

A solid orange vertical bar is positioned on the far left side of the image.

harve
escola de inovação

A man with glasses and a white shirt is sitting at a desk, looking at a computer monitor. The monitor displays several data visualizations, including a line chart with multiple colored lines and a bar chart. The background is slightly blurred, showing a typical office environment.

Projeto Final

Data Science



Juliane Andressa Camatti

Turma T07

A atividade busca aprimorar os conhecimentos de Python oferecidos em aula, como uma das etapas importantes para atender a necessidade atual de indústrias em processar seus dados em tempo real em busca de mais assertividade nas tomadas de decisão.

Objetivo Geral:

Elaborar uma análise descritiva de dados oriundos de máquinas CNC portadoras de sistemas legados, com o intuito de conhecer o real desempenho de máquina.



Objetivo Específico do projeto:

Fazer a junção dos dados fornecidos pela área em uma mesma tabela para identificar:

- Qualidade dos dados;
- Oportunidades de análise dos mesmos;
- **Tempo total de funcionamento da máquina;





Execução do Projeto

Forneça detalhes das etapas e ferramentas para o projeto

- Quais bases de dados foram usadas?
- Quais dados elas continham?
- Qual foram os critérios de seleção do Modelo.



001.xlsx
002.xlsx
003.xlsx
004.xlsx
005.xlsx
006.xlsx
007.xlsx
008.xlsx
009.xlsx
010.xlsx
011.xlsx
012.xlsx
013.xlsx
014.xlsx



Forneça detalhes das etapas e ferramentas para o projeto

1º Passo: salvar os dados na pasta no drive

2º Passo: importar os dados



```
from google.colab import drive
drive.mount('/content/drive')
%cd drive/MyDrive/
%pwd
```



Forneça detalhes das etapas e ferramentas para o projeto

3º Passo: Importar todas as planilhas para análise, elegendo a coluna horário como fixa.

```
[3] import pandas as pd
df1 = pd.read_excel('001.xlsx', index_col = 0)
df2 = pd.read_excel('002.xlsx', index_col = 0)
df3 = pd.read_excel('003.xlsx', index_col = 0)
df4 = pd.read_excel('004.xlsx', index_col = 0)
df5 = pd.read_excel('005.xlsx', index_col = 0)
df6 = pd.read_excel('006.xlsx', index_col = 0)
df7 = pd.read_excel('007.xlsx', index_col = 0)
df8 = pd.read_excel('008.xlsx', index_col = 0)
df9 = pd.read_excel('009.xlsx', index_col = 0)
df10 = pd.read_excel('010.xlsx', index_col = 0)
df11 = pd.read_excel('011.xlsx', index_col = 0)
df12 = pd.read_excel('012.xlsx', index_col = 0)
df13 = pd.read_excel('013.xlsx', index_col = 0)
df14 = pd.read_excel('014.xlsx', index_col = 0)
```



Forneça detalhes das etapas e ferramentas para o projeto

4º Passo: Concat. Usar essa função para colocar todas as planilhas em uma mesma coluna de acordo com a coluna Index (representa hora e data)

```
[4] df= pd.concat([df1, df2, df3, df4, df5, df6, df7, df8, df9, df10, df11, df12, df13, df14],axis = 1)
```

```
[5] df1.shape #tamanho planilha 1
```

(106, 1)

```
[6] df.shape #tamanho planilha toda
```

(544, 14)

Divergências nas colunas de dados



Forneça detalhes das etapas e ferramentas para o projeto

```
dfttotal.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 544 entries, 0 to 543
```

```
Data columns (total 15 columns):
```

#	Column	Non-Null Count	Dtype
0	index	544 non-null	object
1	MODO AUTO	106 non-null	float64
2	MODO MDI	100 non-null	float64
3	MODO JOG	106 non-null	float64
4	FEED DISABLE	100 non-null	float64
5	FEEDRATE OVERRIDE	106 non-null	float64
6	FEED STOP GEOAXIS 1	100 non-null	float64
7	HORAS TOTAIS DE FUNCIONAMENTO	186 non-null	float64
8	MINUTOS TOTAIS DE FUNCIONAMENTO	198 non-null	float64
9	CONTADOR TROCA DE PALETES	167 non-null	float64
10	CONTADOR DE PEÇAS	100 non-null	float64
11	SPINDLE OVERRIDE	101 non-null	float64
12	MOTOR STATUS	107 non-null	float64
13	ALARME TEMPERATURA MOTOR	100 non-null	float64
14	pinça spindlle	199 non-null	float64

```
dtypes: float64(14), object(1)
```

```
memory usage: 63.9+ KB
```

```
dfttotal.isnull().sum() #dados faltantes
```

index	0
MODO AUTO	438
MODO MDI	444
MODO JOG	438
FEED DISABLE	444
FEEDRATE OVERRIDE	438
FEED STOP GEOAXIS 1	444
HORAS TOTAIS DE FUNCIONAMENTO	358
MINUTOS TOTAIS DE FUNCIONAMENTO	346
CONTADOR TROCA DE PALETES	377
CONTADOR DE PEÇAS	444
SPINDLE OVERRIDE	443
MOTOR STATUS	437
ALARME TEMPERATURA MOTOR	444
pinça spindlle	345
dtype: int64	



Forneça detalhes das etapas e ferramentas para o projeto

```
✓ [138] df_analise = pd.concat([df1, df3],axis = 1)
```

0s

```
✓ [139] df_novo = df_analise.reset_index(drop=False)  
df_novo.head(10)
```

0s

	hora	MODO	AUTO	MODO	JOG
0	09/03/2021 10:57:01		1		0
1	09/03/2021 16:59:30		1		0
2	09/03/2021 17:59:33		1		0
3	09/03/2021 18:59:37		1		0
4	09/03/2021 19:59:39		1		0
5	09/03/2021 20:59:41		1		0
6	09/03/2021 21:59:43		1		0
7	09/03/2021 22:59:46		1		0
8	09/03/2021 23:59:48		1		0
9	10/03/2021 00:59:50		1		0



Forneça detalhes das etapas e ferramentas para o projeto

Formatar hora

```
[17] df_novo['data'] = pd.to_datetime(df_novo['hora'], dayfirst= True)  
df_novo.head(10)
```

	hora	MODO AUTO	MODO JOG	data
0	09/03/2021 10:57:01	1	0	2021-03-09 10:57:01
1	09/03/2021 16:59:30	1	0	2021-03-09 16:59:30
2	09/03/2021 17:59:33	1	0	2021-03-09 17:59:33
3	09/03/2021 18:59:37	1	0	2021-03-09 18:59:37
4	09/03/2021 19:59:39	1	0	2021-03-09 19:59:39
5	09/03/2021 20:59:41	1	0	2021-03-09 20:59:41
6	09/03/2021 21:59:43	1	0	2021-03-09 21:59:43
7	09/03/2021 22:59:46	1	0	2021-03-09 22:59:46
8	09/03/2021 23:59:48	1	0	2021-03-09 23:59:48
9	10/03/2021 00:59:50	1	0	2021-03-10 00:59:50



Forneça detalhes das etapas e ferramentas para o projeto

Selecionar data

```
df_novo['hora_novo_agrupamento'] = df_novo['data'].dt.to_period('D')  
df_novo.head(10)
```

	hora	MODO	AUTO	MODO	JOG	data	hora_novo_agrupamento
0	09/03/2021 10:57:01		1		0	2021-03-09 10:57:01	2021-03-09
1	09/03/2021 16:59:30		1		0	2021-03-09 16:59:30	2021-03-09
2	09/03/2021 17:59:33		1		0	2021-03-09 17:59:33	2021-03-09
3	09/03/2021 18:59:37		1		0	2021-03-09 18:59:37	2021-03-09
4	09/03/2021 19:59:39		1		0	2021-03-09 19:59:39	2021-03-09
5	09/03/2021 20:59:41		1		0	2021-03-09 20:59:41	2021-03-09
6	09/03/2021 21:59:43		1		0	2021-03-09 21:59:43	2021-03-09
7	09/03/2021 22:59:46		1		0	2021-03-09 22:59:46	2021-03-09
8	09/03/2021 23:59:48		1		0	2021-03-09 23:59:48	2021-03-09
9	10/03/2021 00:59:50		1		0	2021-03-10 00:59:50	2021-03-10



Forneça detalhes das etapas e ferramentas para o projeto

Agrupamento baseado na hora e na soma do status ligado

```
[22] df_novo.groupby('hora_novo_agrupamento')['MODO AUTO'].sum()
```

```
hora_novo_agrupamento
2021-03-09      9
2021-03-10     24
2021-03-11     24
2021-03-12     24
2021-03-13     19
Freq: D, Name: MODO AUTO, dtype: int64
```



Forneça detalhes das etapas e ferramentas para o projeto

Criar uma coluna nova sobre o status da máquina

```
def faixa(x):  
    if x == 0:  
        return('desligada')  
    elif x == 1:  
        return('ligada')  
    else:  
        return('Valor não encontrado!')  
  
df_novo['status'] = df_novo['MODO AUTO'].apply(faixa)  
df_novo.head(20)
```

	hora	MODO AUTO	MODO JOG	data	hora_novo_agrupamento	status
0	09/03/2021 10:57:01	1	0	2021-03-09 10:57:01	2021-03-09	ligada
1	09/03/2021 16:59:30	1	0	2021-03-09 16:59:30	2021-03-09	ligada
2	09/03/2021 17:59:33	1	0	2021-03-09 17:59:33	2021-03-09	ligada
3	09/03/2021 18:59:37	1	0	2021-03-09 18:59:37	2021-03-09	ligada
4	09/03/2021 19:59:39	1	0	2021-03-09 19:59:39	2021-03-09	ligada
5	09/03/2021 20:59:41	1	0	2021-03-09 20:59:41	2021-03-09	ligada
6	09/03/2021 21:59:43	1	0	2021-03-09 21:59:43	2021-03-09	ligada



Forneça detalhes das etapas e ferramentas para o projeto

Desenvolver uma tabela dinâmica com o pivot_table

```
▶ table = pd.pivot_table(df_novo, index=['status','hora_novo_agrupamento'], values = ['MOD0 AUTO'])  
table.head(30)
```

MOD0 AUTO		
status	hora_novo_agrupamento	
desligada	2021-03-10	0
	2021-03-11	0
	2021-03-12	0
ligada	2021-03-09	1
	2021-03-10	1
	2021-03-11	1
	2021-03-12	1
	2021-03-13	1





Conclusões

Relembrando:

Fazer a junção dos dados fornecidos pela área em uma mesma planilha ✓
para identificar:

- Qualidade dos dados; ✓
- Oportunidades de análise dos mesmos; ✓
- Tempo total de funcionamento da máquina; ✓



Forneça detalhes das etapas e ferramentas para o projeto

```
df.head(10) # para observar as 10 primeiras linhas da planilha toda
```



09/03/2021 10:57:01	1.0	0.0	0.0	0.0	26.0	0.0	7290.0	437400.0	85200.0	0.0	14.0	1.0
09/03/2021 16:59:30	1.0	0.0	0.0	0.0	26.0	0.0	17642.0	1058559.0	3399.0	0.0	14.0	1.0
09/03/2021 17:59:33	1.0	0.0	0.0	0.0	26.0	0.0	17643.0	1058618.0	3401.0	0.0	14.0	1.0
09/03/2021 18:59:37	1.0	0.0	0.0	0.0	26.0	0.0	17644.0	1058678.0	3401.0	0.0	14.0	1.0
09/03/2021 19:59:39	1.0	0.0	0.0	0.0	26.0	0.0	17645.0	1058738.0	3403.0	0.0	14.0	1.0



Trabalhos futuros

Quais passos podem ser necessários para um futuro desenvolvimento do projeto.

- ✓ Fazer uma limpeza dos dados;
- ✓ Padronizar os dados em relação a coluna tempo;
- ✓ Fazer mais análises mais detalhadas como a correlação, média entre outros;
- ✓ Gerar gráficos e dashboard no powerbi (talvez);
- ✓ Processo seja automático toda vez que for liberado relatório novo no servidor da empresa;



Referências

Informe os links de notebooks, blogs e livros que foram utilizados:

- Análise descritiva de dados;
Python para ciência e dados;
- Orientação: Prof. Bruna;
- Help: Dani e do Carlos;
- Programe Python - **YouTube**





VOCÊ EM CONEXÃO
com o mercado digital