierea s-a realizat cu succes")
```

- Metoda writelines() în regim "a"

```
with open("Welcome.txt", 'a') as f:  
    f.writelines(["de la ciclul I, licenta -", " Universitatea  
Tehnica a Moldovei"])  
    print("Completarea s-a realizat cu succes")
```

Citirea datelor dintr-un fișier text – metoda read()

- Sintaxa metodei read():

data= nume_variabila.read() - citește tot textul

data= nume_variabila.read(n) - citește primele n caractere

- Metoda read()

```
with open("Welcome.txt", 'r') as f:  
    data=f.read()  
    print(data)
```

- Metoda read(n)

```
with open("Welcome.txt", 'r') as f:  
    data=f.read(29)  
    print(data)
```

Citirea datelor dintr-un fișier text – metodele `readline()` și `readlines()`

- Sintaxa metodelor `readline()` și `readlines()`:

`data= nume_variabila.readline()` - citește o linie din fișierul text

`data= nume_variabila.readlines(n)` - citește toate liniile returnând o listă cu acestea

- Metoda `readline()`

```
with open("Welcome.txt", 'r') as f:
    data1=f.readline()
    print(data1)
    data2=f.readline()
    print(data2)
```

- Metoda `readlines()`

```
with open("Welcome.txt", 'r') as f:
    data=f.readlines()
    print(data)
    for i in data:
        print(i)
```


Metoda tell()

- Metoda `tell()` – returnează poziția curentă a cursorului în fișierul text
- Exemplu de utilizare:

```
with open("Welcome.txt", "r") as f:  
    print(f.tell())  
    print(f.read(3))  
    print(f.tell())  
    print(f.read(6))  
    print(f.tell())
```

Metoda seek()

- Metoda seek() – permite deplasarea cursorului la poziția dorită față de început
- Sintaxa metodei seek():

nume_variabila.seek(numar_poziții)

- Exemplu de utilizare:

```
with open("Welcome.txt", "r") as f:  
    print(f.tell())  
    f.seek(3)  
    print(f.tell())  
    f.seek(10)  
    print(f.tell())
```

Citirea și înscrierea datelor în regim r+

- În regim r+ cursorul se află la începutul documentului
- **Citirea apoi înscriere** – inițial se citesc datele existente apoi se adaugă cele noi
- **Exemplu:**

```
with open("Welcome.txt", "r+") as f:  
    print(f.read())  
    f.write("\nFacultatea Electronica si Telecomunicatii")
```

- **Înscrierea apoi citirea** – date înscrise se suprapun celor existente apoi se citesc cele ramase
- **Exemplu:**

```
with open("Welcome.txt", "r+") as f:  
    f.write("Hei")  
    print(f.read())
```

Citirea și înscrierea datelor în regim w+

- În regim w+ cursorul se află la începutul documentului
- **Citirea apoi înscriere** – se șterg toate datele inițiale de aceea nu se citește nimic și apoi se înscriu datele noi

- **Exemplu:**

```
with open("Welcome.txt", "w+") as f:  
    print(f.read())  
    f.write("Bun venit la cursul de Python")
```

- **Înscrierea apoi citirea** – se șterg toate datele inițiale, se înscriu datele noi iar pentru citire e nevoie de poziționat cursorul la începutul documentului altfel nu se citește nimic
- **Exemplu:**

```
with open("Welcome.txt", "w+") as f:  
    f.write("Facultatea Electronica si Telecomunicatii ")  
    f.seek(0)  
    print(f.read())
```

Citirea și înscrierea datelor în regim a+

- În regim r+ cursorul se află la sfârșitul documentului
- **Citirea apoi înscriere** – pentru citire e nevoie de poziționat cursorul la începutul documentului altfel nu se citește nimic și apoi înscriu noile date

- **Exemplu:**

```
with open("Welcome.txt", "a+") as f:  
    f.seek(0)  
    print(f.read())  
    f.write(" Universitatea Tehnica a Moldovei")
```

- **Înscrierea apoi citirea** – se adaugă datele noi după datele inițiale, iar pentru citire e nevoie de poziționat cursorul la începutul documentului altfel nu se citește nimic
- **Exemplu:**

```
with open("Welcome.txt", "a+") as f:  
    f.write(" Universitatea Tehnica a Moldovei")  
    f.seek(0)  
    print(f.read())
```