



Inteligência Artificial e Big Data

Aula 01

Prof. Me Daniel Vieira

SENAI

Agenda

- 1- Apresentação
- 2 - Projetos realizados
- 3 - Ementa do curso
- 4- Capacidades técnicas e Socioemocionais
- 5 - Calendário das aulas
- 6 - Critérios de avaliação
- 7 - Recursos necessários
- 8 - Tipos de dados e formatação de saída
- 9 - Bibliotecas utilizadas em Data Science
- 10 - Biblioteca Pandas

Agenda

11- Criando um DataFrame

12 - Acessando colunas do Dataframe

13 - Exercícios

Apresentação

Doutorando e Mestre em Engenharia Elétrica - FEEC / Unicamp

Engenheiro Eletricista - UniMetrocamp Wyden

Técnico de Mecatrônica - Senai Roberto Mange

Áreas de atuação :

Projetos eletrônicos

Hardware e firmware

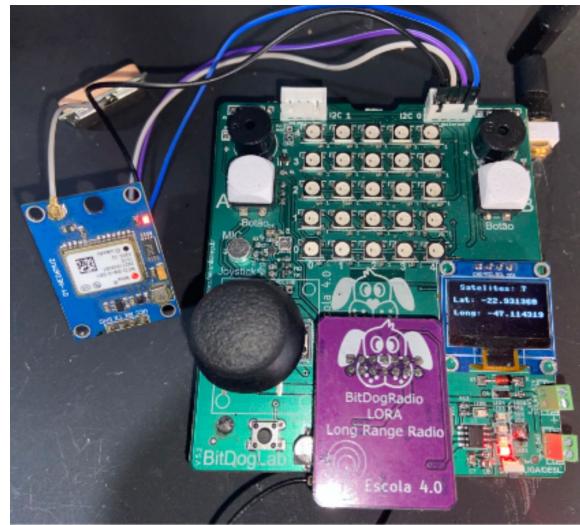
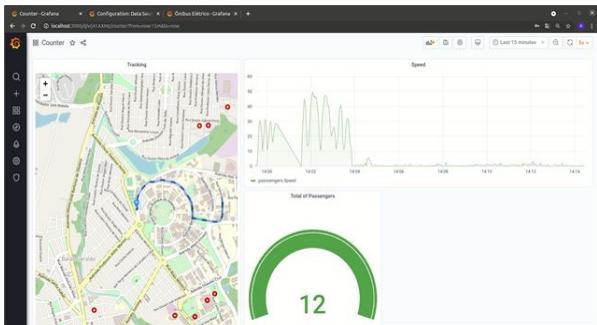
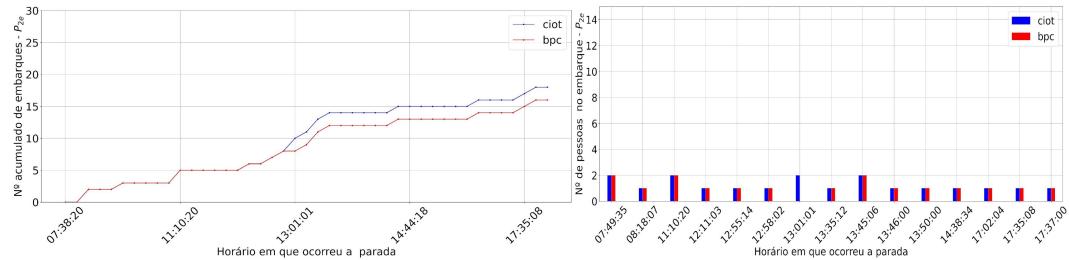
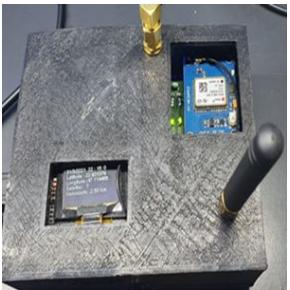
Desenvolvimento de aplicativos C++, C#, Python, Mobile

Análise de dados

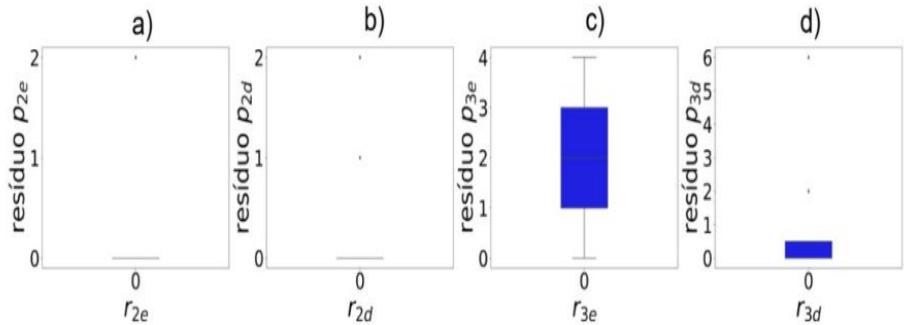
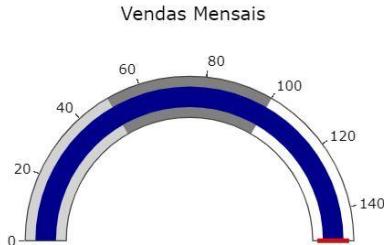
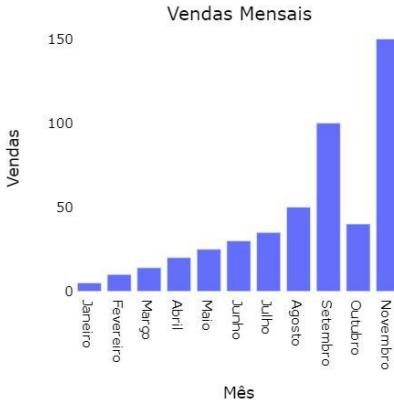
Pesquisa com IA, Machine Learning, Lógica Fuzzy

Projetos realizados

Monitoramento de pessoas e monitoramento por GPS



Projetos realizados



Mês	Vendas
1	5
2	10
3	14
4	20
5	25
6	30
7	35
8	50
9	100
10	40



Ementa do curso

Carga horária: 80 horas
presencial + 40 horas (EAD)

Inteligência Artificial e Big Data

Data Mining

Histórico
O que é Data Mining ?
Etapas Data Mining
Definições, aplicações
Fundamentos da aprendizagem
de máquina
KDD
Crisp DM
PCA

Inteligência Artificial

Histórico
Aplicações
Aprendizagem de máquinas
Modelos Supervisionados
Modelos não supervisionados
Modelos semi-supervisionados
Modelos preditivos

Big Data & Analytics

Fundamentos
Importância
5V's
Volume
Variedade
Velocidade
Veracidade
Valor

Ementa do curso

Carga horária: 80 horas
presencial + 40 horas (EAD)

Inteligência Artificial e Big Data

Data Mining

Algoritmos
Análise de Conglomerados -
Cluster Analysis
K-Means
KNN
H-Clust
Árvore de decisão
Random Forest
Support Vector Machine
Redes Neurais

Plataformas Big Data

Características
Distribuídos
Armazenamento
Processamento
Tempo de busca
Tempo de gravação
Cluster
Fundamentos
Arquitetura
Configuração
Ecossistema Apache,
Zookeeper, Oozie, Spark

Python

Algoritmos
Bibliotecas
Pandas
Numpy
Seaborn
Matplotlib
Plotly
Scikit-Learn

Capacidades técnicas/ Socioemocionais

- | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ol style="list-style-type: none">1. Aplicar algoritmos para mineração de dados2. Aplicar modelos de aprendizagem de máquinas no treinamento da IA3. Aplicar técnicas de Big Data para obtenção dos dados a serem tratados | <ol style="list-style-type: none">1. Demonstrar visão crítica2. Demonstrar organização3. Demonstrar atenção a detalhes4. Demonstrar capacidade de síntese5. Demonstrar capacidade de tomar decisão6. Demonstrar capacidade de solucionar problemas |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Calendário das aulas

Calendário das aulas

Mês	Setembro							Outubro										Novembro										
Dia	16 - 20				23 - 27			30 - 04				07-11			18 -21			25 - 28			01							
Nº aula	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56
CONTEÚDO FORMATIVO	Estudo de caso Clusterização de dados coletados de máquinas Temperatura, corrente elétrica, vibração Biblioteca Plotly Exercícios							Séries temporais Estudo de caso predição do preço de combustíveis							Big Data e Analytics Fundamentos, importância EVs Exercícios							Atividade Formativa						
Tempo real																												
Obs.																												

Calendário das aulas

Mês	Novembro												Dezembro											
Dia	04 - 08				11 - 22				18 - 29				25				2 - 6				9 - 13			
Nº aula	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
CONTEÚDO FORMATIVO																								
Tempo real	Redes Neurais Conceitos Exercícios								Estudo de caso Predição de nível de tanque Aplicação do algoritmo KNN Aplicação do algoritmo da árvore de decisão Estudo de caso Regressão logística e linear															
Obs.									Atividade Somativa								Revisão Tira dúvidas							
																		Recuperação						
																			Correção da Recuperação Finalização do semestre					

Calendário das aulas

Datas importantes

Atividade formativa: 07/10/2024 Turma B

Atividade formativa: 11/10/2024 Turma A

Atividade somativa: 18/11/2024 - Turma B + Artigo do projeto de análise de dados

Atividade somativa: 29/11/2024 - Turma A + Artigo do projeto de análise de dados

Recuperação: 02/12/2024 - Turma B

Recuperação: 06/12/2024 - Turma A

MF = Nota Somativa + Nota EAD / 2

Metodologias das aulas

Aulas teóricas, expositivas, exercícios e projetos

Aulas de Segunda - Turma B e Quinta - Turma A

Horário : 18:45 - 22:25

Intervalo: 20:50 - 21:05

Critérios de avaliação

Natureza dos Critérios	Fundamentos Técnicos e Científicos ou Capacidades Técnicas	Critérios de avaliação	Alunos							
			Crítico	Desejável						
1. Aplicar algoritmos para mineração de dados		Realizou a análise do estudo de caso e selecionou o algoritmo de mineração de dados apropriado para solução do problema Classificação, associação ou agrupamento?	F							
		O aluno acertou 7 questões dos formulários sobre conceitos teóricos sobre mineração de dados ?	S							

Critérios de avaliação

Competências Técnicas	2. Aplicar modelos de aprendizagem de máquinas no treinamento da IA.	<p>Realizou a análise do estudo de caso e selecionou o algoritmo de aprendizagem de máquina apropriado para solução do problema (Regressão logística, KNN, redes neurais)?</p> <p>O aluno acertou 7 questões dos formulários sobre conceitos teóricos sobre aprendizagem de máquinas e tipos de algoritmos utilizados para aprendizagem de máquina ?</p>	F	S								
			F	S								
3. Aplicar técnicas de Big Data para obtenção dos dados a serem tratados		<p>Realizou a análise do estudo de caso e selecionou a plataforma Hadoop para lidar com dados volumosos?</p> <p>O ambiente foi configurado corretamente ?</p>	F	S								
		F	S									

Critérios de avaliação

Competências Socioemocionais	1.Demonstrar visão crítica	O aluno acertou 7 questões dos formulários sobre conceitos teóricos sobre Big Data, 5Vs ?	F								
		O aluno realizou uma análise exploratória e estatística dos dados ?	F								
		O aluno verificou se os dados estão enviesados e se refletem uma amostra de dados generalista ou se refere a um pequeno <u>grupo</u> ?	F								

Critérios de avaliação

Competências Socioemocionais	2. Demonstrar organização	O aluno organizou o algoritmo implementado com comentários e documentação sobre cada etapa do processo de mineração de dados?	F				
		O aluno verificou se os dados estão enviesados e se refletem uma amostra de dados generalista ou se refere a um pequeno grupo ?	S				
	3. Demonstrar atenção aos detalhes	O aluno analisou a qualidade dos dados antes de iniciar o processo de treinamento e avaliação do modelo preditivo ?	F				
		Após realizar a análise exploratória dos dados, o aluno avaliou os resultados obtidos através de métricas estatísticas, Fscore?	S				
	4. Demonstrar capacidade de síntese	O aluno utilizou uma linguagem clara e objetiva para relatar suas observações sobre a análise de dados ?	F				
		De acordo com o estudo de caso o aluno foi capaz de selecionar as informações mais relevantes para de fazer inferências?	S.				

Critérios de avaliação

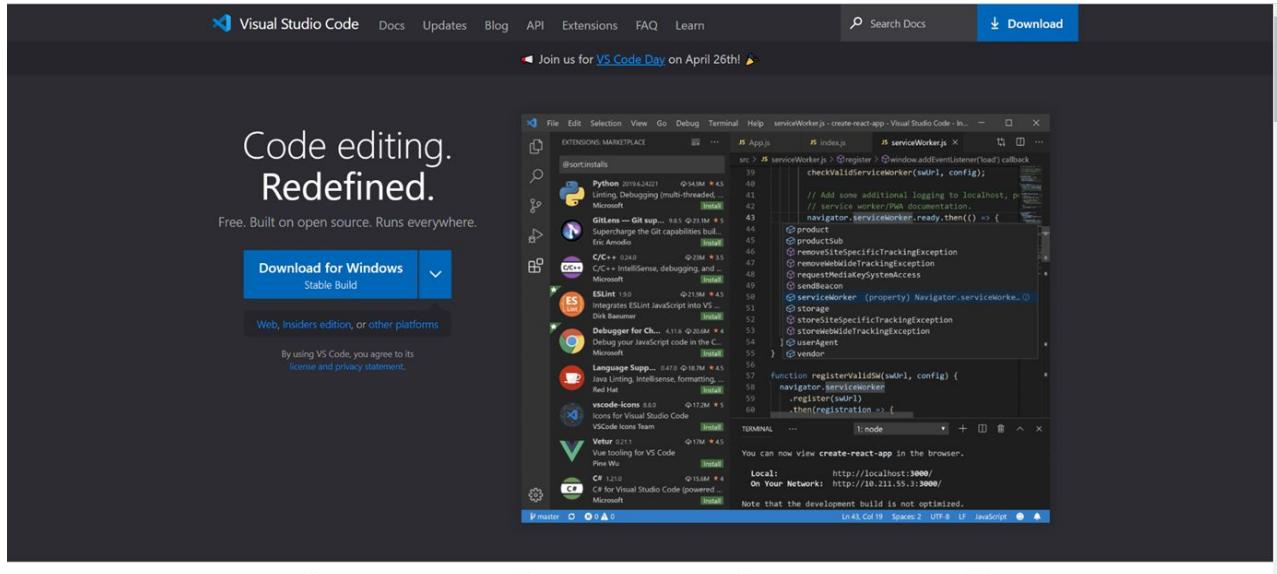
<p>5. Demonstrar capacidade de tomar decisão</p>	<p>O aluno é capaz de compreender o contexto daqueles dados e elaborar uma proposta de intervenção para melhoria dos resultados obtidos? Exemplo treinamento da rede neural com outro conjunto de dados, ajustes dos pesos da rede neural ?</p>	<p>F .</p>					
	<p>O aluno é capaz de avaliar incertezas na análise de dados obtida?</p>	<p>S .</p>					

Critérios de avaliação

Página 14 de 14

NÍVEIS DE DESEMPENHO	NÍVEIS	NOTA
Atingiu todos os critérios críticos e todos desejáveis	17	100
Atingiu todos os critérios críticos e 7 dos desejáveis	16	95
Atingiu todos os critérios críticos e 6 dos desejáveis	15	90
Atingiu todos os critérios críticos e 5 dos desejáveis	14	85
Atingiu todos os critérios críticos e 4 dos desejáveis	13	80
Atingiu todos os critérios críticos e 3 dos desejáveis	12	75
Atingiu todos os critérios críticos e 2 dos desejáveis	11	70
Atingiu todos os critérios críticos e 1 dos desejáveis	10	60
Atingiu todos os critérios críticos e nenhum desejável	9	50
Atingiu 4 critérios críticos e quaisquer desejáveis	5	40
Atingiu 3 critérios críticos e quaisquer desejáveis	4	30
Atingiu entre 1 e 2 critérios críticos e quaisquer desejáveis	3	20
Não atingiu nenhum critério crítico e quaisquer desejáveis	1	10
Não atingiu nenhum critério	0	0

Recursos necessários



Tipos de dados e formatação de saída

Tipos de variáveis

int - Armazena numeros inteiros, 3, 100, -5

float numeros com ponto flutuante 3.14,

100, 0.5

str - Armazena texto "Alura", "Daniel"

boolean True False

type () - função que retorna o tipo da variavel utilizada

Funções para conversão de dados

int(int(dado para conversão))

float(float(dado para conversão))

str(str(dado para conversão))

bool(bool(dado para conversão))

Tipos de dados e formatação de saída

f-string - Formatação de string para escrever valores de variaveis operadores de formatação

%s - string

%d - inteiro

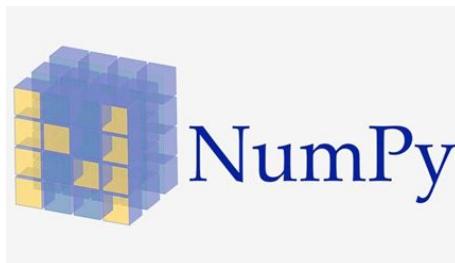
%f - float

%c - caractere

Tipos de dados e formatação de saída

```
nome_aluno = ' Daniel'  
idade_aluno = 28  
media_aluno = 8.45  
print('Nome do aluno é %s, ele tem %d  
anos e sua média é %f.' %(nome_aluno,  
idade_aluno, media_aluno))
```

Bibliotecas utilizadas para Data Science



Biblioteca Pandas

A biblioteca Pandas é uma biblioteca responsável por realizar a leitura de arquivos externos no formato csv, xlsx, parquet.



Biblioteca Pandas

Limpeza, tratamento de dados, análise exploratória dos dados, suporte em atividades de Machine Learning, consultas, queries em bancos de dados, visualização de dados, web scraping e etc

Biblioteca Pandas



Séries e Dataframes

As séries são arrays unidimensionais e com um eixo de rótulos chamado index

0	1.4
1	1.4
2	1.3
3	1.5
4	1.4
...	
145	5.2
146	5.0
147	5.2
148	5.4
149	5.1

Biblioteca Pandas



Séries e Dataframes

As séries são arrays unidimensionais e com um eixo de rótulos chamado index

0	1.4
1	1.4
2	1.3
3	1.5
4	1.4
...	
145	5.2
146	5.0
147	5.2
148	5.4
149	5.1

Dataframes - são objetos bidimensionais de tamanho variável. Formato de tabela

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
...
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

Biblioteca Pandas - Instalação

`pip install pandas`

Caso não tenha o pacote pip instalado na máquina
instalar o pacote pip

`python -m ensurepip --upgrade`

Biblioteca Pandas - Instalação e utilização

pip install pandas

Caso não tenha o pacote pip instalado na máquina

instalar o pacote pip

python -m ensurepip –upgrade

```
import pandas as pd # importação da biblioteca
```

Biblioteca Pandas - Criando um data frame

```
▶ import pandas as pd
Dicionario = {
    'A':[1,2,3],
    'B':[-3,-2,-1],
    'C':[0,10,20],
}
Label = ['1º Linha', '2º Linha', '3º Linha']
DataFrame_01 = pd.DataFrame(Dicionario, index= Label)
print(DataFrame_01)
[2] ✓ 0.2s
          A   B   C
1º Linha  1  -3   0
2º Linha  2  -2  10
3º Linha  3  -1  20
```

Python

[+ Code](#) [+ Markdown](#)

Biblioteca Pandas - Criando um data frame

```
DataFrame_01[['A', 'B']]
```

✓ 0.4s

Python

1º Linha

A

B

1

-3

2º Linha

2

-2

3º Linha

3

-1

Biblioteca Pandas - Criando uma nova coluna

```
DataFrame_01['Nova Coluna'] = DataFrame_01['A']*DataFrame_01['B']
```

✓ 0.1s

Python

```
DataFrame_01
```

✓ 0.3s

Python

	A	B	C	Nova Coluna
1º Linha	1	-3	0	-3
2º Linha	2	-2	10	-4
3º Linha	3	-1	20	-3

+ Code

+ Markdown

Biblioteca Pandas - Excluindo uma coluna

Excluir uma coluna

```
[73] DataFrame_01.drop('Nova_Coluna', axis=1, inplace=True)  
DataFrame_01
```

	A	B	C
1º Linha	1	-3	0
2º Linha	2	-2	10
3º Linha	3	-1	20

Biblioteca Pandas - Excluindo uma coluna

O parâmetro inplace = True quer dizer que vamos excluir da base original
inplace = False iria apenas excluir na situação atual

O parâmetro axis = 1 quer dizer que iremos excluir a coluna

Quando quisermos excluir uma linha axis = 0

Biblioteca Pandas - Localizando uma linha inteira

Localizar uma linha inteira

```
[80] DataFrame_01.loc['1º Linha']
```

A 1

B -3

C 0

Name: 1º Linha, dtype: int64

Biblioteca Pandas - Localizando diversas linhas e colunas

Localizar diversas linhas e colunas

```
[81] DataFrame_01.loc[['1º Linha', '3º Linha'], ['A', 'C']]
```

	A	C
1º Linha	1	0
3º Linha	3	20

Biblioteca Pandas - Localizando diversas linhas e colunas com parâmetros numéricos

Localizar diversas linhas e colunas usando parâmetros numéricos através da posição da base de dados

```
[90] DataFrame_01.iloc[2:3, 1:]
```

	B	C
3º Linha	-1	20

```
DataFrame_01.iloc[2:3, 1:]
```



Biblioteca Pandas - Comandos sob a base de dados

Verificando se há valores menores de 0 na base inteira

```
[93] DataFrame_01 > 0
```

	A	B	C
1º Linha	True	False	False
2º Linha	True	False	True
3º Linha	True	False	True

Biblioteca Pandas - Comandos sob a base de dados

Filtrando dados na base de dados com parâmetros em variáveis

```
[96] Filtro = DataFrame_01['C'] > 0
    DataFrame_02 = DataFrame_01[Filtro]
    DataFrame_02['A']
```

```
2º Linha    2
3º Linha    3
Name: A, dtype: int64
```

Nesse caso estamos :

1º Criando uma variável na qual estamos passando como parâmetro todos os casos da coluna C maior que 0

* Lembrando que nesse caso o pandas irá retornar verdadeiro ou falso

2º Criamos uma nova base de dados e passamos como filtro a variável definida na etapa 1

3º Retornando apenas os valores da coluna A

Biblioteca Pandas - Comandos sob a base de dados

Filtrando os dados com diversos parâmetros e condições

```
[97] DataFrame_01[  
      ( DataFrame_01['A'] > 1 ) & ( DataFrame_01['C'] > 0 )  
]
```

	A	B	C
2º Linha	2	-2	10
3º Linha	3	-1	20

Nesse caso estamos filtrando todos os casos da coluna A maior que 1 e passando outro parâmetro para filtrar todos os casos da coluna C maior 0.

Biblioteca Pandas - Comandos sob a base de dados

Transformando o index em uma coluna

```
[102] DataFrame_01.reset_index()
```

	index	A	B	C
0	1º Linha	1	-3	0
1	2º Linha	2	-2	10
2	3º Linha	3	-1	20

Nesse caso transportamos o index das linhas para virar uma coluna na base de dados. Assim o index foi resetado e ficou como numérico a partir de agora. Se colocar o comando '**inplace=True**' entre os parênteses, esse comando será aplicado para a base de origem.

Biblioteca Pandas - Comandos sob a base de dados

```
[109] # Criando varios dicionarios
Dicionario_01 = {'A' : [1, 2, 3],
                 'B' : [-32, -21, -15],
                 'C' : [60, 10, 20],
                 'Chave' : ['AA', 'BB', 'CC'] }

Dicionario_02 = {'A' : [6, 7, 8],
                 'B' : [-39, -28, -17],
                 'C' : [1000, 10, 60],
                 'Chave' : ['AA', 'BB', 'CC'] }

Dicionario_03 = {'A' : [11, 12, 13],
                 'B' : [-39, -22, -11],
                 'C' : [30, 10, 20],
                 'Chave' : ['AA', 'BB', 'CC'] }

# Criando varias listas para serem os labels
Label_01 = ['1º Linha', '2º Linha', '3º Linha']
Label_02 = ['4º Linha', '5º Linha', '6º Linha']
Label_03 = ['7º Linha', '8º Linha', '9º Linha']

# Estruturando as bases de dados
DataFrame_01 = pd.DataFrame( Dicionario_01, index=Label_01 )
DataFrame_02 = pd.DataFrame( Dicionario_02, index=Label_02 )
DataFrame_03 = pd.DataFrame( Dicionario_03, index=Label_03 )
```

Biblioteca Pandas - Comandos sob a base de dados

Empilhando os dados

```
[110] pd.concat(  
    [ DataFrame_01, DataFrame_02, DataFrame_03 ]  
)
```

	A	B	C	Chave
1º Linha	1	-32	60	AA
2º Linha	2	-21	10	BB
3º Linha	3	-15	20	CC
4º Linha	6	-39	1000	AA
5º Linha	7	-28	10	BB
6º Linha	8	-17	60	CC
7º Linha	11	-39	30	AA
8º Linha	12	-22	10	BB
9º Linha	13	-11	20	CC

O comando **concat** irá empilhar todas as bases de dados, desde que todas tenham a mesma estrutura (colunas).

Biblioteca Pandas - Comandos sob a base de dados

Função Mesclar permite que mescle os dados de diferentes base de dados. Essa função é semelhante a mesclagem de tabelas do SQL

```
[112] pd.merge(  
    DataFrame_01, DataFrame_02, how='inner', on='Chave'  
)
```

	A_x	B_x	C_x	Chave	A_y	B_y	C_y
0	1	-32	60	AA	6	-39	1000
1	2	-21	10	BB	7	-28	10
2	3	-15	20	CC	8	-17	60

O comando **merge** uni as colunas baseando em uma chave, muito similar ao SQL.

O parâmetro 'how' há diversas forma:

- **inner** = apenas os casos que localizou em ambos os lados
- **left** = apenas os dados da base de dados do lado esquerdo
- **right** = apenas os dados da base de dados do lado direito
- **outer** = união das chaves em ambos os lados
- **cross** = cria o produto cartesiano de ambos os quadros

Biblioteca Pandas - Comandos sob a base de dados

```
# Criando varios dicionarios
Dicionario_01 = {'A' : [1, 2, 3],
                 'B' : [-32, -21, -15],
                 'C' : [60, 10, 20] }

Dicionario_02 = {'D' : [6, 7, 8],
                 'E' : [-39, -28, -17],
                 'F' : [1000, 10, 60] }

# Criando varias listas para serem os labels
Label_01 = ['1º Linha', '2º Linha', '3º Linha']

# Estruturando as bases de dados
DataFrame_01 = pd.DataFrame( Dicionario_01, index=Label_01 )
DataFrame_02 = pd.DataFrame( Dicionario_02, index=Label_01 )

# Aplicando a função join
DataFrame_01.join(DataFrame_02)
```

	A	B	C	D	E	F
1º Linha	1	-32	60	6	-39	1000
2º Linha	2	-21	10	7	-28	10
3º Linha	3	-15	20	8	-17	60

Colunas A,BC eram da base de dados 1 e as colunas D,E,F eram da base de dados 2. Assim o join uniu todas elas

Biblioteca Pandas - Como realizar a leitura de um arquivo ?

A biblioteca pandas permite a leitura de diversos tipos de arquivo: csv, xls, url e etc

Arquivo excel

```
df = pd.read_excel('/content/contador df.xlsx')
```

Arquivo csv

```
df_evento = pd.read_csv('DF06.10.csv', sep=';')
```

Biblioteca Pandas -

Como realizar a leitura de um arquivo ?

Url

```
url = 'https://raw.githubusercontent.com/pdpcosta/lab2learn/master/DATA/ebmocsna.csv'  
#endereço de onde estão os dados que serão utilizados  
df = pd.read_csv(url)
```

Exercícios

- 1) Criar um dataframe utilizando a biblioteca pandas para os seguintes dados:

Nomes: Alice, Julia, Bruna, Daniel, Filipe, Silvio.

Idade: 19, 22, 28, 34, 28, 48

Profissões: Direito, Engenharia, Enfermagem, Engenheiro Civil, Cientista da Computação, Engenheiro Eletrônico

ID: 1, 2, 3, 4, 5, 6

- 2) Criar um dataframe utilizando a biblioteca pandas para os seguintes dados:

Número de produtos fabricados: 50, 30, 40, 100

Setores: 1, 2, 3, 4

- 3) Você tem disponível dois conjuntos de dados, um com o ID, nome dos funcionários e setores, o outro com id e o salário dos funcionários. Juntar os dois dataframes utilizando a função merge para transformar em um único dataframe utilizando a função merge

ID: 1, 2, 3, 4,

Nome: Ana, Pedro, Maria, João

Setores: Rh, Vendas, Administrativo, Produção

Dataframe 2

ID: 1, 2, 3, 4

Salários: 3000, 2800, 3200, 2500

Exercícios

4) Criar dois dataframes com dados de vendas de uma empresa

Data	Produto	Quantidade	Valor
2023-01-01	Item A	10	100.00
2023-01-02	Item B	5	50.00
2023-01-03	Item A	8	80.00

Data	Produto	Quantidade	Valor
2023-02-01	Item C	12	120.00
2023-02-02	Item B	6	60.00
2023-02-03	Item D	3	30.00

Obrigado!

Prof. Me Daniel Vieira

Email: danielvieira2006@gmail.com

Linkedin: Daniel Vieira

Instagram: Prof daniel.vieira95

