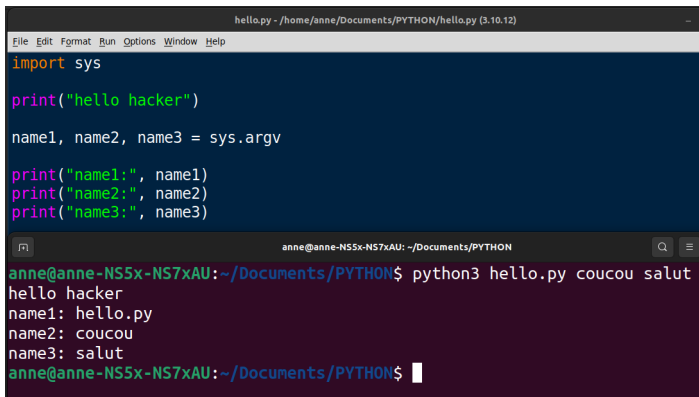


Programmation en Python

Anne Garcia-Sanchez

M2i cyber2 dev - CFA CCI Avignon
16 novembre 2023

arguments de la commande d'appel du script



The image shows a code editor window titled 'hello.py - /home/anne/Documents/PYTHON/hello.py (3.10.12)' with the following Python code:

```
import sys

print("hello hacker")

name1, name2, name3 = sys.argv

print("name1:", name1)
print("name2:", name2)
print("name3:", name3)
```

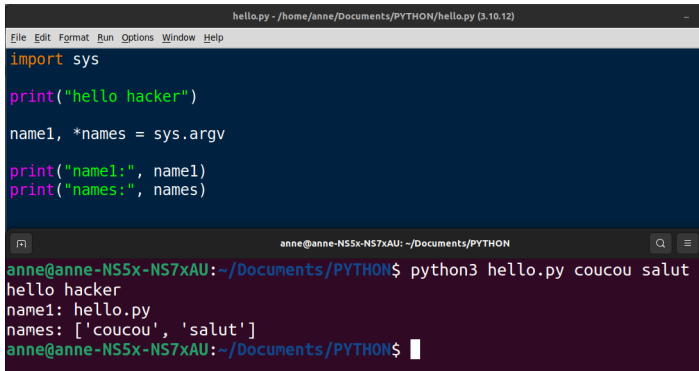
Below the code editor is a terminal window titled 'anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON'. It shows the command 'python3 hello.py coucou salut' being executed, resulting in the following output:

```
hello hacker
name1: hello.py
name2: coucou
name3: salut
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$
```

remarque:

premier argument: nom du fichier contenant script (hello.py)
et non celui de la commande qui a lancé le script (python3)

arguments de la commande d'appel du script

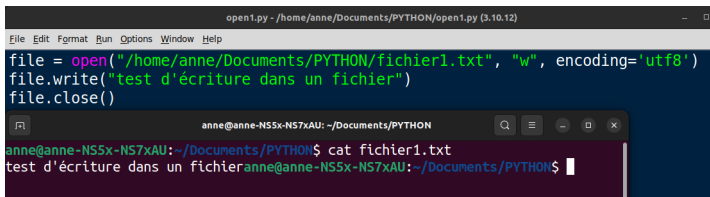


The image shows a code editor window titled 'hello.py - /home/anne/Documents/PYTHON/hello.py (3.10.12)' and a terminal window below it. The code editor contains a Python script that imports the 'sys' module, prints 'hello hacker', and then processes command-line arguments. The terminal shows the execution of 'python3 hello.py coucou salut', which produces the output 'hello hacker', 'name1: hello.py', and 'names: ['coucou', 'salut']'.

```
hello.py - /home/anne/Documents/PYTHON/hello.py (3.10.12)
File Edit Format Run Options Window Help
import sys
print("hello hacker")
name1, *names = sys.argv
print("name1:", name1)
print("names:", names)

anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ python3 hello.py coucou salut
hello hacker
name1: hello.py
names: ['coucou', 'salut']
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$
```

ouverture en écriture



The screenshot shows a Python script named `open1.py` in a text editor. The script uses the `open` function to open a file named `fichier1.txt` in write mode ('w') with UTF-8 encoding. It then writes the string `"test d'écriture dans un fichier"` to the file and closes it. Below the script, a terminal window shows the command `cat fichier1.txt` being executed, which outputs the text `test d'écriture dans un fichier`.

```
open1.py - /home/anne/Documents/PYTHON/open1.py (3.10.12)
File Edit Format Run Options Window Help
file = open("/home/anne/Documents/PYTHON/fichier1.txt", "w", encoding='utf8')
file.write("test d'écriture dans un fichier")
file.close()

anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ cat fichier1.txt
test d'écriture dans un fichieranne@anne-NS5x-NS7xAU:~/Documents/PYTHON$
```

fermeture du fichier:

nombre de fichiers ouverts limité par le système d'exploitation

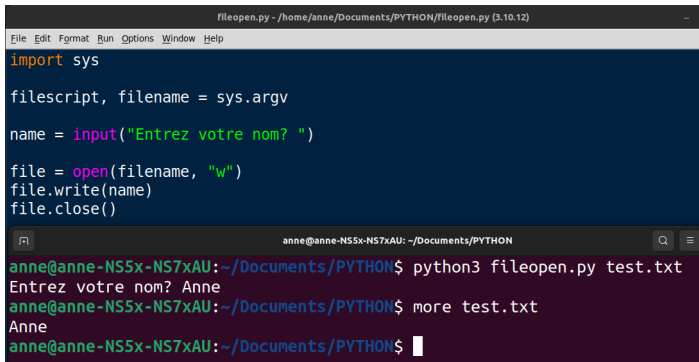
attention à l'encodage

remarque pour les windowsiens:

utiliser caractère d'échappement `\` ou raw string

`r"chemin\blabla"`

ouverture en écriture



The screenshot shows a code editor window titled 'fileopen.py - /home/anne/Documents/PYTHON/fileopen.py (3.10.12)'. The code in the editor is as follows:

```
import sys

filescrip, filename = sys.argv

name = input("Entrez votre nom? ")

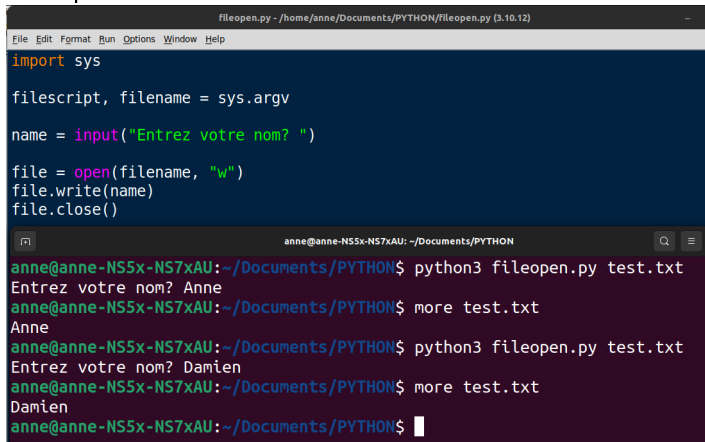
file = open(filename, "w")
file.write(name)
file.close()
```

Below the code editor is a terminal window with the title 'anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON'. The terminal shows the execution of the script:

```
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ python3 fileopen.py test.txt
Entrez votre nom? Anne
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ more test.txt
Anne
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$
```

ouverture en écriture

attention: si le fichier existe déjà, création d'un fichier vide
donc perte des données



```
fileopen.py - /home/anne/Documents/PYTHON/fileopen.py (3.10.12)
File Edit Format Run Options Window Help

import sys

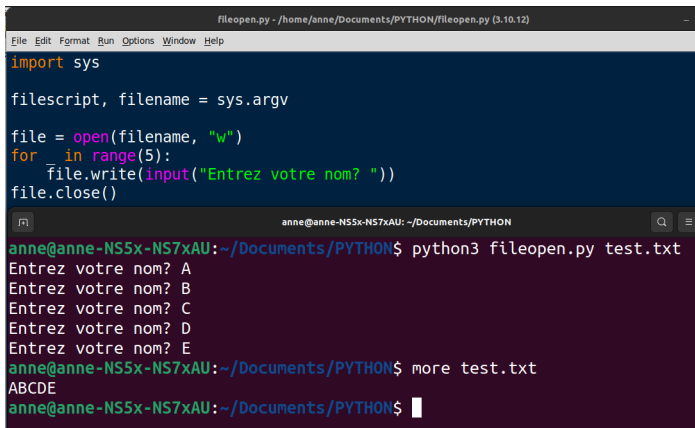
filescript, filename = sys.argv

name = input("Entrez votre nom? ")

file = open(filename, "w")
file.write(name)
file.close()

anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ python3 fileopen.py test.txt
Entrez votre nom? Anne
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ more test.txt
Anne
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ python3 fileopen.py test.txt
Entrez votre nom? Damien
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ more test.txt
Damien
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$
```

ouverture en écriture



```
fileopen.py - /home/anne/Documents/PYTHON/fileopen.py (3.10.12)
File Edit Format Run Options Window Help

import sys

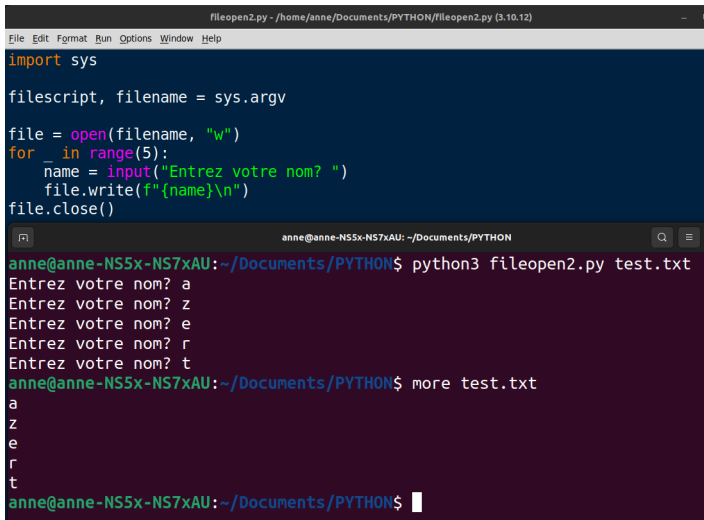
filescript, filename = sys.argv

file = open(filename, "w")
for _ in range(5):
    file.write(input("Entrez votre nom? "))
file.close()

anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ python3 fileopen.py test.txt
Entrez votre nom? A
Entrez votre nom? B
Entrez votre nom? C
Entrez votre nom? D
Entrez votre nom? E
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ more test.txt
ABCDE
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$
```

write n'ajoute pas de de passage à la ligne comme print

ouverture en écriture



The screenshot shows a code editor window titled 'fileopen2.py - /home/anne/Documents/PYTHON/fileopen2.py (3.10.12)' with a menu bar (File, Edit, Format, Run, Options, Window, Help). The code in the editor is as follows:

```
import sys

filescript, filename = sys.argv

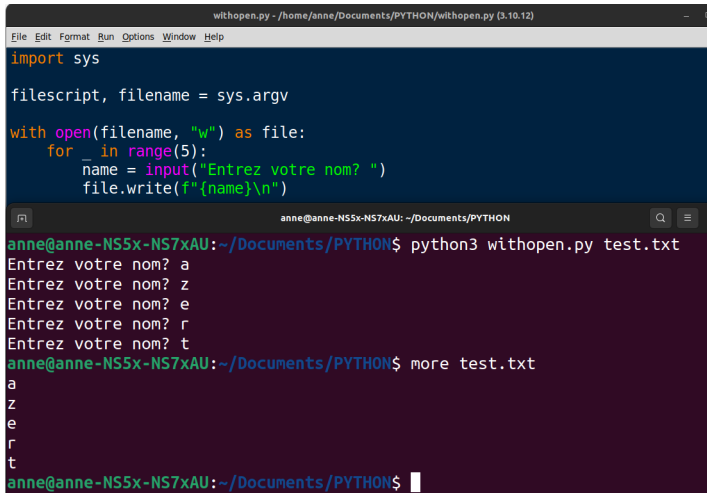
file = open(filename, "w")
for _ in range(5):
    name = input("Entrez votre nom? ")
    file.write(f"{name}\n")
file.close()
```

Below the code editor is a terminal window titled 'anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON'. It shows the execution of the script:

```
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ python3 fileopen2.py test.txt
Entrez votre nom? a
Entrez votre nom? z
Entrez votre nom? e
Entrez votre nom? r
Entrez votre nom? t
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ more test.txt
a
z
e
r
t
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$
```


context manager: with

gère la fermeture du fichier quoi qu'il arrive (exceptions)



The screenshot shows a Python IDE window titled 'withopen.py - /home/anne/Documents/PYTHON/withopen.py (3.10.12)'. The code in the editor is as follows:

```
import sys

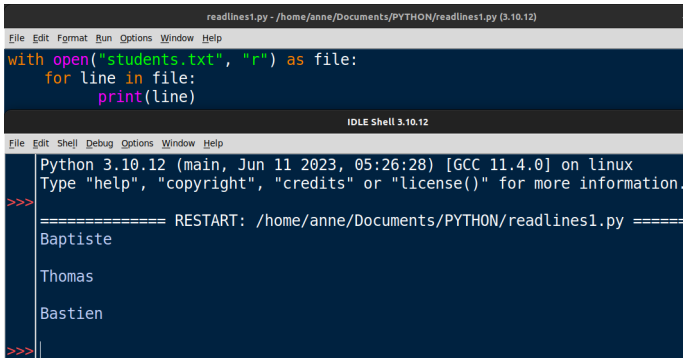
filescrip, filename = sys.argv

with open(filename, "w") as file:
    for _ in range(5):
        name = input("Entrez votre nom? ")
        file.write(f"{name}\n")
```

Below the editor is a terminal window with the prompt 'anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON'. The user runs the command 'python3 withopen.py test.txt'. The program then prompts 'Entrez votre nom?' five times, and the user enters 'a', 'z', 'e', 'r', and 't' respectively. Finally, the user runs 'more test.txt', which displays the entered names on separate lines.

```
anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON$ python3 withopen.py test.txt
Entrez votre nom? a
Entrez votre nom? z
Entrez votre nom? e
Entrez votre nom? r
Entrez votre nom? t
anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON$ more test.txt
a
z
e
r
t
anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON$
```

ouverture en lecture



The screenshot shows an IDE window titled "readlines1.py - /home/anne/Documents/PYTHON/readlines1.py (3.10.12)". The editor contains the following Python code:

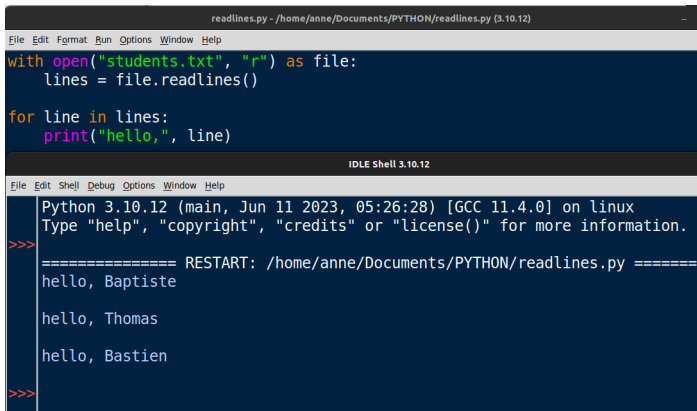
```
with open("students.txt", "r") as file:
    for line in file:
        print(line)
```

Below the editor is the "IDLE Shell 3.10.12" window. It displays the Python 3.10.12 prompt and the output of the script:

```
>>> Python 3.10.12 (main, Jun 11 2023, 05:26:28) [GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license()" for more information.
>>> ===== RESTART: /home/anne/Documents/PYTHON/readlines1.py =====
Baptiste
Thomas
Bastien
>>>
```

utilisation du fichier comme itérateur

ouverture en lecture



```
readlines.py - /home/anne/Documents/PYTHON/readlines.py (3.10.12)
File Edit Format Run Options Window Help
with open("students.txt", "r") as file:
    lines = file.readlines()

for line in lines:
    print("hello,", line)

IDLE Shell 3.10.12
File Edit Shell Debug Options Window Help
Python 3.10.12 (main, Jun 11 2023, 05:26:28) [GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license()" for more information.
>>> ===== RESTART: /home/anne/Documents/PYTHON/readlines.py =====
hello, Baptiste

hello, Thomas

hello, Bastien
>>>
```

enregistrer dans une liste les lignes du fichier

ouverture en lecture

readlines2.py - /home/anne/Documents/PYTHON/readlines2.py (3.10.12)

File Edit Format Run Options Window Help

```
with open("students.txt") as file:
    lines = file.readlines()
print("lines: ",lines)
for line in sorted(lines):
    print("Hello,", line.rstrip())
```

IDLE Shell 3.10.12

File Edit Shell Debug Options Window Help

Python 3.10.12 (main, Jun 11 2023, 05:26:28) [GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license()" for more information.

>>>

===== RESTART: /home/anne/Documents/PYTHON/readlines2.py =====

lines: ['Baptiste\n', 'Thomas\n', 'Bastien\n', 'Maxime\n', 'Alain\n']

Hello, Alain

Hello, Baptiste

Hello, Bastien

Hello, Maxime

Hello, Thomas

>>>

ouverture en lecture

readlines.py - /home/anne/Documents/PYTHON/readlines.py (3.10.12)

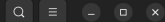
File Edit Format Run Options Window Help

```
import sys

filescript, filename = sys.argv

try:
    with open(filename, "r") as file:
        lines = file.readlines()
except FileNotFoundError:
    print(f"Le fichier '{filename}' n'existe pas")
else:
    for line in lines:
        print("Hello,", line)
```

anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON



anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON\$ python3 readlines.py students.txt

Hello, Baptiste

Hello, Thomas

Hello, Bastien

anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON\$ python3 readlines.py nimportequoi.txt

Le fichier 'nimportequoi.txt' n'existe pas

anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON\$

modes d'ouverture

https:

[//docs.python.org/fr/3/library/functions.html#open](https://docs.python.org/fr/3/library/functions.html#open)

Caractère	Signification
'r'	ouvre en lecture (par défaut)
'w'	ouvre en écriture, en effaçant le contenu du fichier
'x'	ouvre pour une création exclusive, échouant si le fichier existe déjà
'a'	ouvre en écriture, ajoutant à la fin du fichier s'il existe
'b'	mode binaire
't'	mode texte (par défaut)
'+'	ouvre en modification (lecture et écriture)

Le mode par défaut est **'r'** (ouverture pour lire du texte, un synonyme pour **'rt'**). Les modes **'w+'** et **'w+b'** ouvrent et vident le fichier. Les modes **'r+'** et **'r+b'** ouvrent le fichier sans le vider.

ouverture d'un fichier binaire

```
openbin.py - /home/anne/Documents/PYTHON/openbin.py
File Edit Format Run Options Window Help
with open("fichier1.bin", "bw") as file:
    file.write(b'hello, world!\n')

anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ cat fichier1.bin
hello, world!
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$
```

objets manipulés ne sont pas de type str mais de type bytes

ouverture d'un fichier binaire

exemple:

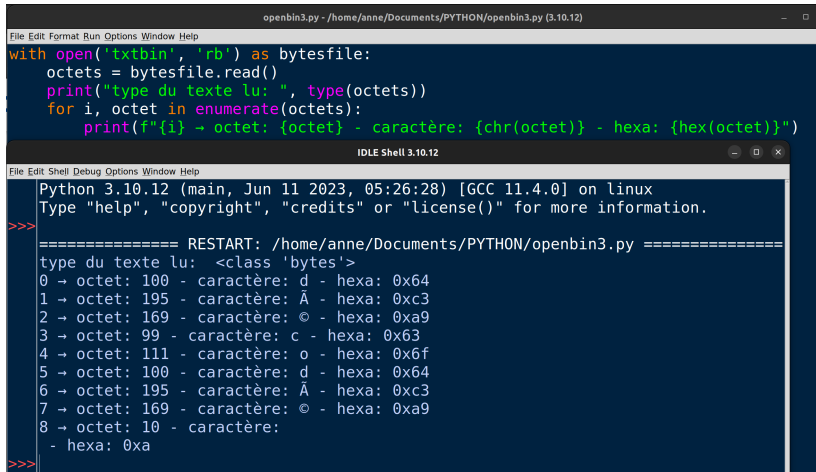
on écrit en mode texte

```
openbin2.py - /home/anne/Documents/PYTHON/openbin2.py (3.10.12)
File Edit Format Run Options Window Help
with open('txtbin', 'w', encoding='utf-8') as output:
    output.write("décodé\n")

anne@anne-NS5x-NS7xAU: ~/Documents/PYTHON
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ file txtbin
txtbin: Unicode text, UTF-8 text
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$ cat txtbin
décodé
anne@anne-NS5x-NS7xAU:~/Documents/PYTHON$
```


ouverture d'un fichier binaire

on lit en mode binaire:



The image shows two windows from the IDLE Python IDE. The top window, titled 'openbin3.py - /home/anne/Documents/PYTHON/openbin3.py (3.10.12)', contains a Python script that opens a file in binary mode ('rb') and prints the type and hexadecimal representation of each byte. The bottom window, titled 'IDLE Shell 3.10.12', shows the execution output, including a restart message and the results of the script's execution.

```
File Edit Format Run Options Window Help
with open('txtbin', 'rb') as bytesfile:
    octets = bytesfile.read()
    print("type du texte lu: ", type(octets))
    for i, octet in enumerate(octets):
        print(f"{i} → octet: {octet} - caractère: {chr(octet)} - hexa: {hex(octet)}")
```

```
Python 3.10.12 (main, Jun 11 2023, 05:26:28) [GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /home/anne/Documents/PYTHON/openbin3.py =====
type du texte lu: <class 'bytes'>
0 → octet: 100 - caractère: d - hexa: 0x64
1 → octet: 195 - caractère: Ã - hexa: 0xc3
2 → octet: 169 - caractère: © - hexa: 0xa9
3 → octet: 99 - caractère: c - hexa: 0x63
4 → octet: 111 - caractère: o - hexa: 0x6f
5 → octet: 100 - caractère: d - hexa: 0x64
6 → octet: 195 - caractère: Ã - hexa: 0xc3
7 → octet: 169 - caractère: © - hexa: 0xa9
8 → octet: 10 - caractère:
   - hexa: 0xa
>>>
```

on regarde les octets un par un

le caractère 'é' a été encodé sur 2 octets