# Python for Data Analysis: Real-Time Examples



This document provides real-time examples of Python coding for data analysis. It covers key topics such as data manipulation, visualization, and statistical analysis using Python. Each section includes code snippets for practical applications.

## 1. Loading Data

Before starting data analysis, you need to load your dataset. Python's pandas library is commonly used for this purpose.

Example:  
import pandas as pd  
  
# Load a CSV file  
data = pd.read\_csv('data.csv')  
  
# Display the first few rows  
print(data.head())

## 2. Data Cleaning

Data cleaning is essential to ensure the accuracy of your analysis. Tasks include handling missing values, removing duplicates, and converting data types.

Example:  
# Check for missing values  
print(data.isnull().sum())  
  
  
# Remove duplicate rows  
data.drop\_duplicates(inplace=True)

## 3. Exploratory Data Analysis (EDA)

EDA involves summarizing the main characteristics of the data and visualizing it.

Example:  
import matplotlib.pyplot as plt  
import seaborn as sns  
  
# Summary statistics  
print(data.describe())  
  
# Histogram of a specific column  
plt.hist(data['column\_name'], bins=10)  
plt.title('Histogram of Column Name')  
plt.show()  
  
# Correlation heatmap  
sns.heatmap(data.corr(), annot=True, cmap='coolwarm')  
plt.title('Correlation Heatmap')  
plt.show()

## 4. Data Visualization

Data visualization helps in understanding trends and patterns in the data. Libraries like Matplotlib and Seaborn are commonly used.

Example:  
# Line plot  
plt.plot(data['date'], data['value'])  
plt.title('Line Plot')  
plt.xlabel('Date')  
plt.ylabel('Value')  
plt.show()  
  
# Bar plot  
sns.barplot(x='category', y='value', data=data)  
plt.title('Bar Plot')  
plt.show()

## 5. Statistical Analysis

Perform statistical analysis to derive insights from the data.

Example:  
from scipy import stats  
  
# T-test  
t\_stat, p\_value = stats.ttest\_ind(data['group1'], data['group2'])  
print('T-statistic:', t\_stat)  
print('P-value:', p\_value)

## 6. Machine Learning Basics (Optional)

Using libraries like Scikit-learn, you can build and train machine learning models for predictive analysis.

Example:  
from sklearn.model\_selection import train\_test\_split  
from sklearn.linear\_model import LinearRegression  
  
# Split the data into train and test sets  
X = data[['feature1', 'feature2']]  
y = data['target']  
X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)  
  
# Train a linear regression model  
model = LinearRegression()  
model.fit(X\_train, y\_train)  
  
# Evaluate the model  
print('Model Coefficients:', model.coef\_)  
print('Model Intercept:', model.intercept\_)

File Link : <https://colab.research.google.com/drive/15lDtAwv5Oj8KROeWV6UmWzGF15mpU_eZ?usp=sharing>

Github Link : <https://github.com/Samu-git07/Excel>