Opções possíveis para o comando

df['Data'].dt.to\_period('M')

O método to\_period() do **pandas** converte uma coluna de datas (datetime) em um **período temporal** (objeto Period), e o argumento passado (no seu exemplo 'M') define a **frequência do período**.

**📌 Opções mais comuns para to\_period(freq)**

Aqui estão as mais usadas (há muitas variações):

* **Ano**
  + 'A' ou 'Y' → período anual (terminando em dezembro).
  + 'A-JAN', 'A-FEB', … 'A-DEC' → ano fiscal terminando em um mês específico.
* **Semestre e Trimestre**
  + '6M' ou '2Q' → semestres.
  + 'Q' → trimestre (terminando em dezembro).
  + 'Q-JAN', 'Q-FEB', … → trimestre terminando em um mês específico.
* **Mensal**
  + 'M' → período mensal.
* **Semanal**
  + 'W' → período semanal (terminando no domingo).
  + 'W-MON', 'W-TUE', … 'W-SAT' → semana terminando em um dia específico.
* **Diário**
  + 'D' → período diário.
* **Hora, Minuto, Segundo**
  + 'H' → período de hora.
  + 'T' ou 'min' → período de minuto.
  + 'S' → período de segundo.
* **Sub-segundos (se precisar de granularidade extrema)**
  + 'L' → milissegundo.
  + 'U' → microssegundo.
  + 'N' → nanossegundo.

**📌 Exemplo prático**

import pandas as pd

# Criar exemplo

df = pd.DataFrame({'Data': pd.date\_range('2023-01-01', periods=3, freq='15D')})

df['Ano'] = df['Data'].dt.to\_period('Y') # Ano

df['Mes'] = df['Data'].dt.to\_period('M') # Mês

df['Trime'] = df['Data'].dt.to\_period('Q') # Trimestre

df['Semana']= df['Data'].dt.to\_period('W') # Semana

df['Dia'] = df['Data'].dt.to\_period('D') # Dia

print(df)

Saída (exemplo):

**Data Ano Mes Trime Semana Dia**

**0 2023-01-01 2023 2023-01 2023Q1 2022-12-26/2023-01-01 2023-01-01**

**1 2023-01-16 2023 2023-01 2023Q1 2023-01-16/2023-01-22 2023-01-16**

**2 2023-01-31 2023 2023-01 2023Q1 2023-01-30/2023-02-05 2023-01-31**

👉 Resumindo: o argumento do to\_period() é a **freq string**, que segue o mesmo padrão de frequência do pandas (DateOffset aliases).