Share on your Social Media

Data Science with Python Tutorial

Published On: August 9, 2024

Data Science with Python Tutorial

Data scientists are using Python as a programming language extensively. Because Python includes built-in mathematical libraries and functions, performing data analysis and calculating mathematical problems is made simpler. Learn comprehensively in our **data science with Python tutorial.**

[**Download Data Science with Python Tutorial PDF**](#)

Introduction to Data Science with Python

Insightful information and knowledge can be extracted from data using statistical and computational techniques in the related discipline of data science. Python is a well-liked and adaptable programming language that has gained popularity among data

Related Cour at SLA

Related Posts

Tableau Developer Salary in Chennai

Published On: October 12, 2024

Introduction A Tableau Developer designs, develops, and maintains dashboards and visualizations using Tableau software. Key...

scientists due to its flexibility, large library, and ease of use. We cover the following in this data science with Python tutorial:

- Overview of Data Science with Python
- Exploratory Analysis Using Pandas
- Data Wrangling Using Pandas

Overview of Data Science with Python

Python's readability, simplicity, and versatility make it a preferred language in data science. Data scientists may concentrate on solving problems rather than coding complexities due to its vast libraries and frameworks, which simplify complicated jobs. The most popular programming language in the world, Python is also quite easy to learn.

Data Science with Python Interview Questions

Important Python Libraries for Data Science

NumPy: A basic Python library for numerical operations that supports big, multi-dimensional arrays and matrices is called NumPy.

Pandas: An effective toolkit for data analysis and manipulation that provides data structures like DataFrames for managing structured data.

Scikit-learn: An extensive machine learning package that offers effective and user-friendly tools for data mining and analysis.

Matplotlib and Seaborn: Software tools for generating static, animated, and interactive data visualizations that facilitate the identification of trends and patterns in the data.

Basic Cocnepts of Data Science with Python

Below are the important data science concepts:

Data Exploration: Examining datasets to comprehend their organization, key components, and possible links is

---

VMware Tutorial for Cloud Computing Aspirants
Published On: October 12, 2024

VMware Tutorial for Cloud Computing Aspirants VMware software allows you to run a virtual machine...

VBA Macros Tutorial for Beginners
Published On: October 10, 2024

VBA Macros Tutorial for Beginners VBA macros are programs that automate repetitive operations in Microsoft...

VB.Net Tutorial for Beginners
Published On: October 10, 2024

VB.Net Tutorial for Beginners Visual Basic is an easy-to-learn programming language that is type-safe. Get...

known as data exploration.

- It involves using statistics to summarize data and charts and graphs to visualize it.
- Finding patterns, trends, and anomalies is an important part of this process since it provides information for additional study.

Data Cleaning: Data cleaning involves addressing missing information, fixing mistakes, and eliminating duplicates from raw data to prepare it for analysis.

- Clean data ensures accurate and trustworthy outcomes.
- Among the methods are normalization, outlier identification, and imputation for missing variables.

Data Visualization: By converting data into graphical formats, data visualization makes it easier to see correlations, patterns, and trends.

Strong libraries like Matplotlib and Seaborn are available for Python, which makes it possible to create a wide variety of visualizations, from simple line graphs to complex heatmaps.

Statistics: Data analysis has a mathematical underpinning with statistics.

Data can be summarized and inferred using fundamental statistical techniques, including mean, median, mode, standard deviation, and correlation coefficients.

Exploratory Analysis Using Pandas

An essential phase in the data science process is exploratory data analysis (EDA), which aids in understanding the primary features of the data before drawing any conclusions. For this, a potent Python package called Pandas is frequently utilized.

Data Science with Python Syllabus PDF

Step-by-Step Tutorial for Exploratory Analysis Using Pandas

Loading Data

Your data must first be loaded into a Pandas DataFrame. Numerous sources, including databases, Excel, and CSV files, can be used for this.

import pandas as pd

data = pd.read_csv('your_data_file.csv')

Viewing Data

To comprehend the structure of the data, it is imperative to scrutinize the initial few rows once they have been loaded.

print(data.head())

Comprehending Data Structures

Verify the column names, data types, and DataFrame dimensions.

print(data.shape)

print(data.columns)

print(data.dtypes)

Summary Statistics

To comprehend the variability, central tendency, and distribution of the data, generate summary statistics.

print(data.describe())

Missing Values

Missing values can interfere with your analysis and model performance, so find and fix them.

print(data.isnull().sum())

data_cleaned = data.dropna()

data_filled = data.fillna(method='ffill')

Data Distribution

Display the data distribution for each of the columns.

```
import matplotlib.pyplot as plt

data['column_name'].hist()

plt.title('Distribution of column_name')

plt.xlabel('Values')

plt.ylabel('Frequency')

plt.show()
```

Correlation Analysis

Correlation matrices can be used to understand correlations between numerical features.

```
correlation_matrix = data.corr()

print(correlation_matrix)
```

Group By and Aggregation

Run group by operations to obtain the aggregated information.

```
grouped_data = data.groupby('group_column').mean()

print(grouped_data)
```

[Data Science with Python Training](#)

Data Wrangling Using Pandas

Data wrangling is the process of converting and formatting raw data into a format that can be analyzed. It is sometimes referred to as data cleaning or munging. Pandas is a robust Python package offering many functions to simplify data manipulation.

Step-by-Step Tutorial to Data Wrangling Using Pandas

Loading Data

Your data must first be loaded into a Pandas DataFrame. A variety of sources, including databases, Excel files, and

CSV files, can be used for this.

```
import pandas as pd

data = pd.read_csv('your_data_file.csv')
```

Inspecting Data

Recognize the data's content and structure.

```
print(data.head())

print(data.shape)

print(data.columns)

print(data.dtypes)
```

Handling Missing Values

Determine and address any missing values.

```
print(data.isnull().sum())

data_cleaned = data.dropna()

data_filled = data.fillna(method='ffill')  # Forward fill
```

Removing Duplicates

Find and eliminate duplicate rows.

```
print(data.duplicated().sum())

data = data.drop_duplicates()
```

Data Type Conversion

Change the columns' data types to the proper ones.

```
data['date_column'] =
pd.to_datetime(data['date_column'])

data['category_column'] =
data['category_column'].astype('category')

data['numeric_column'] =
pd.to_numeric(data['numeric_column'],
errors='coerce')
```

Renaming Columns

To make columns easier to read, rename them.

*data.rename(columns={'old_name': 'new_name',
'another_old_name': 'another_new_name'},
inplace=True)*

Filtering Data

Sort data according to criteria.

filtered_data = data[data['column_name'] > value]

*filtered_data = data[(data['column1'] > value1) &
(data['column2'] == 'value2')]*

Handling Categorical Data

If necessary, transform categorical data into numerical
representation.

*data = pd.get_dummies(data, columns=
['categorical_column'])*

*data['categorical_column'] =
data['categorical_column'].astype('category').cat.codes*

Creating New Columns

Take the current data and create new columns.

*data['new_column'] = data['column1'] +
data['column2']*

*data['new_column'] =
data['existing_column'].apply(lambda x: x * 2)*

Data Aggregation

Utilizing group by operations, aggregate data.

grouped_data = data.groupby('group_column').mean()

print(grouped_data)

Conclusion

We've covered the essential ideas in this data science
with Python tutorial and some useful examples to get

you going. We invite you to explore Python's endless opportunities and start your data science journey with our **data science with Python training in Chennai.**

Share on your Social Media

Softlogic Academy

Softlogic Systems

KK Nagar [Corporate Office]

No.10, PT Rajan Salai, K.K. Nagar, Chennai – 600 078.
Landmark: Karnataka Bank Building
Phone: +91 86818 84318
Email: enquiry@softlogicsys.in
Map: Google Maps Link

OMR

No. E1-A10, RTS Food Street
92, Rajiv Gandhi Salai (OMR),
Navalur, Chennai - 600 130.
Landmark: Adj. to AGS Cinemas
Phone: +91 89256 88858
Email: info@softlogicsys.in
Map: Google Maps Link

Courses

Python

Software Testing

Navigation

About Us

Blog Posts

Careers

Contact

Placement Training

Corporate Training

Hire With Us

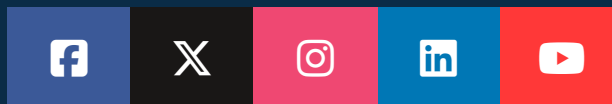Job Seekers

SLA's Recently Placed Students

Reviews

Sitemap

Important Links

Disclaimer

Privacy Policy

Terms and Conditions

Social Media Links

Full Stack Developer

Java

Power BI

Clinical SAS

Data Science

Embedded

Cloud Computing

Hardware and Networking

VBA Macros

Mobile App Development

DevOps

Review Sources

Google

Trustpilot

Glassdoor

Mouthshut

Sulekha

Justdial

Ambitionbox

Indeed

Software Suggest

Sitejabber