



Python II


1

2

3

4

5



Arquivos

Módulos e Scripts
Orientação a Objetos
Exercício
Leituras Recomendadas

1

2

3

4

5

Manipulando Arquivos

- Similar a C, tanto nas funções como nos modos de acesso.
- Abrir com `file(<file>, <mode>)`
 - Modos de
 - leitura ('r')
 - escrita ('w')
 - append ('a')
 - Modo binário ('b')
 - Modo misto com '+' (leitura e escrita)

```
- open/file
- close
- write
- read
- readline
- write
- writeline
- seek
- tell
```

Manipulando Arquivos

- Abrindo um arquivo para leitura

```
open('foobar.txt', 'r')
```

- Abrindo um arquivo para escrita
(APAGA O CONTEÚDO DO ARQUIVO!!!)

```
open('foobar.txt', 'w')
```

Manipulando Arquivos

- Uso comum: iterando linhas:

```
arq = open('foobar.txt', 'r')  
  
for linha in arq:  
    print linha  
  
arq.close()
```

O objeto **file** é seu próprio iterador. A cada iteração do **for**, o método **next** de file é chamado.

- Removendo o '\n' no final da linha: **line.strip()**
- **close()** é opcional (chamado automaticamente quando é destruído)

Exercício

- Abra um arquivo de texto **A** para leitura
- Crie (para escrita) um novo arquivo **B**
- Escreva o conteúdo do arquivo **A** no arquivo **B**, intercalando suas linhas com linhas em branco

– Exemplo:


A

```
Linha 1  
Linha 2  
Linha 3  
Linha Direta
```



B

```
Linha 1  
  
Linha 2  
  
Linha 3  
  
Linha Direta
```



Arquivos

Módulos e Scripts

Orientação a Objetos

Exercício

Leituras Recomendadas

1

2

3

4

5

Módulos

- Arquivo contendo definições e comandos python (scripts...)
- Podem ser em Python ou nativos (C/C++)
- Importando:

```
import nomedomodulo  
import nomedomodulo as mod  
from nomedomodulo import foobar
```

- Módulo 'principal' tem o atributo `__name__` como “`__main__`”

Módulos

- Módulo `sys` – Informações do sistema
 - `sys.exit(0)` – Sai do interpretador ou programa
 - `sys.argv` – Lista de argumentos passados para o programa

`args.py`

```
import sys

print sys.argv
for arg in sys.argv:
    print arg
```

Execute na linha de comando:

```
python args.py argumento1 argumento2
```



Exercício

- Modifique o programa de cópia de arquivos para receber os nomes dos arquivos de origem e de destino pela linha de comando.



Arquivos
Módulos e Scripts
Orientação a
Objetos
Exercício
Leituras Recomendadas

1

2

3

4

5

Classes

Automovel

+ placa : str

__init__(str) : None

get_placa() : str

dirigir(int) : None

métodos

```
class Automovel:
```

construtor

```
    def __init__(self, placa='XX-123'):  
        self.placa = placa
```

```
    def get_placa(self):  
        return self.placa
```

self

```
    def dirigir(self, velocidade):  
        print 'Estou dirigindo a %d' \  
              ' km/h' % velocidade
```

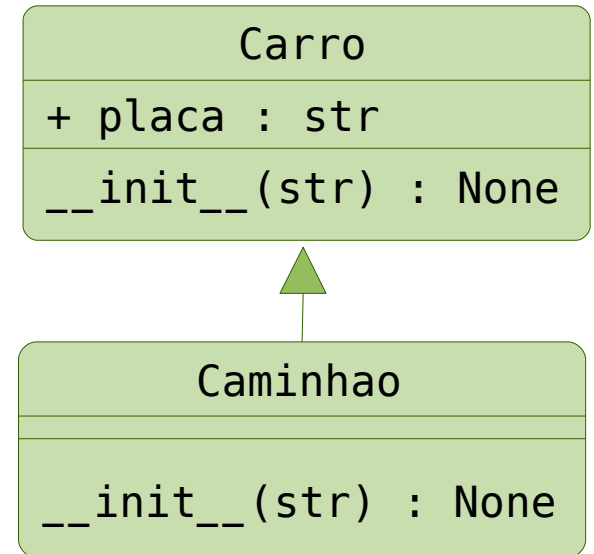
Herança

- Java

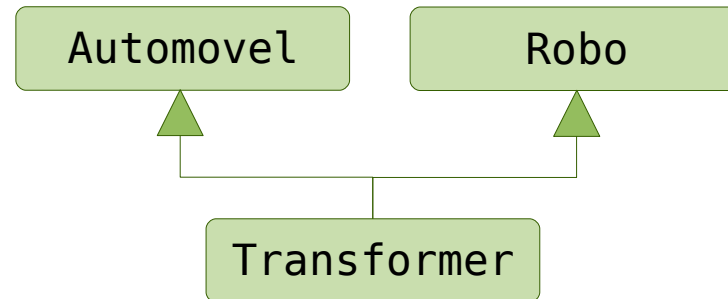
```
public class Caminhao extends Carro {  
    public Caminhao(String placa) {  
        super(placa);  
    }  
}
```

- Python

```
class Caminhao (Carro):  
    def __init__(self, placa):  
        Carro.__init__(self, placa)
```



Herança Múltipla



- C++

```
class Transformer : public Automovel, public Robo {
    ...
};
```

- Python

```
class Transformer(Automovel, Robo):
    ...
```




Arquivos
Módulos e Scripts
Orientação a Objetos
Exercício
Leituras Recomendadas

1

2

3

4

5

Exercício

1. Crie uma classe **Ponto** conforme diagrama ao lado, salve o código num arquivo de nome **ponto.py**

```
class Ponto:  
    + nome : str  
    + x : int  
    + y : int  
    __init__(nome, x, y) : None  
    __str__() : str
```

- O método **__str__** retorna uma string se alguém der um “**print objeto**”. Faça com que mostre os dados do objeto no formato: “**Nome: (x, y)**”

2. Crie outro script que importe **ponto**, e leia um arquivo contendo informações sobre vários pontos, criando um objeto **Ponto** para cada entrada lida.

Exercício (cont.)

3. Coloque cada objeto Ponto numa lista.
4. Imprima cada elemento da lista.

`pontos.txt`

```
A
100 200
B
130 150
C
500 239
OutroPonto
199 54
```


Nota Sobre List Comprehensions

- **List Comprehensions** criam listas a partir de listas, de forma análoga a uma expressão matemática.

```
A = {0, 2, ... 99}
```

```
S = {2x | x A, x > 10 e x < 50}
```



```
A = range(100)
```

```
S = [2*x for x in A if x > 10 and x < 50]
```



Arquivos

Módulos e Scripts

Orientação a Objetos

Exercício

Leituras

Recomendadas

1

2

3

4

5

Leituras Recomendadas

- **Workshop de Python no CIn, parte 1**
 - http://www.cin.ufpe.br/~cinlug/wiki/index.php/Workshop_de_Python_no_CIn%2C_parte_1
- **File Management in Python**
 - <http://www.devshed.com/c/a/Python/File-Management-in-Python/>
- **Python Library Reference** (em Javanês: “a API de Python”)
 - <http://docs.python.org/lib/lib.html>
- **Curso de Python em 5 Horas**
 - <http://gustavobarbieri.com.br/palestras/python-5hs/>
- **Resumo Python**
 - http://www.gustavobarbieri.com.br/python/aulas_python/resumo.pdf



CinLUG 

<http://www.cinlug-br.org>



•Atribuição-Compartilhamento pela mesma licença 2.5 (<http://creativecommons.org/licenses/by-sa/2.5/deed.pt>)

•Você pode:

- copiar, distribuir, exibir e executar a obra
- criar obras derivadas
- fazer uso comercial da obra

•Sob as seguintes condições:



•**Atribuição.** Você deve dar crédito ao autor original, da forma especificada pelo autor ou licenciante.



•**Compartilhamento pela mesma Licença.** Se você alterar, transformar, ou criar outra obra com base nesta, você somente poderá distribuir a obra resultante sob uma licença idêntica a esta.

- Para cada novo uso ou distribuição, você deve deixar claro para outros os termos da licença desta obra.
- Qualquer uma destas condições podem ser renunciadas, desde que Você obtenha permissão do autor.
- Qualquer direito de uso legítimo (ou "fair use") concedido por lei, ou qualquer outro direito protegido pela legislação local, não são em hipótese alguma afetados pelo disposto acima.

•*Este é um sumário para leigos da Licença Jurídica*

(na íntegra: <http://creativecommons.org/licenses/by-sa/2.5/br/legalcode>).

Termo de exoneração de responsabilidade:

<http://creativecommons.org/licenses/disclaimer-popup?lang=pt>