

Fundamentos de Programação

1º Semestre
2023/2024

Aulas 21 e 22



Plano

- Manipulação de Ficheiros
- Classes e objetos (novas tarefas)

Classes de E/S (Entrada e Saída)

- Principais grupos de classes de E/S

Tipo	Entrada	Saída
Binário	<code>pickle.load()</code>	<code>pickle.dump()</code>
Texto	<code>read()</code>	<code>write()</code>



Um ficheiro binário é lido e escrito por programas em execução
Um ficheiro de texto é lido e escrito por editores de texto

Classes de E/S (Entrada e Saída)

- Classes de E/S mais utilizadas


Permitem guardar e recuperar objetos (e outros dados binários)

Tipo	Entrada	Saída
Binário	<code>pickle.load()</code>	<code>pickle.dump()</code>
Texto	<code>read()</code>	<code>write()</code>

Permitem ler e escrever texto

Ler a partir de um Ficheiro (1)

```
def __dados_teste(self):  
    # 1º Localização do ficheiro que vai ser lido  
    f = open("entrada.txt", "r")
```



Abrir em modo leitura
Outras opções:
r – leitura
w – escrita
rb – leitura binário
wb – escrita binário

Ler a partir de um Ficheiro (2)

```
def __dados_teste(self):  
    # 1º Localização do ficheiro que vai ser lido  
    f = open("entrada.txt", "r")  
    # 2º Leitura do ficheiro, linha a linha até EOF  
    for line in f:  
        print(line)
```

Ler a partir de um Ficheiro (3)

```
def __dados_teste(self):  
    # 1º Localização do ficheiro que vai ser lido  
    f = open("entrada.txt", "r")  
    # 2º Leitura do ficheiro, linha a linha até EOF  
    for line in f:  
        print(line)  
    # 3º Fechar o ficheiro para libertar o handle do Sistema Operativo  
    f.close()
```

Ler a partir de um Ficheiro (4)

Utiliza-se frequentemente o código de forma condensada:

```
with open("loja.pickle", "wb") as outfile:
```

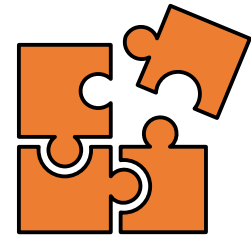
A instrução “with” abre o ficheiro, permite a sua utilização com o nome definido e garante, no final, que é realizada a respetiva libertação de recursos

[Built-in Functions — Python 3.11.0 documentation](#)

Tarefa 19

Trabalhar sobre o seguinte documento:

- **Tarefa 19 – Leitura de ficheiros.pdf**



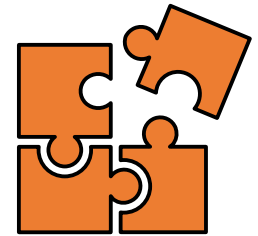
Escrever a partir de um Ficheiro

```
def mostrar(self):  
    f = open("saidas.txt", "w")  
    f.write("Texto a escrever no ficheiro de saida")  
    f.close()
```

Tarefa 20

Trabalhar sobre o seguinte documento:

- **Tarefa 20 – Leitura e escrita de ficheiros.pdf**



Serialização e Deserialização

- Processo para converter objetos (instâncias de classes) em sequências de *bytes* e de sequências de bytes em objetos novamente
- Útil para manter o estado de um objeto quando temos de desligar a aplicação – tudo o que estiver em RAM, resultante do normal funcionamento da aplicação, será perdido
 - Solução: Guardar em ficheiro e recuperar a informação quando necessário (normalmente no arranque da aplicação)

Serialização e Deserialização

- Utilizam-se os formatos JSON ou binário

```
import pickle
```

```
def __serialize(self):  
    with open("loja.pickle", "wb") as outfile:  
        pickle.dump(self, outfile)
```

```
def __deserialize(self):  
    with open("loja.pickle", "rb") as infile:  
        pequena_loja = pickle.load(infile)  
        pequena_loja.mostrar()
```

Objeto a seriar

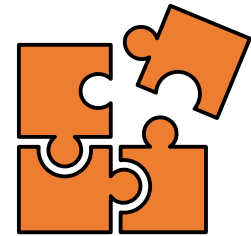
Instância deseriada

[pickle — Python object serialization — Python 3.11.0 documentation](#)

Tarefa 21

Trabalhar sobre o seguinte documento:

- **Tarefa 21 – Ficheiros e Classes.pdf**



Comentários?

