

HEROIS DOS DADOS

UMA JORNADA EM SQL E PYTHON



THIAGO DE JESUS

Heróis dos Dados - Uma Jornada em SQL e Python

TIAGO DE Jesus

INTRODUÇÃO AO PYTHON PARA CIÊNCIA DE DADOS

Por Que Python?

Python é uma linguagem poderosa e fácil de aprender, tornando-se a escolha ideal para cientistas de dados em todo o mundo. Com uma sintaxe clara e legível, permite que você se concentre na solução de problemas de dados, não na complexidade do código.

Exemplo Prático: Carregando Dados

Vamos começar importando dados de um arquivo CSV para análise:

```
Pyhton -  
  
# Carregar dados de um arquivo CSV  
dados = pd.read_csv('exemplo.csv')  
print(dados.head())
```

Este exemplo mostra como ler um arquivo CSV usando Pandas, uma biblioteca essencial em ciência de dados para manipulação e análise de dados.



01

Manipulação de Dados com Pandas

MANIPULAÇÃO DE DADOS COM PANDAS

Transformando Dados em Informações

Pandas não só lê dados, mas também permite manipulá-los de forma eficiente para preparar sua análise.

Exemplo Prático: Limpeza de Dados

Imagine que você quer remover colunas desnecessárias e linhas com valores nulos:

```
Pyhton -  
  
# Remover colunas desnecessárias  
dados = dados.drop(columns=['coluna_inutil'])  
  
# Remover linhas com valores nulos  
dados = dados.dropna()  
print(dados.head())
```

Esse código limpa seu conjunto de dados, removendo colunas e linhas que não contribuem para sua análise.

02

Visualização de Dados: A Arte de Ver Números

VISUALIZAÇÃO DE DADOS: A ARTE DE VER NÚMEROS

Gráficos que Falam

Visualizar dados é crucial para entender as tendências e padrões. Python oferece bibliotecas como Matplotlib e Seaborn para isso.

Exemplo Prático: Criando um Gráfico de Barras

Vamos criar um gráfico para visualizar a distribuição de uma variável:

```
Python

import matplotlib.pyplot as plt

dados['variavel_interesse'].value_counts().plot(kind='bar')
plt.title('Distribuição da Variável de Interesse')
plt.xlabel('Variável')
plt.ylabel('Frequência')
plt.show()
```

Este código gera um gráfico de barras mostrando a frequência de valores em uma coluna específica.

03

Machine Learning com Scikit-Learn

MACHINE LEARNING COM SCIKIT-LEARN

Previsões Baseadas em Dados

O Python facilita a implementação de algoritmos de machine learning com a biblioteca Scikit-Learn.

Exemplo Prático: Modelo de Regressão Linear

Suponha que você queira prever resultados baseados em dados históricos:

```
Pyhton

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

# Preparar dados
X = dados[['variavel_independente']]
y = dados['variavel_dependente']
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2)

# Treinar modelo
modelo = LinearRegression()
modelo.fit(X_train, y_train)

# Avaliar modelo
y_pred = modelo.predict(X_test)
print('Erro Quadrático Médio:', mean_squared_error(y_test, y_pred))
```

Este código treina um modelo de regressão linear e avalia seu desempenho com o erro quadrático médio.

Conclusão:

Agradecimentos

CONCLUSÃO: TORNE-SE UM HERÓI DOS DADOS

Dominar Python para ciência de dados não só capacita você a entender melhor os dados, mas também a fazer previsões valiosas e insights que podem guiar decisões importantes. Continue explorando e aprendendo para aprimorar suas habilidades e tornar-se um verdadeiro herói dos dados!



<https://github.com/sthjes>