Relatório 4 - Prática: Principais Bibliotecas e Ferramentas Python para Aprendizado de Máquina (I)

João Pedro Gomes

1. Introdução

O objetivo do card 4, foi estudar sobre, Jupyter Notebook, Python para análise com numpy e pandas e realizar exercícios que estavam no docs que tinha mas mesmas aulas e também realizar a prática dos dois temas e fazer um relatório.

2. Desenvolvimento

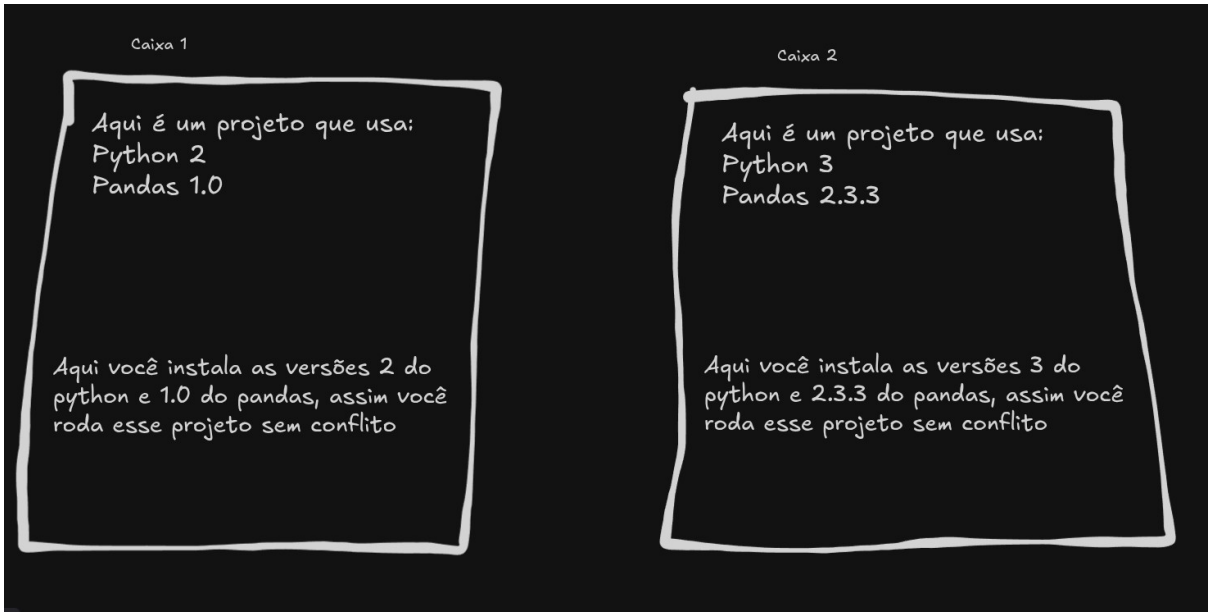
O card é divido em 3 seções para estudo, Jupyter Notebook, Numpy e Pandas, eu vou falar sobre as 3 e depois explicar a minha prática com o pandas.

Jupyter Notebook:

O jupyter notebook é uma ferramenta que você baixa no pc pra rodar codigo python através do navegador, ele funciona com um esquema de célula ao inves de ter várias linhas igual em um vscode, você tem uma célula que pode ter vários codigos dentro dela e depois rodar ela sozinha podendo ter várias célular com diferentes códigos em cada uma, ao rodar ele exibe o resultado do código abaixo da célula, indo por partes por assim dizer.

Ainda na seção do jupyter ele fala sobre ambientes virtuais, é tipo como se fosse uma caixa separada onde fica cada projeto com suas versões de bibliotecas, python e etc..., exemplo você tem um projeto que usa pandas 1.0, e no outro pandas 2.3.3, aí na caixa que usa pandas 1.0 vocẽ instala o pandas 1.0 e na outra o pandas 2.3.3, assim não da conflito de versão.

Um exemplo visual:

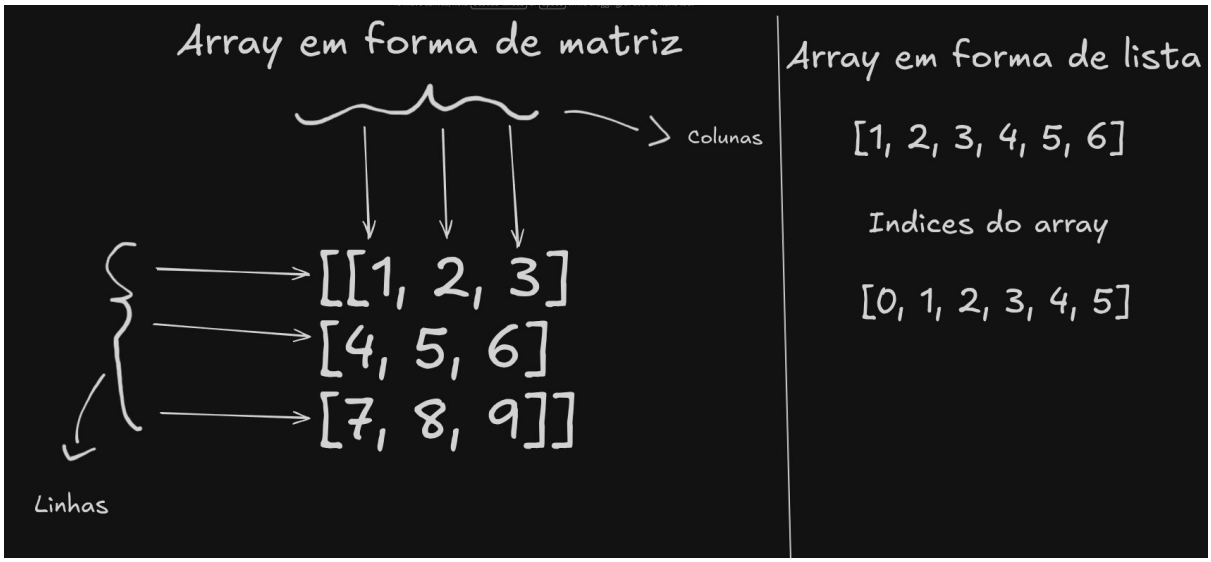


Numpy:

É a biblioteca de algebra linear do python, ela é muito rápida porque ela é escrita em C, com ela a gente mexe com arrays e ela tem funções pra manipular os dados desses arrays.

Um array vem de duas formas, em forma de vetor ou lista, e em matriz.

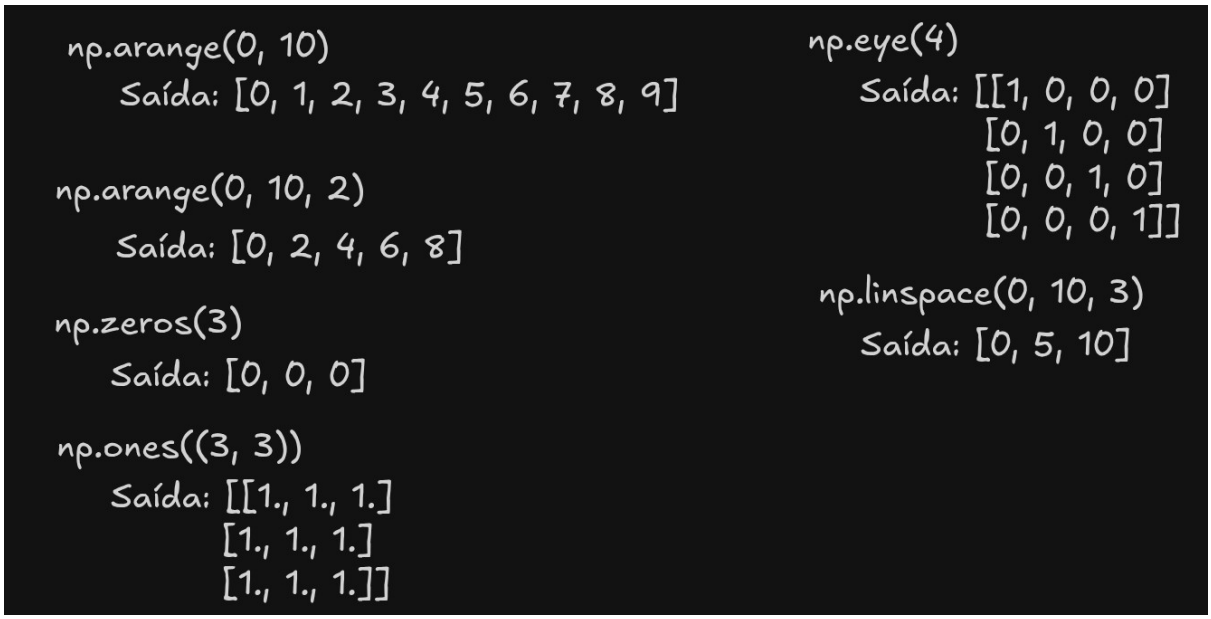
Um exemplo visual:



O array não tem limite de tamanho, ele pode ser até mesmo uma representação de um objeto 3D como um cubo. Pra criar um array usando o numpy é só importar o

numpy e dar um apelido geralmente é “np” e existem várias funções e formas de criar vou passar por algumas.

Se você cria uma lista tem como usar o np.array(e passar sua lista aqui como parametro e ela vira um array numpy), isso funciona se você tiver mais de uma lista ai ela vira uma matriz usando o mesmo modo.



Esses são alguns dos principais, o arange cria um array de 10 indices indo de 0 a 9, o terceiro parametro que é colocado no arange é o espaçamento se você quiser, no exemplo ele vai somando 2 em 2, o np.zeros() cria um array apenas com zeros do tamanho que quiser, mas se você coloca 4 paranteses e dentro coloca 2 valores ele joga pra uma matriz no exemplo eu usei a função ones pra criar uma matriz 3x3 só de uns, a função eye cria a matriz identidade no exemplo eu fiz uma 4x4 por útlimo o linspace faz um array de 0 a 10 e pega 3 números igualmente espaçados.

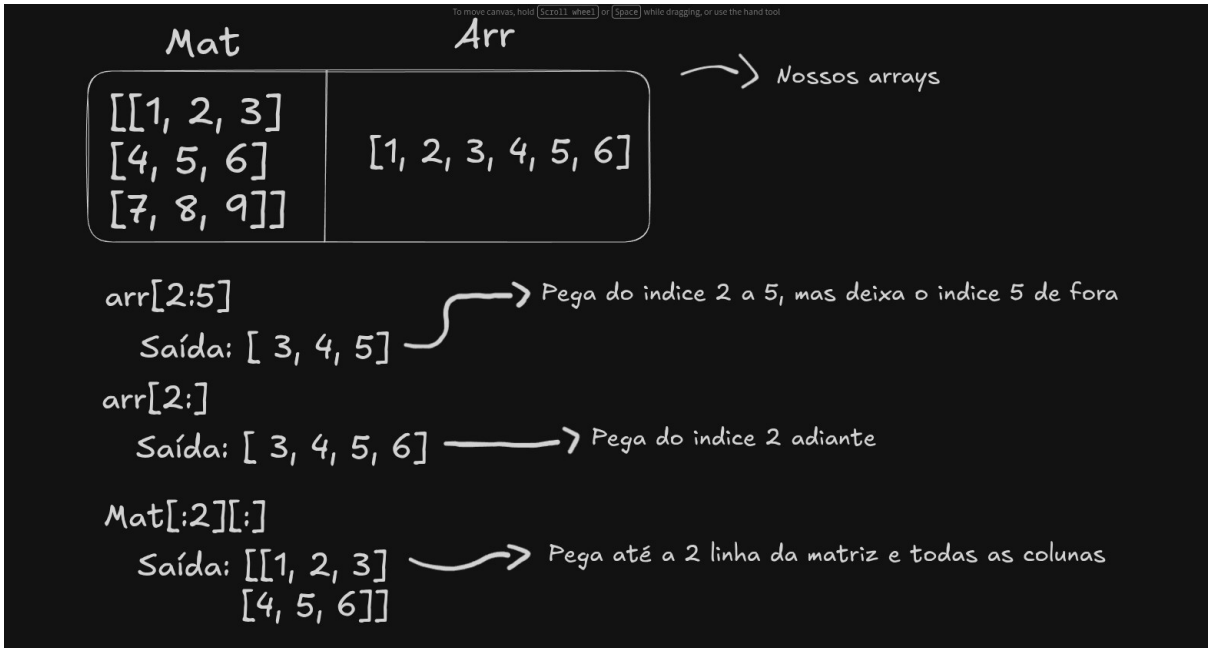
Existem as funções random que podem ser usado pra criar arrays, existe a random.rand(x) que pega a quantidade x de números aleatorios entre 0 e 1, a random.randn(x) faz a mesma coisa que a rand só que pega qualquer número aleatoriamente e não só de 0 a 1 e a random.randint(onde começa, onde para, quantidade de número que ira gerar) ex: random.randint(10, 60, 2) pega 2 numeros entre 10 e 60.

E para manipular os arrays existem:

O reshape que faz um array em forma de linha virar uma matriz e somente shape pega o formato da matriz exemplo: (5,4)

Tem funções que retornam os maiores e menores valores e também funções que dão o número do indice de onde esses valores estão. Existem as operações básicas como soma, subtração, divisão e multiplicação, as que pegam a raiz quadrada de cada valor, os valores exponenciais de cada valor, calcula a media do array todo, o desvio padrão e o seno de cada valor.

Também tem como fatiar os arrays da seguinte forma:

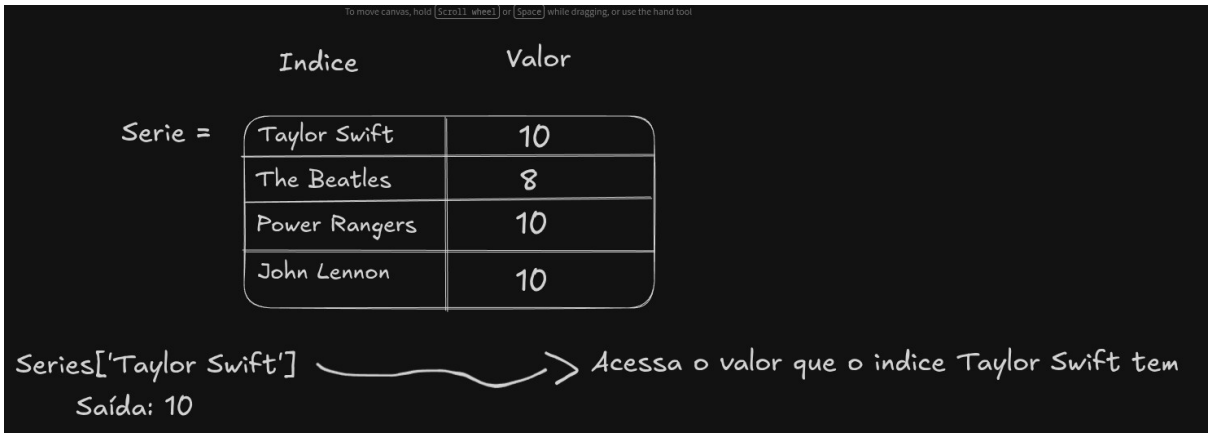


Pandas:

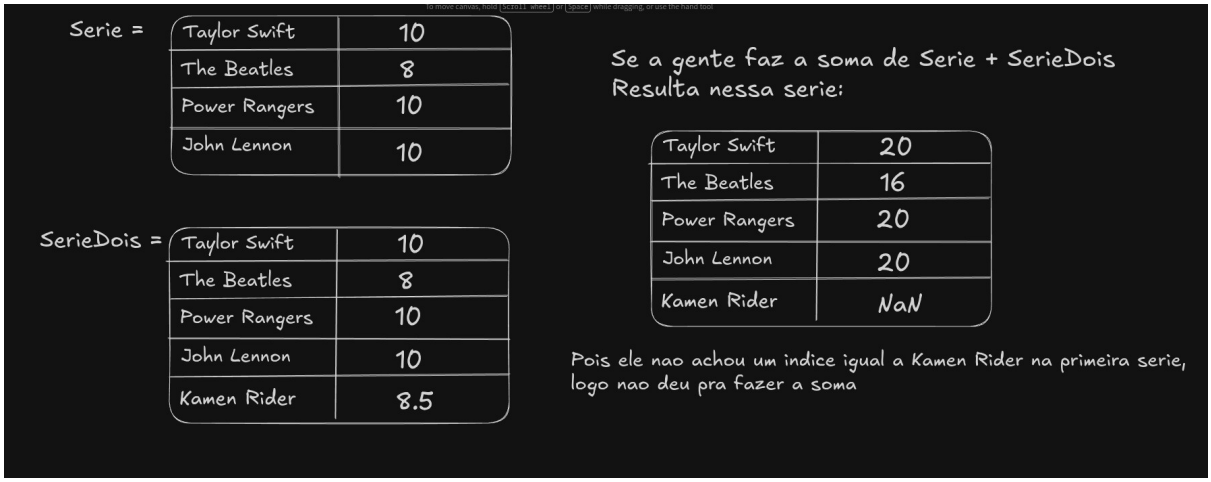
O pandas é outra biblioteca do python mais voltada pra analise de dados, ela tem estruturas como Series, Dataframes e outras funcionalidades que irei falar sobre.

Uma serie é uma estrutura muito parecida com um dicionário em python, ela tem chave – valor, onde a chave seria o index dessa estrutura pra você poder acessar os valores.

Exemplo visual:



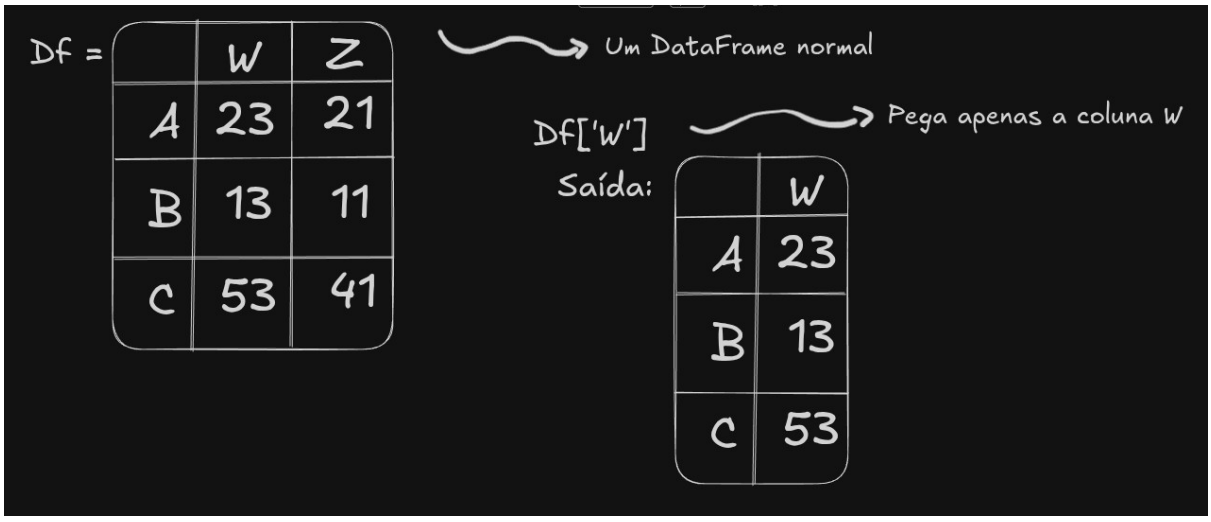
Existe as operações normais com as series, um outro exemplo:



DataFrames:

Um dataframe é uma estrutura que é formada por várias séries, ela já é uma tabela muito mais completa, podendo ter colunas onde cada coluna é uma series.

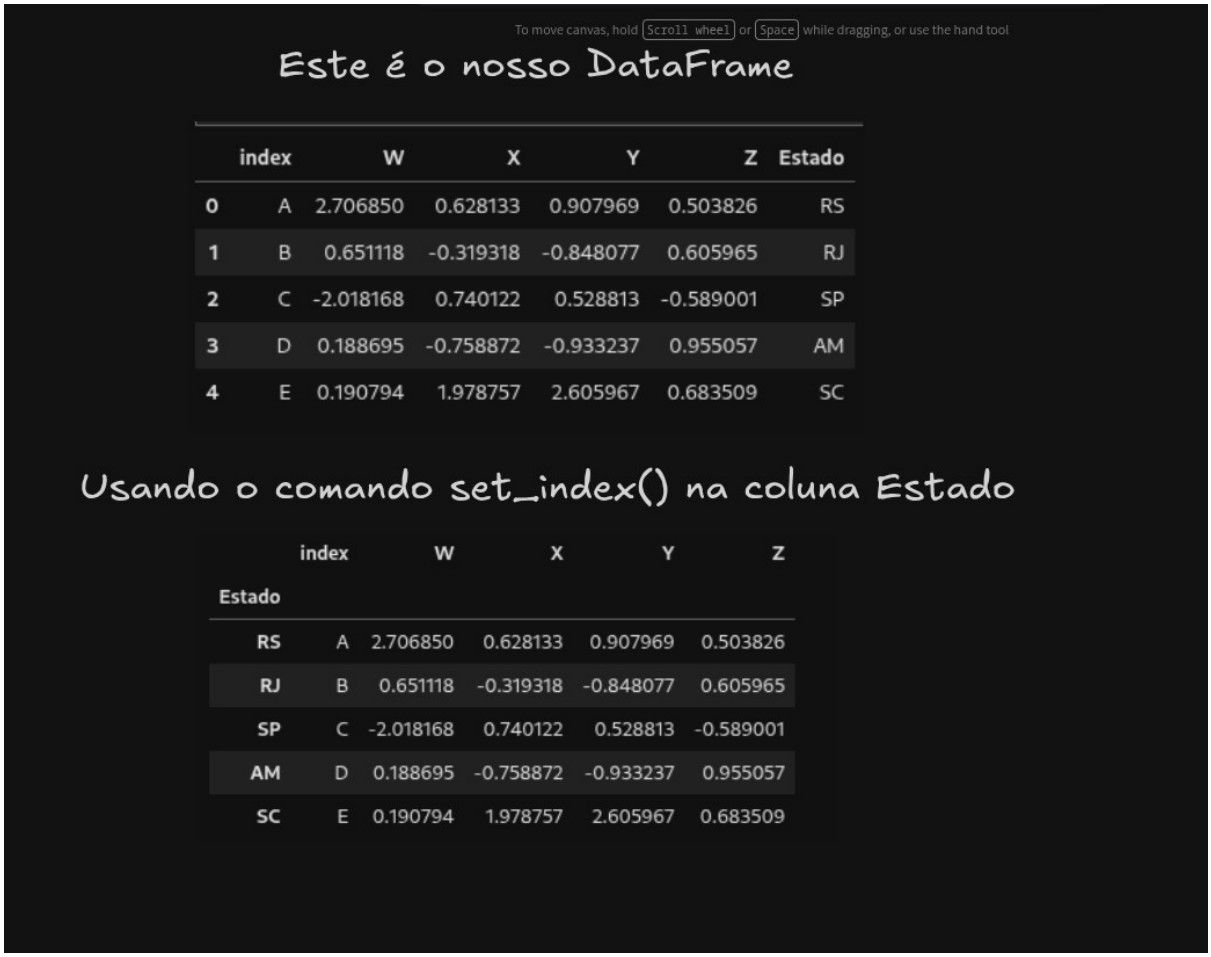
Exemplo visual:



E tem como criar colunas novas, apenas fingindo que elas já existem fazendo assim definindo o nome da coluna e os dados. E pra deletar ela só usar o comando drop e o nome da coluna.

Pra fazer buscas por algum valor exato se usa o comando loc e passa a linha e a coluna da onde o valor está e usando o iloc você passa o número da linha e coluna.

Existe também uma forma de setar um index novo com alguma das colunas, exemplo:



	index	W	X	Y	Z	Estado
0	A	2.706850	0.628133	0.907969	0.503826	RS
1	B	0.651118	-0.319318	-0.848077	0.605965	RJ
2	C	-2.018168	0.740122	0.528813	-0.589001	SP
3	D	0.188695	-0.758872	-0.933237	0.955057	AM
4	E	0.190794	1.978757	2.605967	0.683509	SC

Estado	index	W	X	Y	Z
RS	A	2.706850	0.628133	0.907969	0.503826
RJ	B	0.651118	-0.319318	-0.848077	0.605965
SP	C	-2.018168	0.740122	0.528813	-0.589001
AM	D	0.188695	-0.758872	-0.933237	0.955057
SC	E	0.190794	1.978757	2.605967	0.683509

E ainda exite uma forma de colocar dois indexs em um dataframe só, podendo ter duas formas de acessar valores usando a função xs no df.

Uma forma de tratar dados que o curso mostrou é de preencher valores NaN, usando funções fillna, dropna que excluem ou aplicam um valor novo.

Também da pra juntar dataframes de várias formas, concatenando que junta eles mas sem somar os dados, a mescla que junta os dataframe por algum objeto que eles tem iguais como parametro, e os métodos são iguais ao do SQL e a junção usando o indice como a referencia pra mescla e usa o left como padrão, é tipo um left join no sql.

As operações com os dataframes são bem parecidas com as da series, mas normalmente você ecolhe se vao ser de coluna a coluna ou linha por linha.

O pandas tem a opção de ler e exportar arquivos de .csv ou arquivo excel pra dataframe e vice e versa, também é possível ler paginas html e jogar elas pra um dataframe.

3. Conclusão

Em resumo, fazendo o card 4 deu pra aprender como usar o jupyter notebook, o numpy e o pandas, que são ferramentes importantes pra analise de dados em python. O aprendizado foi rápido e eficaz e a utilização do jupyter notebook foi essencial pra organização do código,o numpy foi muito bom para entender como mexer com arrays e matrizes e o pandas ajudou na organização de tabelas e dados estruturados.