



**UNIVERSIDADE ESTÁCIO DE SÁ**  
**Polo Jereissati 1 - Maracanaú/CE**  
**Tecnologia em Desenvolvimento Full Stack**  
**RPG0033 - Tratando a Imensidão dos Dados**

**Aluno:** Emanuel Roseira Guedes

**Matrícula:** 202212181407

**GitHub:** <https://github.com/guedesert/tratando-a-imensidao-dos-dados>

## **RELATÓRIO DA MISSÃO PRÁTICA**

### **1 INTRODUÇÃO**

O tratamento adequado de dados é essencial para garantir análises precisas e confiáveis, permitindo a extração de informações relevantes a partir de grandes volumes de dados brutos. A limpeza e organização dos dados são etapas fundamentais para evitar inconsistências e melhorar a qualidade dos insights gerados.

Este relatório documenta a execução da missão prática Tratando a Imensidão dos Dados, cujo objetivo é aplicar a biblioteca Pandas em Python para manipulação e tratamento de dados em arquivos CSV. As principais atividades incluem leitura de arquivos, criação de subconjuntos, configuração de visualização de dados, tratamento de valores ausentes e formatação de colunas.

### **2 OBJETIVOS**

- Ler um arquivo CSV utilizando a biblioteca Pandas;
- Criar subconjuntos de dados a partir do conjunto original;
- Configurar o número máximo de linhas exibidas na visualização;
- Exibir as primeiras e últimas "N" linhas do conjunto de dados;
- Apresentar informações gerais sobre as colunas, linhas e dados;
- Realizar tratamento de dados ausentes e formatar colunas corretamente.

### **3 DESENVOLVIMENTO**

#### **3.1 Leitura de arquivo CSV**

O arquivo CSV foi lido utilizando o Pandas, garantindo a correta interpretação dos dados.

```
import pandas as pd
```

```
df = pd.read_csv('dados.csv', sep=';', engine='python',  
encoding='utf-8')  
print(df.head())
```

### 3.2 Criação de subconjuntos

Foi gerado um subconjunto contendo somente as colunas ID, Duration e Calories.

```
df_sub = df[['ID', 'Duration', 'Calories']]  
print(df_sub.head())
```

### 3.3 Configuração de visualização

A configuração do Pandas foi ajustada para exibir até 9999 linhas.

```
pd.set_option('display.max_rows', 9999)  
print(df.to_string())
```

### 3.4 Exibição de linhas

Foram exibidas as 10 primeiras e últimas linhas do conjunto de dados.

```
print(df.head(10))  
print(df.tail(10))
```

### 3.5 Informações gerais

Foi utilizada a função .info() para visualizar informações sobre o dataset.

```
print(df.info())
```

### 3.6 Tratamento de dados

- Substituição de valores nulos na coluna `Calories` por 0.

```
df['Calories'].fillna(0, inplace=True)
```

- Substituição de valores nulos na coluna `Date` por `NaN`, conversão para datetime e correção do formato incorreto.

```
df['Date'].replace('1900/01/01', pd.NA, inplace=True)  
df['Date'] = pd.to_datetime(df['Date'], errors='coerce')  
df['Date'] = df['Date'].astype('datetime64[ns]')
```

- Remoção de registros contendo valores nulos na coluna `Date`.

```
df.dropna(subset=['Date'], inplace=True)
```

#### **4 CONSIDERAÇÕES FINAIS**

O processo realizado garantiu a limpeza e formatação do conjunto de dados, tornando-o adequado para análises futuras. Por exemplo, esses dados podem ser utilizados para identificar padrões de consumo energético ao longo do tempo, prever tendências de comportamento baseado na duração das atividades registradas ou até mesmo servir como base para um modelo preditivo de consumo calórico em atividades físicas. Todas as microatividades foram cumpridas com êxito, demonstrando a habilidade de manipulação de dados com a biblioteca Pandas.