

Tratamentos Realizados no DataFrame

1. Conversão para Tipo Numérico:

A coluna `idade` foi convertida para valores numéricos, com a opção `errors="coerce"`, que converte valores não numéricos em NaN:

```
df['idade'] = pd.to_numeric(df["idade"], errors="coerce")
```

O mesmo foi feito para a coluna `nota`:

```
df['nota'] = pd.to_numeric(df["nota"], errors="coerce")
```

2. Preenchimento de Valores Faltantes:

Todos os valores NaN (Not a Number) nas colunas foram substituídos por 0:

```
df.fillna(0, inplace=True)
```

3. Tratamento de Valores de "Ativo":

A coluna `ativo` foi normalizada para valores booleanos (True ou False), com a substituição de várias formas de escrita (como 'sim', 'não', 'TRUE', 'Yes', etc.) para os valores corretos:

```
df['ativo'] = df['ativo'].replace({
    'sim': True,
    'Sim': True,
    'não': False,
    'Nao': False,
    'nao': False,
    'Não': False,
    'TRUE': True,
    'FALSE': False,
    'e': False,
    'Yes': True
})
```

4. Tratamento de Data de Inscrição:

A coluna `data_inscricao` foi corrigida para tratar formatos errados de data. Alguns valores foram substituídos por datas válidas:

```
df['data_inscricao'] = df['data_inscricao'].replace({
    0: '2023-09-11',
    '2022-08-XX': '2022-08-01',
    '2023/13/01': '2023-12-01',
    '15/09/2022': '2022-09-15',
    '2023-02-30': '2023-02-28'
})
```

5. Substituição de Idade e Nota:

A coluna `idade` teve valores 0.0 substituídos por 20.0:

```
df['idade'] = df['idade'].replace({
    0.0: 20.0
})
```

Da mesma forma, a coluna `nota` teve valores 0.0 substituídos por 10.0:

```
df['nota'] = df['nota'].replace({
    0.0: 10.0
})
```

6. Conversão para Data:

A coluna `data_inscricao` foi convertida para o tipo de dados `datetime`, no formato `'%Y-%m-%d'`. Qualquer valor que não pôde ser convertido foi tratado com a opção `errors='coerce'`, resultando em `NaT` (Not a Time) para valores inválidos:

```
df['data_inscricao'] = pd.to_datetime(df['data_inscricao'],
format='%Y-%m-%d', errors='coerce')
```