Instituto Politécnico da Guarda
Escola Superior de Tecnologia e Gestão

Manual
Algoritmos e Programação em Python

6 a 10 de novembro

16 de Outubro de 2018

Estruturas (records)
Classes
Ficheiros binários (Binary files)
Ficheiros de texto (Text files)

Curso: Engenharia Informática
Unidade Curricular:
Algoritmos e Estruturas de Dados
Algorithms and Data Structures
Ano Letivo: 2018/2019
Docente: Paulo Jorge Costa Nunes
Coordenador da área disciplinar: Noel Lopes

Conteúdo

Capítulo 1

Orientação tutorial

1.1 Python

1. Python list — [1]
2. Python set — [2]
3. Python tuple — [3]
4. Ptyhon classes — [4]
5. Ficheiros — [5]
6. Módulos — [6]
7. **EduMaven / Python Programming** (879 pages) — `https://edumaven.com/python-programmi`

1.2 Python: tuple / list / set

Tuple

- A tuple is a fixed-length immutable list. It cannot change its size or content.
- A tuple is denoted with parentheses: (1,2,3)

List

- Elements of a list can be changed via their index or via the list slice notation.
- A list can grow and shrink using append and pop methods or using the slice notation.
- A list is denoted with square brackets: [1, 2, 3]

Fonte: [7]

[1] `https://www.tutorialspoint.com/python/python_lists.htm`
[2] `https://www.tutorialspoint.com/python/python_sets.htm`
[3] `https://www.tutorialspoint.com/python/python_tuples.htm`
[4] `https://www.tutorialspoint.com/python/python_classes_objects.htm`
[5] `https://www.tutorialspoint.com/python/python_files_io.htm`
[6] `https://www.tutorialspoint.com/python/python_modules.htm`
[7] `https://edumaven.com/python-programming/tuple`

```
1  tup1 = ('physics', 'chemistry', 1997, 2000);
2  tup2 = (1, 2, 3, 4, 5 );
3  tup3 = "a", "b", "c", "d";
4
5  print(tup1)
6  print(tup2)
7  print(tup3)
8
9  print(tup1 + tup2 + tup3)
10
11 # lists
12 a = [1, 2, 3]
13 b = [4, 5, 3]
14
15 print (a)
16 print (b)
17
18 # sets
19 a= set(a)
20 b= set(b)
21 print (a.union(b))
22 print (a.difference(b))
23
24 >>>
25 ('physics', 'chemistry', 1997, 2000)
26 (1, 2, 3, 4, 5)
27 ('a', 'b', 'c', 'd')
28 ('physics', 'chemistry', 1997, 2000, 1, 2, 3, 4, 5, 'a', 'b', 'c', 'd')
29 [1, 2, 3]
30 [4, 5, 3]
31 {1, 2, 3, 4, 5}
32 {1, 2}
```

Listing 1.1: Tuple, list and set

1.3 Classes e ficheiros de texto

1.3.1 Classes

"**Class** — A user-defined prototype for an object that defines a set of *attributes* that characterize any object of the class. The attributes are data members (class variables and instance variables) and methods, accessed via dot notation".

Class variable — A variable that is shared by all instances of a class. Class variables are defined within a class but outside any of the class's methods. Class variables are not used as frequently as instance variables are."Fonte: [8]

[8] https://www.tutorialspoint.com/python/python_classes_objects.htm

1.3.2 Example

```
def WriteCars():
    # fname = input("Enter filename: ")
    fname = 'carros.txt'
    outfile = open(fname, 'at')
    print('BMW', "00-AA-23", 'Blue', file=outfile, sep=';')
    outfile.close()

def ReadAllCars():
    #fname = input("Enter filename: ")
    fname = 'carros.txt'
    infile = open(fname,'r')
    data = infile.read()
    print(data)
    return data;

WriteCars()
ReadAllCars()
print (Employee.__doc__)
```

Listing 1.2: Escrever e ler ficheiro de texto

```
class Employee:
    'Common base class for all employees'
    def __init__(self):
        self.name = ''
        self.sex = ''
        self.age = 0
        self.salary = 0

    def displayEmployee(self):
        print ("%-6s : %s"   % ("Name", self.name))
        print ("%-6s : %s"   % ("Sex ", self.sex))
        print ("%-6s : %i"   % ("Age", self.age))
        print ("%-6s : %.2f" % ("Salary", self.salary))

emp = Employee()
emp.name = "Carlos Santos"
emp.sex  = "Male"
emp.age = 18
emp.salary =  2000

print(emp.salary)
emp.displayEmployee()
```

Listing 1.3: Definição da classe *Employees*

```
def GetUserData():
    emp = Employee()
    emp.name = input("Name ? ")
    emp.sex = input("Sex (Male/Female) ?")
    emp.age = int(input("Age ?"))
    emp.salary = float(input("Salary ?"))
    return emp;

def InsertEmployee2(nome_ficheiro):
    print ('Insert Employee')
    emp = GetUserData()
    f = open(nome_ficheiro, "at");
    print(emp.name, emp.sex, emp.age ,emp.salary, file=f,sep=';')
    f.close()

def InsertEmployee(nome_ficheiro):
    print ('Insert Employee')
    emp = Employee()
    emp.name = "Carlos Santos"
    emp.sex = "Male"
    emp.age = 18
    emp.salary = 2000
    f = open(nome_ficheiro, "at");
    print(emp.name, emp.sex, emp.age ,emp.salary, file=f,sep=';')
    f.close()

def ListEmployees(nome_ficheiro):
    print ('List of Employees')
    infile = open(nome_ficheiro, "rt")
    #Python treats the file itself as a sequence of lines!
    for line in infile:
        # process the line here
        line = line.rstrip("\n")
        print (line);
    infile.close()

def ListEmployees2(nome_ficheiro):
    print ('List of Employees')
    emp = Employee()
    infile = open(nome_ficheiro, "rt")
    #Python treats the file itself as a sequence of lines!
    for line in infile:
        # process the line here
        line = line.rstrip("\n")
        emp.name, emp.sex, emp.age, emp.salary = line.split(';')
        print(emp.name, emp.sex, emp.age, emp.salary)
    infile.close()

def menu():
    import os
    os.system('cls')
    nome_ficheiro = 'Employers.txt'
    while True:
        print ('{0:^60}'.format('Management of Employees'))
        print ('\n\n')
        s = 10
        print('1 - Inserir   | Insert')
        print('2 - Inserir 2 | Insert 2')
        print('3 - Listar    | List')
        print('4 - Listar 2  | List 2')
        print('0 - Terminar  | Exit')
        print();
        op = input ("? ")
        if op == '0':
            break
        elif (op == '1'):
            InsertEmployee(nome_ficheiro)
        elif (op == '2'):
            InsertEmployee2(nome_ficheiro)
        elif (op == '3'):
            ListEmployees(nome_ficheiro)
        elif (op == '4'):
            ListEmployees2(nome_ficheiro)
        else:
            print ('Deve escolher uma da opção da lista')
            time.sleep(1)
        os.system('cls')
menu()
```

Listing 1.4: Menu, inserir e listar Empregados (Employees)

Bibliografia