Data Structures in Pandas

kumar_satyam

Pandas is an open-source library that uses for working with relational or labeled data both easily and intuitively. It provides various data structures and operations for manipulating numerical data and time series. It offers a tool for cleaning and processes your data. It is the most popular Python library that is used for data analysis. In this article, We are going to learn about Pandas Data structure.

It supports two data structures:

- Series
- Dataframe

Series

Pandas is a one-dimensional labeled array and capable of holding data of any type (integer, string, float, python objects, etc.)

Syntax: *pandas.Series(data=None, index=None, dtype=None, name=None, copy=False, fastpath=False)*

Parameters:

- ***data****: array- Contains data stored in Series.*
- ***index****: array-like or Index (1d)*
- ***dtype****: str, numpy.dtype, or ExtensionDtype, optional*
- ***name****: str, optional*

- ***copy**: bool, default False*

Example 1: Series holding the char data type.

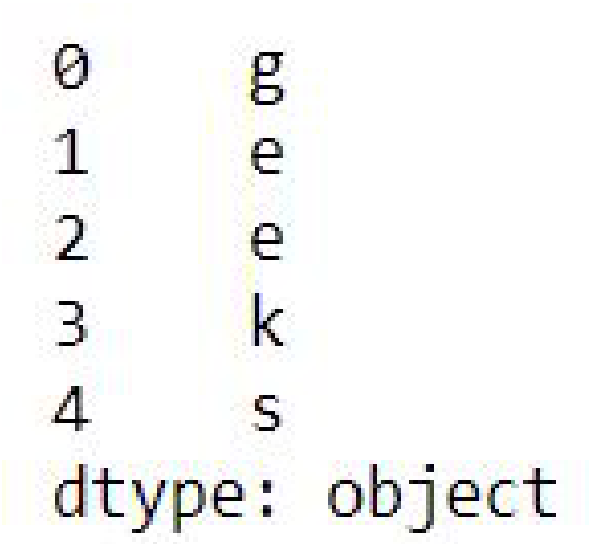
Python3

```
import pandas as pd

# a simple char list
list = ['g', 'e', 'e', 'k', 's']

# create series form a char list
res = pd.Series(list)
print(res)
```

Output:



Example 2: Series holding the Int data type.

Python3

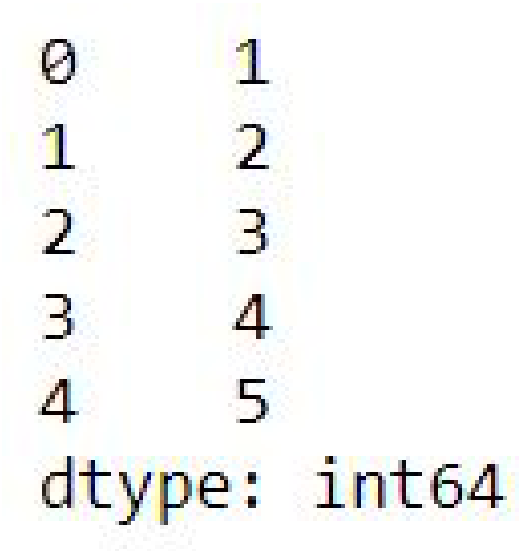
```
import pandas as pd

# a simple int list
```

```
list = [1,2,3,4,5]

# create series form a int list
res = pd.Series(list)
print(res)
```

Output:



Example 3: Series holding the dictionary.

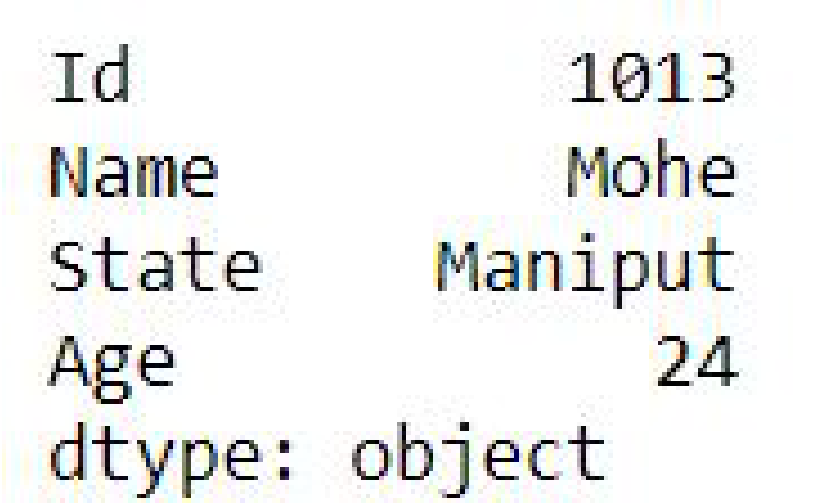
Python3

```
import pandas as pd

dic = { 'Id': 1013, 'Name': 'MOhe',
       'State': 'Maniput','Age': 24}

res = pd.Series(dic)
print(res)
```

Output:



Dataframe

Pandas DataFrame is a two-dimensional size-mutable, potentially heterogeneous tabular data structure with labeled axes (rows and columns). A Data frame is a two-dimensional data structure, i.e., data is aligned in a tabular fashion in rows and columns like a spreadsheet or SQL table, or a dict of Series objects. . Pandas DataFrame consists of three principal components, the **data**, **rows**, and **columns**.

Creating a Pandas DataFrame

In the real world, a Pandas DataFrame will be created by loading the datasets from existing storage, storage can be SQL Database, CSV file, and Excel file. Pandas DataFrame can be created from the lists, dictionary, and from a list of dictionary etc. Dataframe can be created in different ways here are some ways by which we create a dataframe:

Example 1: DataFrame can be created using a single list or a list of lists.

Python3

```
# import pandas as pd
import pandas as pd

# list of strings
lst = ['Geeks', 'For', 'Geeks', 'is',
            'portal', 'for', 'Geeks']

# Calling DataFrame constructor on list
df = pd.DataFrame(lst)
display(df)
```

Output:

	0
0	Geeks
1	For
2	Geeks
3	is
4	portal
5	for
6	Geeks

Example 2: Creating DataFrame from dict of ndarray/lists.

To create DataFrame from dict of narray/list, all the narray must be of same length. If index is passed then the length index should be equal to the length of arrays. If no index is passed, then by default, index will be range(n) where n is the array length.

Python3

```
# Python code demonstrate creating
# DataFrame from dict narray / lists
# By default addresses.

import pandas as pd

# initialise data of lists.
data = {'Name':['Tom', 'nick', 'krish', 'jack'],
        'Age':[20, 21, 19, 18]}

# Create DataFrame
df = pd.DataFrame(data)

# Print the output.
display(df)
```

Output:

	Name	Age
0	Tom	20
1	nick	21
2	krish	19
3	jack	18

Dealing with a column and row in DataFrame

Selection of column: In Order to select a column in Pandas DataFrame, we can either access the columns by calling them by their columns name.

Python3

```
# Import pandas package
import pandas as pd

# Define a dictionary containing employee data
data = {'Name':['Jai', 'Princi', 'Gaurav', 'Anuj'],
        'Age':[27, 24, 22, 32],
        'Address':['Delhi', 'Kanpur', 'Allahabad', 'Kannauj'],
        'Qualification':['Msc', 'MA', 'MCA', 'Phd']}

# Convert the dictionary into DataFrame
df = pd.DataFrame(data)

# select two columns
print(df[['Name', 'Qualification']])
```

Output:

	Name	Qualification
0	Jai	Msc
1	Princi	MA
2	Gaurav	MCA
3	Anuj	Phd

How to Select Rows and Column from Pandas DataFrame?

Example 1: Selecting rows.

pandas.DataFrame.loc is a function used to select rows from Pandas DataFrame based on the condition provided.

> ***Syntax:*** *df.loc[df['cname'] 'condition']*
>
> ***Parameters:***
>
> - ***df:*** *represents data frame*
> - ***cname:*** *represents column name*
> - ***condition:*** *represents condition on which rows has to be selected*

Python3

```
# Importing pandas as pd
from pandas import DataFrame

# Creating a data frame
Data = {'Name': ['Mohe', 'Shyni', 'Parul', 'Sam'],
        'ID': [12, 43, 54, 32],
        'Place': ['Delhi', 'Kochi', 'Pune', 'Patna']
       }

df = DataFrame(Data, columns = ['Name', 'ID', 'Place'])

# Print original data frame
print("Original data frame:\n")
display(df)
```

```
# Selecting the product of Electronic Type
select_prod = df.loc[df['Name'] == 'Mohe']

print("\n")

# Print selected rows based on the condition
print("Selecting rows:\n")
display (select_prod)
```

Output:

```
Original data frame:
```

	Name	ID	Place
0	Mohe	12	Delhi
1	Shyni	43	Kochi
2	Parul	54	Pune
3	Sam	32	Patna

```
Selecting rows:
```

	Name	ID	Place
0	Mohe	12	Delhi

Example 2: Selecting column.

Python3

```
# Importing pandas as pd
from pandas import DataFrame

# Creating a data frame
Data = {'Name': ['Mohe', 'Shyni', 'Parul', 'Sam'],
        'ID': [12, 43, 54, 32],
        'Place': ['Delhi', 'Kochi', 'Pune', 'Patna']
       }

df = DataFrame(Data, columns = ['Name', 'ID', 'Place'])

# Print original data frame
print("Original data frame:")
display(df)

print("Selected column: ")
display(df[['Name', 'ID']] )
```

Output:

Original data frame:

	Name	ID	Place
0	Mohe	12	Delhi
1	Shyni	43	Kochi
2	Parul	54	Pune
3	Sam	32	Patna

Selected column:

	Name	ID
0	Mohe	12
1	Shyni	43
2	Parul	54
3	Sam	32

Last Updated : 18 Oct, 2021

Similar Reads

1. Converting nested JSON structures to Pandas DataFrames
2. Inbuilt Data Structures in Python
3. User Defined Data Structures in Python
4. scipy.spatial - Spatial data structures and algorithms
5. Python Data Structures
6. Python Data Structures and Algorithms
7. Python | pandas.to_markdown() in Pandas
8. Add a Pandas series to another Pandas series
9. Python Pandas - pandas.api.types.is_file_like() Function

10. Python | Data Comparison and Selection in Pandas

Related Tutorials

1. Data Analysis Tutorial
2. Flask Tutorial
3. Natural Language Processing (NLP) Tutorial
4. Data Science for Beginners
5. Data Science With Python Tutorial

Previous Next

Article Contributed By :

kumar_satyam
kumar_satyam

Vote for difficulty

Current difficulty : Hard

Easy | Normal | Medium | Hard | Expert

Improved By : varshagumber28, sagartomar9927

Article Tags : Python pandas-datatypes, Python-pandas, Python

Practice Tags : python

Improve Article | Report Issue

feedback@geeksforgeeks.org

Database Management System

Software Engineering

Digital Logic Design

Engineering Maths

Python Tkinter

OpenCV Python Tutorial

Python Interview Question

Data Science & ML

Data Science With Python

Data Science For Beginner

Machine Learning Tutorial

Maths For Machine Learning

Pandas Tutorial

NumPy Tutorial

NLP Tutorial

Deep Learning Tutorial

DevOps

Git

AWS

Docker

Kubernetes

Azure

GCP

Competitive Programming

Top DSA for CP

Top 50 Tree Problems

Top 50 Graph Problems

Top 50 Array Problems

Top 50 String Problems

Top 50 DP Problems

Top 15 Websites for CP

System Design

What is System Design

Monolithic and Distributed SD

Scalability in SD

Databases in SD

High Level Design or HLD

Low Level Design or LLD

Top SD Interview Questions

Interview Corner

Company Preparation

Preparation for SDE

Company Interview Corner

Experienced Interview

Internship Interview

Competitive Programming

Aptitude

GfG School

CBSE Notes for Class 8

CBSE Notes for Class 9

CBSE Notes for Class 10

CBSE Notes for Class 11

CBSE Notes for Class 12

English Grammar

Commerce

Accountancy

Business Studies

Microeconomics

UPSC

Polity Notes

Geography Notes

History Notes

Macroeconomics

Statistics for Economics

Indian Economic Development

Science and Technology Notes

Economics Notes

Important Topics in Ethics

UPSC Previous Year Papers

SSC/ BANKING

SSC CGL Syllabus

SBI PO Syllabus

SBI Clerk Syllabus

IBPS PO Syllabus

IBPS Clerk Syllabus

Aptitude Questions

SSC CGL Practice Papers

Write & Earn

Write an Article

Improve an Article

Pick Topics to Write

Write Interview Experience

Internships

Video Internship