

GERENCIAMENTO DE ARQUIVOS

MÓDULO `os`

O módulo `os` oferece um conjunto de métodos para gerenciamento de arquivos e diretórios de forma independente de plataforma.

Os métodos funcionam de forma semelhante em diferentes sistemas operacionais, desde que a funcionalidade exista.

Caso a funcionalidade não exista a chamada gera uma exceção `OSERROR`.

O sistema operacional pode ser identificado através da seguinte chamada:

```
import os  
os.name
```

Os métodos disponíveis em `os` focam predominantemente a manipulação de um arquivo ou diretório de cada vez.

Complementarmente, o módulo `shutil` oferece funções para manipulação de diversos arquivos ou diretórios em uma única operação.

Métodos e Chamadas Equivalentes em Linux (MAC)

a) Retorna o diretório atual (WORKING) na forma de string:

```
os.getcwd()
```

Linux: `pwd`

b) Retorna o conteúdo do diretório atual:

```
os.listdir()
```

Linux: `ls`

c) Altera o diretório atual:

```
os.chdir('/caminho/absoluto' ou 'caminho/relativo')
```

Linux: `cd /caminho/absoluto` ou `cd caminho/relativo`

d) Cria um caminho adicionando um subdiretório:

```
os.path.join(path, sub1, sub2, ...)
```

e) Cria um diretório:

```
os.makedirs('dir/subdir')
```

Linux: `mkdir -p`

f) Copia um arquivo ou diretório:

```
shutil.copy2('caminho_arquivo_origem', 'caminho_diretorio_destino')
```

Linux: `cp caminho_arquivo_origem caminho_diretorio_destino/.o`

g) Move um arquivo ou diretório:

```
shutil.move('caminho_arquivo_origem', 'caminho_diretorio_destino')
```

Linux: `mv caminho_arquivo_origem caminho_diretorio_destino/.`

* Também usado para renomear arquivos no Linux

h) Remove um arquivo:

```
os.remove('caminho_arquivo')
```

Linux: `rm caminho_arquivo`

i) Remove um diretório e todos os arquivos e subdiretórios dentro dele:

```
shutil.rmtree('caminho_diretorio')
```

Linux: `rm -rf caminho_arquivo`

EXERCICIO 1: Método os.walk() - navegação em diretórios e subdiretórios

1) Criar a pasta EXERCICIO1 e copiar os arquivos exercicio01.py e script01.py para dentro dela. O arquivo exercicio01.py implementa a chamada do método os.walk(). Executar o script exercicio01.py e observar as mensagens no console.

```
import os
for root, dirs, files in os.walk(".", topdown=False):
    for name in files:
        print(os.path.join(root, name))
    for name in dirs:
        print(os.path.join(root, name))
```

2) Crie os seguintes subdiretórios na pasta EXERCICIO1: TESTE1, TESTE2. Crie mais um subdiretório TESTE21 dentro de TESTE2 e coloque um arquivo qualquer na pasta.

Dica: Executar o script01.py.

Executar novamente o script exercicio01.py e observar as mensagens no console.

3) Altere o parâmetro topdown=True no script exercicio01.py.

Executar novamente o script exercicio01.py e observar as mensagens no console.

Executar o script exercicio01.py iniciando a navegação das pastas a partir de TESTE2 e observar as mensagens no console.

Dica: Você pode inserir o comando os.chdir('TESTE2') no script exercício01.py antes do for.

Questão 1

a) Copiar as mensagens observadas em (2).

b) Copiar as mensagens observadas em (3).

EXERCICIO 2

1) Criar a pasta EXERCICIO2 e copiar o arquivo exercicio02.py para dentro dela. Dentro desta pasta crie também um arquivo TEXTO.txt com qualquer conteúdo.

2) Alterar o script exercicio02.py para realizar as seguintes operações.

- Imprimir o diretório de trabalho corrente
- Criar os subdiretórios SUBDIR1 e SUBDIR2
- Copiar o arquivo TEXTO.txt para o SUBDIR1
- Mover o arquivo TEXTO.txt do SUBDIR1 para o SUBDIR2
- Apagar o SUBDIR1 e tudo o que estiver dentro

Executar o script exercicio02.py.

Dica: Caso precise executar mais de uma vez, verifique se o diretório ou arquivo já está criado, e criar apenas se não existir.

```
if not os.path.exists('SUBDIR1'):
    os.makedirs('SUBDIR1')
```

3) Executar o script exercicio01.py iniciando a navegação das pastas a partir de EXERCICIO2 e observar as mensagens no console.

Dica: Não esquecer tirar o comentário do comando os.chdir.

Questão 2

a) Copiar o script exercicio02.py com os comandos solicitados

b) Copiar as mensagens observadas em (3)