Dio Análise de Dados com Python e Pandas.

Repositório criado para análise do aprendizado do curso de Análise de Dados com Python e Pandas.

Informações do Registro de Ocupação Hospitalar COVID-19

<https://dados.gov.br/dataset/registro-de-ocupacao-hospitalar>

Afim de testar os conhecimentos adquiridos no curso de análise de dados com Python e Pandas.

Busquei fonte de dados na internet e a busca me levou até o site: www.dados.gov.br, que se trata do portal brasileiro de dados abertos.

Tendo muitas opções de DataSet disponíveis optei nas Informações do Registro de Ocupação Hospitalar COVID-19.

Através do endereço:https://dados.gov.br/dataset/registro-de-ocupacao-hospitalar

Mãos à massa com o DataSet em mão

Entrei no google colab:https://colab.research.google.com

E também abri no navegador os sites de documentação do Python, Pandas e Matplotlib:

Python:https://docs.python.org/3/

Pandas:https://pandas.pydata.org/docs/

Matplotlib:https://matplotlib.org/3.5.1/index.html

O dataset tem mais de 260mb de informações sobre a ocupação de leitos COVID19 pelo País, separei uma amostra com um pouco mais de 18mb para aplicar os conhecimentos adquiridos no curso. Que me deu uma boa base para aprender mais sobre Análise de dados.

Desde de a importação de dados até a geração e visualização dos mesmo em gráficos diversos usando Python, pandas e matplotlib.

Utilizando o google colab criaremos um notebook onde a utilizando os recursos acima mencionado iremos tratar, manipular, analisar e visualizar as informações do dataset.

Importando as bibliotecas necessárias:

#Importando as bibliotecas

import pandas as pd

import matplotlib.pyplot as plt

#Configurando o estilo a ser usado.

#plt.style.available

plt.style.use("seaborn-dark")

Para importar o dataset podemos usar:

Dessa forma usaremos o files do google para fazer o upload do nosso arquivo para gerar o nosso dataframe

#upload do arquivo de dados

from google.colab import files

arq = files.upload()

Lendo o arquivo após o upload:

#Criando nosso DataFrame

df = pd.read\_csv("LeitoOcupacao.csv")

#Visualizando as 5 primeiras linhas

df.head(2)

#Quantidade de linhas e colunas

df.shape

#Tipo de dados do dataframe

df.dtypes

#Formatando campos data de Objeto para datetime

df["dataNotificacao"] = pd.to\_datetime(df["dataNotificacao"])

df["\_created\_at"] = pd.to\_datetime(df["\_created\_at"])

df["\_updated\_at"] = pd.to\_datetime(df["\_updated\_at"])

#Tipo de dados do dataframe

df.dtypes

df.head(2)

df.describe()

#Filtrando os estados informados na notificações !

df["estado"].unique()

#Agrupando por estado e contando as ocorrências.

df.groupby(["estado"])["municipio"].count()

#Agrupando os municipios por estado e contando as ocorrências.

df.groupby(["estado","municipio"])["municipio"].count()

#Separando por Ano e Mês

df["ano"] = df["dataNotificacao"].dt.year

df["mes"] = df["dataNotificacao"].dt.month

#Agrupando ocorrencias por estado, ano

df.groupby(["ano","estado"])["municipio"].count()

#Agrupando os munícipios por estado e contando as ocorrências.

df.groupby(["estado"])["municipio"].count().sort\_values(*ascending*=True)

#Gráfico de barras notificações por estado

df.groupby(["estado"])["municipio"].count().sort\_values(*ascending*=False).plot.barh(*title*="Ocorrências" , *color*='limegreen')

plt.xlabel("Qtde ocorrências", *color*='blue')

plt.ylabel("Estado", *color*='blue')

*;*

#Gráfico de ocorrências por ano/mês

df.groupby(["ano", "mes"])["estado"].count().plot.barh(*title*="Ocorrências" , *color*='limegreen')

plt.xlabel("Estado", *color*='blue')

plt.ylabel("Ocorrências", *color*='blue')

*;*

#Criando Dataframe somente com o ultimo trimestre.

df202112 = df.loc[(df["ano"]>= 2021) & (df["mes"] >= 12 )]

df202201 = df.loc[(df["ano"]>= 2022) & (df["mes"] >= 1 )]

# Concatenando os arquivos criados em um único.

dfTrimestre = pd.concat([df202112, df202201])

#Verficando linha e colunas.

dfTrimestre.shape

#Gerando gráfico de barras do último trimestre

dfTrimestre.groupby(["ano", "mes"])["estado"].count().sort\_values(*ascending*=False).plot.barh(*title*="Ocorrências", *color*="limegreen")

plt.xlabel("Quantidade Ocorrências", *color*='blue')

plt.ylabel("Ano Mêss ", *color*='blue')