

```

import pandas as pd

# Caminho do arquivo Excel
file_path = '/content/drive/MyDrive/Python/Chuva-AlertaRJ_2014-2024.xlsx'

# Carregar as planilhas Atlas2013-2023 e Mensal_2014-2024
df_alertas = pd.read_excel(file_path, sheet_name='Atlas2013-2023')
df_mensal = pd.read_excel(file_path, sheet_name='Mensal_2014-2024')

# Ajustar o formato das colunas de datas em 'df_mensal'
# Para garantir que as colunas de datas sejam strings no formato 'YYYY-MM'
df_mensal.columns = df_mensal.columns.map(lambda x: x.strftime('%Y-%m') if isinstance(x, pd.Timestamp) else x)

# Selecionar a coluna 'Mês/Ano' da planilha de alertas
mes_ano_alertas = df_alertas['Mês/Ano'].dropna().unique()

# Lista para armazenar os dados dos 5 maiores valores de médias mensais por Mês/Ano
all_maiores_valores = []

# Iterar por cada Mês/Ano
for mes_ano in mes_ano_alertas:
    if mes_ano in df_mensal.columns:
        dados_mes_ano = df_mensal[['Estação', mes_ano]].copy()
        maiores_valores = dados_mes_ano[mes_ano].nlargest(5).values

        all_maiores_valores.append({
            'Mês/Ano': mes_ano,
            '1º Maior': maiores_valores[0] if len(maiores_valores) > 0 else None,
            '2º Maior': maiores_valores[1] if len(maiores_valores) > 1 else None,
            '3º Maior': maiores_valores[2] if len(maiores_valores) > 2 else None,
            '4º Maior': maiores_valores[3] if len(maiores_valores) > 3 else None,
            '5º Maior': maiores_valores[4] if len(maiores_valores) > 4 else None
        })

# Criar o DataFrame a partir da lista
tabela_maiores_valores = pd.DataFrame(all_maiores_valores)

# Exibir a tabela final
print(tabela_maiores_valores)

```

```
# Salvar o resultado final em formato Parquet
tabela_maiores_valores.to_parquet('/content/drive/MyDrive/Python/Ta
bela_Maiores_Valores.parquet', index=False)
```