

Arquivos

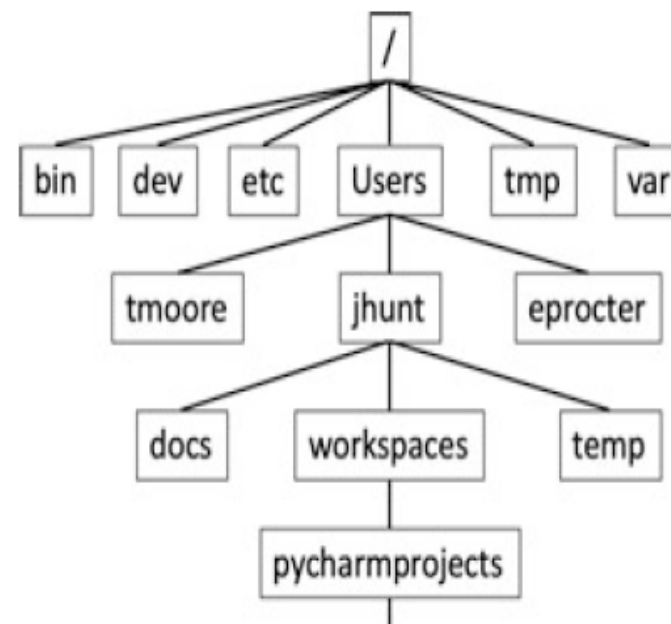


Arquivos

- A linguagem python possui diversos recursos para acessar arquivos.
- Alguns destes recursos são nativos e outros estão disponíveis em pacotes adicionais, como, por exemplo Pandas.
- A necessidade de acessar dados em arquivos é comum em diversas aplicações.
- Operações de leitura e gravação de arquivos, tradicionalmente denominadas File I/O permitem manter dados em meio estável.

Locais

- No sistema operacional os arquivos estão armazenados em locais, pastas, diretórios.
- O acesso a um arquivo é feito pela definição do caminho completo, incluindo o nome do arquivo.
- Ex. J. Hunt, Advanced
- Guide to Python 3
- Programming





Atributos

- Cada arquivo possui diversos atributos, tais como:
 - Proprietário
 - Tipo: Texto, doc, etc
 - Tamanho
 - Datas de criação, última alteração
 - Se pode ser lido, modificado, etc.
 - Local do arquivo



Textos ASCII

- O formato texto é um dos mais utilizados para leitura e gravação de dados.
- São arquivos que podem ser lidos e gravados usando qualquer linguagem de programação.
- No formato ASCII cada caractere é associado a um código numérico.
- Formatos Unicode (UTF-8, UTF-16) ampliam a capacidade de representação do ASCII.
- Formatos binários apresentam uma representação natural para linguagens, mas não para seres humanos.



Acesso sequencial x randômico

- Leitura e gravação de dados em arquivos pode ser feita utilizando acesso sequencial ou randômico.
- No acesso sequencial os dados são lidos na ordem que estão armazenados, do início até encontrar o fim do arquivo.
- No acesso randômico o programa pode ler qualquer parte do arquivo, em qualquer momento.
- Neste caso é possível se deslocar até um ponto do arquivo e fazer uma leitura de dados a partir deste ponto.



File I/O - python

- Python suporta gravação e leitura em diferentes formatos, além de texto.
- Alguns formatos permitem a compactação de dados, como, por exemplo Pickles.
- Outros, como XML apresentam uma estrutura hierárquica e demandam o uso de API específicas para a sua manipulação.
- O tamanho é influenciado diretamente pelo formato do arquivo.
- O desempenho também varia, sendo melhor em formatos binários.



File I/O - python

- O acesso a um arquivo inicia pela criação de um objeto usando a função `open()`.
- `file_object = open(file_name, access_mode, buffering)`
- O nome do arquivo deve incluir o caminho completo ou relativo, caso ele se encontre em uma pasta local.
- Algumas configurações de modo de acesso:
 - `w` – Escrita
 - `r` – Leitura
 - `rt` ou `wt` – Leitura e escrita de arquivo texto.
 - `rb` ou `wb` – Leitura e escrita de arquivo binário
 - `wb+` ou `rb+` - Sobreescreve arquivo existente ou cria novo



File I/O - python

- Existem diversos métodos e propriedades para leitura e gravação, como, por exemplo:
 - Readlines(), read(), readline() - Leitura das linhas do arquivo.
 - close() - Fecha o arquivo.
 - Write() - Grava dados no arquivo.
 - File.name, file.mode, file.closed() - Dados do arquivo



File I/O - python

- É possível iterar pelas linhas de um arquivo, com repetição for.
- Quando o arquivo texto é aberto o ponteiro é posicionado na linha inicial.
- Após fazer a leitura das linhas, o ponteiro está no final do arquivo, portanto, se for feita nova iteração nada será mostrado ou carregado.
- O conteúdo de cada linha pode ser quebrado usando método split disponível em strings.
- Desta forma é possível armazenar os dados em estruturas de memória.



File I/O - python

- A manipulação de arquivos é facilitada pelo uso de diversos pacotes e APIs.
- O pacote Pandas é um dos recursos mais usados por profissionais que utilizam python para análise de dados.
- Pandas possui diversos métodos para carregar dados para a memória, como `read_csv`, `read_excel`, etc.
- Dados são carregados para objetos da classe `DataFrame`.
- Além disso a gravação de um arquivo pode ser realizado usando um método da classe `DataFrame`, como `to_csv`, `to_excel`.



Desafio

- Ler os dados do arquivo denominado 'precos_imoveis.csv'.
- Carregar os dados e armazenar na memória (listas, dicionários, tuplas, etc).
- Mostrar as seguintes informações:
 - Dados do imóvel mais caro e do mais barato.
 - Preço médio dos imóveis
 - Contagem e preço médio dos imóveis por bairro.