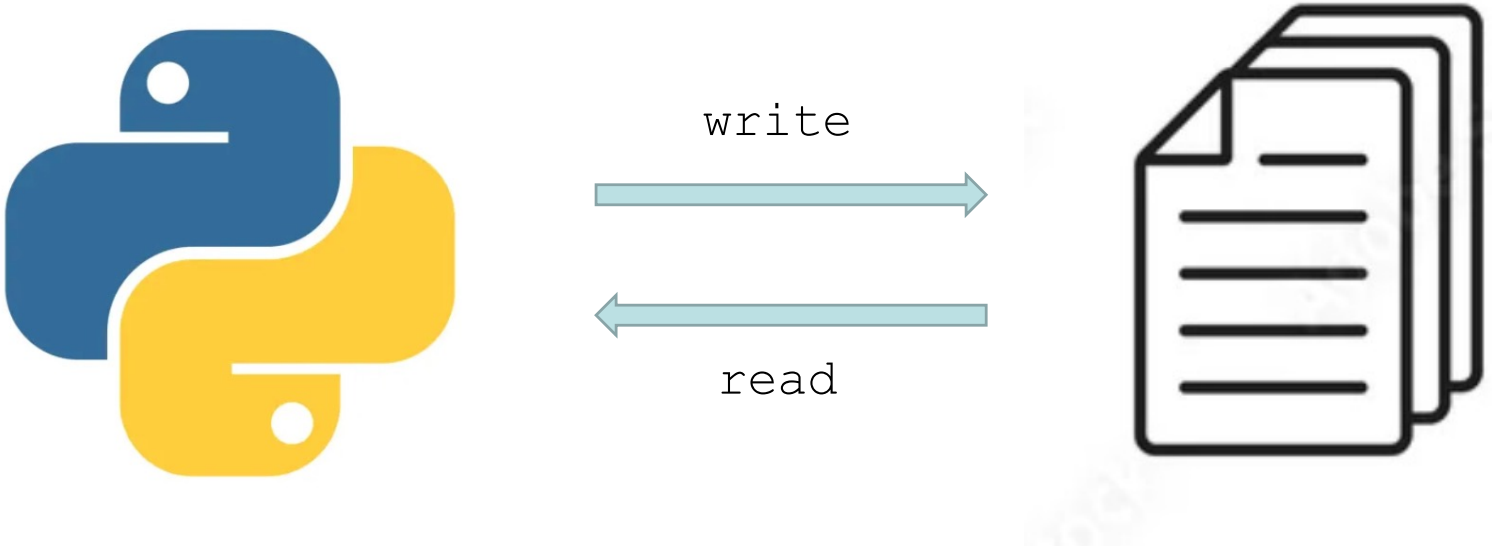
Python – Ficheiros



Python – Ficheiros

- O Python, tal como a maior parte das linguagens de programação, usa ficheiros para **guardar informação de forma persistente em memória secundária** (discos duros, pen drives, cloud, etc);

- Os programas podem também ler dados armazenados em ficheiro e não ficarem apenas dependentes da inserção de dados por parte dos utilizadores;

- É possível utilizar dois tipos de ficheiros:
  - **Ficheiros de texto**: editáveis em modo de texto a partir de outras aplicações como as aplicações do tipo bloco de notas;
  - **Ficheiros binários**: armazenam informação em código binário e não são editáveis por outras aplicações.

- Os ficheiros podem ser acedidos de três formas:
  - Acesso para leitura;
  - Acesso para escrita;
  - Acesso para leitura e escrita.

Python – Ficheiros

- Para aceder a um ficheiro, é necessário proceder à "abertura do ficheiro";
- Para abrir o ficheiro usa-se uma instrução - função `open` - especificando o nome do ficheiro e o modo de abertura;
- Após a utilização, deve-se "fechar" o ficheiro (função `close`).

Modo de Abertura	Acesso
"r"	abre o ficheiro para leitura (modo por defeito)
"w"	abre o ficheiro para escrita, apagando o conteúdo anterior
"x"	cria o ficheiro para escrita, falhando se o ficheiro já existir
"a"	abre o ficheiro para escrita, acrescentando novo conteúdo no final
"b"	abre o ficheiro em modo binário
"t"	abre o ficheiro em modo texto
"+"	abre o ficheiro para leitura e escrita

Python – Ficheiros

Acesso a ficheiros de texto

leitura:

- `read()`: lê o ficheiro na totalidade.
- `readline()`: lê o ficheiro linha a linha.
- `readlines()`: lê todas as linhas, guardando-as numa lista de strings.

escrita:

- `write()`: escreve linha a linha.
- `writelines()`: escreve um conjunto de linhas.

Python – Ficheiros

Leitura linha a linha de um ficheiro de texto (1)

```
i = 0
f = open('readme.txt')

line = f.readline()
while line:
    print( line )
    i += 1
    line = f.readline()

f.close()
```

abertura do ficheiro `readme.txt` (por defeito, é aberto para leitura)

as linhas do ficheiro são lidas e processadas uma a uma

no final, deve-se fechar o ficheiro

Leitura linha a linha de um ficheiro de texto (2)

```
i = 0
with open('readme.txt') as f:
    line = f.readline()
    while line:
        print( "linha", i, " - ", line )
        i += 1
        line = f.readline()
```

pode-se usar esta instrução para abrir ficheiros de forma segura

- a instrução `with ...as` abre o ficheiro de forma segura;
- a instrução `with ...as` fecha o ficheiro no final;
- a instrução `f.close()` não é necessária quando se usa `with ...as`.

Leitura linha a linha de um ficheiro de texto (3)

```
with open('readme.txt') as f:
    for line in f:
        print(line.strip())
```

- a variável `f` pode ser percorrida como se fosse uma lista usando um ciclo `for`;
- `f` é um objeto iterável, ou seja, pode conter 0, 1 ou mais elementos, que são acedidos um a um;
- a instrução `line.strip()` retira a mudança de linha existente em cada linha lida (ver diferenças no uso de `print(line.strip())` e de `print(line)`);

Python – Ficheiros

Leitura de um ficheiro de texto na totalidade – `read()`

```
with open('readme.txt') as f:
    text = f.read()
print(text)
```

Leitura de um ficheiro de texto guardando as linhas numa lista – `read()`

```
with open('readme.txt') as f:
    lines = f.readlines()

for line in lines:
    print(line)
```

ou

```
for i in range(0, len(lines) ):
    print( lines[i] )
```

permite verificar que `lines` é uma lista

Python – Ficheiros

Escrever num ficheiro de texto usando a função `write()`

```
lines = ['escrever em ficheiro', 'usando funcao write']
with open('writeme.txt', 'w') as f:
    for line in lines:
        f.write(line)
        f.write('\n')
```

No caso de existirem caracteres UTF-8

```
lines = ['escrever em ficheiro', 'usando função write']
with open('writeme.txt', 'w', encoding='utf-8') as f:
    for line in lines:
        f.write(line)
        f.write('\n')
```

Python – Ficheiros

Escrever num ficheiro de texto usando a função `writelines()`

```
lines = ['escrever em ficheiro', 'usando writelines']
with open('writeme.txt', 'w') as f:
    f.writelines(lines)
```

Para colocar cada elemento da lista numa linha (usando a função `join`)

```
lines = ['escrever em ficheiro', 'usando write']
with open('writeme.txt', 'w') as f:
    f.write('\n'.join(lines))
```

(função `join` - https://www.w3schools.com/python/ref_string_join.asp)

Python – Ficheiros

Acrescentar conteúdo a um ficheiro

- Ao abrir o ficheiro com a opção `w`, se o ficheiro já existir, o seu conteúdo é apagado;
- Para acrescentar conteúdo a um ficheiro já existente: abrir com a opção `a` (`append`);
- O novo conteúdo é colocado no fim do ficheiro, após o conteúdo já existente.

```
lines = ['escrever em ficheiro', 'usando write']
with open('writeme.txt', 'w') as f:
    f.write('\n'.join(lines))

lines = ['acrescentando linhas', 'a ficheiro existente']
with open('writeme.txt', 'a') as f:
    f.write('\n')
    f.write('\n'.join(lines))
```

Python – Ficheiros

Exemplo – ler linhas de um ficheiro de texto em que cada linha contém vários campos de informação

```
with open('alunos.txt') as f:
    for line in f:
        lista = []
        lista = line.strip().split(':')
        print( "Nome  : ", lista[0] )
        print( "Curso : ", lista[1] )
        print( "Morada: ", lista[2], '\n' )
```

Cada linha do ficheiro contém informação sobre um aluno, dividida em três campos: nome, curso e morada

```
Pedro:informatica:porto
Maria:mecanica:gaia
Paulo:eletronica:espinho
```

alunos.txt

Python – Ficheiros

Exemplo – ler linhas de um ficheiro de texto em que cada linha contém vários campos de informação

```
Pedro:informatica:porto
Maria:mecanica:gaia
Paulo:eletronica:espinho
```

alunos.txt

```
Nome   :  Pedro
Curso  :  informatica
Morada:  porto

Nome   :  Maria
Curso  :  mecanica
Morada:  gaia

Nome   :  Paulo
Curso  :  eletronica
Morada:  espinho
```

O programa lê e processa a informação existente no ficheiro, guardando numa lista os campos existentes em cada linha

Python – Ficheiros

Ler e escrever num ficheiro binário

```
# escrever bytes num ficheiro binario
with open("bytes.dat", "wb") as f:
    f.write(b'\x00\xFF')

# ler bytes de um ficheiro binario
with open("bytes.dat", "rb") as f:
    content = f.read()
    print(content)
```

O ficheiro `bytes.dat` não é editável por outras aplicações (por exemplo, editores de texto) ao contrário do que acontece com os ficheiros de texto.

Python – Ficheiros

Referências

- https://www.pythontutorial.net/python-basics/python-read-text-file/
- https://www.pythontutorial.net/python-basics/python-write-text-file/
- https://medium.com/@niraj.e21/python101-python-file-i-o-a-comprehensive-guide-5141a8f4a402
- https://www.w3schools.com/python/ref_string_join.asp
- https://www.w3schools.com/python/ref_string_split.asp