Pandas Tutorial

Pandas is a powerful and easy-to-use open-source tool built on top of the Menu Python programming language. It is useful for data analysis and manipulation. Python with Pandas is widely used in Statistics, Finance, Neuroscience, Economics, Web Analytics, Advertising, etc.

To work with data sets, clean them, and make them relevant for Data Science is what Pandas do. With that, easily load and read data sets in Excel, CSV, JSON, XML, etc. formats with Pandas and work on them. Easily clean the wrong format data, remove duplicates, and do other tasks with Pandas.

See more results like this page

el Format

DataFrames in Pandas

The Pandas DataFrame is a Two-

dimensional tabular data structure i

Men

table with rows and columns. The **DataFrame()** method is used for this purpose and has the following

- data: The data to be stored in the
 - Pandas DataFrameindex: The index values to be
 - provided for the resultant frame.columns: Set the column labels for

the resultant frame if data does not

- mention beforedtype: It is the datatype and only a
- single type is allowed.copy: To copy the input data
- In this lesson, we will see some examples to:
- - Create a Pandas DataFrame
 Access a group of rows or columns in a Pandas DataFrame
 - Access a group of rows or columns by

Men

DataFrame

- Create a Pandas DataFrame
- Access a group of rows or columns in a Pandas DataFrame
- Access a group of rows or columns by integer positions in a Pandas DataFrame
- Name your own indexes in a Pandas
- Iterating a DataFrame

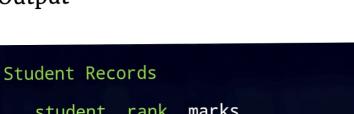
Create a Pandas DataFrame

To create a dataframe in pandas, use the pandas.DataFrame() method. Let us see an example wherein we have student records:

```
import pandas as pd
# Dataset
data
   'student': ["Amit", "John", "Jacob",
  'rank': [1, 4, 3, 5, 2],
'marks': [95, 70, 80, 60, 90]
df = pd.DataFrame(data)
```

print("Student Records\n\n",df)





rank marks student 95 Amit

4 70 John

0 1 2 3 4 80 Jacob 60 David

90 Steve

The 0, 1, 2, etc. are the index or label that

Menu

gets automatically added to the table.

Access a group of rows or columns in a Pandas DataFrame

Discover related topics	
Arrays	>
Combine	>
Python	>
Inverse Matrix	>
Python Pandas	>

The dataframe.loc is used in Pandas to access a group of rows or columns in a

access a group of rows or columns in a DataFrame. Let us see an example:

The dataframe.loc is used in Pandas to

import pandas as pd # Dataset Men data = {

'Student': ["Amit", "John", "Jacok 'Rank': [1, 4, 3, 5, 2], 'Marks': [95, 70, 80, 60, 90] }

Create a DataFrame using the DataFrame(df = pd.DataFrame(data, index=['RowA', print("Student Records\n\n",df) # Access the value in the student column
print("\nValue = ",df.loc['RowA', 'Studen

Output Student Records

Student Rank Marks RowA Amit RowB John RowC Jacob RowD David 1 95 4 70 80 60 RowE Steve 90 Value = Amit

Access a group of rows or

columns by integer positions in a Pandas DataFrame The **dataframe.iloc** is used to access a

group of rows or columns by integers We

세 세 📚 🗊 15:02 | 2.0KB/s 🏵 🏖 🎯 Access a group of rows or

columns by integer positions in a Pandas DataFrame

The dataframe.iloc is used to access a

group of rows or columns by integers. We have also set columns and indexes. I Men see an example:

import pandas **as** pd

Dataset

 $data = {$

Output

Marks

95

70

80

60

90

Marks

70

80

df = pd.DataFrame(data, index=['RowA', print("Student Records\n\n",df) # Access using rows and columns by intege
print("\nValue = \n",df.iloc[[1,2]])

Rank

1

4

Rank

4

3

Student Records

RowA

RowB

RowC

RowD

RowE

Value

RowB

RowC

Student

Amit

John

Jacob

David

Steve

Student John

Jacob

} # Create a DataFrame using the DataFrame(

'Student': ["Amit", "John", "Jacob", "D' 'Rank': [1, 4, 3, 5, 2], 'Marks': [95, 70, 80, 60, 90]

15:02 | 0.1KB/s ۞ ◘ الله الله ﴿ 15:02 | 0.1KB/s ۞ • الله الله ﴿ أَنَّ الله الله ﴿ وَاللَّهُ اللَّهُ الله الله

Name your indexes in a Pandas DataFrame

The **index** argument is used to set and name your indexes in a DataFrame. Let us see an example:

```
import pandas as pd

# Dataset
data = {
    'Student': ["Amit", "John", "Jacob", "Down', "Marks': [1, 4, 3, 5, 2],
    'Marks': [95, 70, 80, 60, 90]
}

# Create a DataFrame using the DataFrame(
# The index argument is used to set the identified of a pd.DataFrame(data, index=['Student1]
print("Student Records\n\n", df)
```

-

Output

Student Records Rank Marks Student 1 4 95 Amit Student1 70 John Student2 80 Jacob Student3 60 David Student4 90 Steve Student5

Iterate a DataFrame

To iterate a DataFrame and display the column names, use the for loop as in the below example:

```
import pandas as pd
# Dataset
data = {
     'Student': ["Amit", "John", "Jacob",
'Rank': [1, 4, 3, 5, 2],
'Marks': [95, 70, 80, 60, 90]
}
# Create a DataFrame using the DataFrame(
# The index argument is used to set
df = pd.DataFrame(data, index=['Stuc Men
print("Student Records\n\n", df)
# Iterating to display the columns
print("\nDisplaying the columns:")
for col in df:
    print(col)
Output
Student Records
                      Rank
                             Marks
            Student
                                95
             Amit
                         1
Student1
                                70
                         4
              John
Student2
                         3
5
2
            Jaco
David
                                80
Student3
                                60
Student4
                                90
Student5
Displaying the columns:
Student
Rank
Marks
```

If you liked the tutorial, spread the word and share the link and our website Studyopedia with others:

Tantidaca Join Our Vou Tubo Channel

For Videos, Join Our YouTube Channel: Join Now