

Day\_06 (Dataframe)

Data Frames

Definition

Pandas DataFrame is 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.

Purpose

DataFrames make manipulating your data easy, from selecting or replacing columns and indices to reshaping your data.

Important

Pandas DataFrame is a 2-D labeled data structure with columns of potentially different type. Just like excel, Pandas DataFrame provides various functionalities to analyze, change, and extract valuable information from the given dataset

Strengths

Python Data Analysis Library

pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language.

While pandas can certainly access data via SQL, or from several other data storage methods, its primary purpose is to make it easier when using Python to do data analysis.

To that end pandas has various methods available that allow some relational algebra operations that can be compared to SQL.

Also Pandas provides easy access to NumPy, which is the fundamental package for scientific computing with Python. It contains among other things:

- a powerful N-dimensional array object
- sophisticated (broadcasting) functions
- tools for integrating C/C++ and Fortran code
- useful linear algebra, Fourier transform, and random number capabilities

Weakness

Disadvantages:

Pandas does not persist data. It even has a (slow) function called TO\_SQL that will persist your pandas data frame to an RDBMS table.

Pandas will only handle results that fit in memory, which is easy to fill. You can either use dask to work around that, or you can work on the data in the RDBMS (which uses all sorts of tricks like temp space) to operate on data that exceeds RAM.

Example 43 :

Task:

create a Dataframe from the dictionary

```
In [ ]: d = {
'one' : pd.Series([1., 2., 3.], index=['a', 'b', 'c']),
'two' : pd.Series([1., 2., 3., 4.], index=['a', 'b', 'c', 'd'])
}

In [ ]: df=pd.DataFrame(d)

In [ ]: d

In [ ]: df
```

Example 44 :

Task:

Create a Dataframe from dictionary and assign different indexes and column to the dataframe

```
In [ ]: pd.DataFrame(d, index=['d', 'b', 'a'])

In [ ]: pd.DataFrame(d, index=['d', 'b', 'a'], columns=['two', 'three'])

In [ ]: df.columns

In [ ]: df.index
```

Example 45 :

Task:

create dictionary from the two series then make a dataframe using that dictionary practicec and play with the series adn dictionaries inside Datafram

```
In [ ]: boys=pd.Series([22,24,55],index=["Muhammad Umair","Muhammad Usama","Muhammad Akram"])

In [ ]: girls=pd.Series([18,25,30,25,45],[ "Alina","Aneeza","Shahnaz","Sajida","Frazeen"])

In [ ]: new_Dict={
'Boys':boys,
'Girls':girls
}

In [ ]: pd.DataFrame(new_Dict)

In [ ]: a_Dict={
'Age':boys+girls,
}

In [ ]: pd.DataFrame(a_Dict)

In [ ]: boys+girls

In [ ]: students_Dict=pd.Series([22,24,55,18,25,30,25,45],index=["Muhammad Umair","Muhammad Usama","Muhammad Akram","Alina","Aneeza","Shahnaz","Sajida","Frazeen"])

In [ ]: students_Dict

In [ ]: new_Students_dictionary={
'Age':students_Dict
}

In [ ]: new_Students_dictionary

In [ ]: pd.DataFrame(new_Students_dictionary)

In [ ]: d = {
'one' : [1., 2., 3., 4.],
'two' : [4., 3., 2., 1.]
}

In [ ]: d

In [ ]: pd.DataFrame(d)

In [ ]: pd.DataFrame(d, index=['a', 'b', 'c', 'd'])

In [ ]: studentsDict

In [ ]: students_series=pd.Series(studentsDict)

In [ ]: dict_students={
'students':students_series
}

In [ ]: dict_students

In [ ]: pd.DataFrame(dict_students)

In [ ]: pd.DataFrame(students_Dict)

In [ ]: studentsDict

In [ ]: studentsDataFrame1=pd.DataFrame(studentsDict)
studentsDataFrame1

In [ ]: studentsDict2={ 'Student4':{ 'name': "Hamdan Ijaz", 'age':20, 'Department': "BSE", 'Semester':6},
'Student5':{ 'name': "Aaqib Munir", 'age':22, 'Department': "BSE", 'Semester':6},
'Student6':{ 'name': "Ammar Naveed", 'age':20, 'Department': "BSE", 'Semester':6}
}

In [ ]: studentsDataFrame2=pd.DataFrame(studentsDict2)
studentsDataFrame2

In [ ]: studentsDataFrame1.append(studentsDataFrame2)

In [ ]: d = {
'one' : [1., 2., 3., 4.],
'two' : [4., 3., 2., 1.]
}

In [ ]: pd.DataFrame(d)

In [ ]: pd.DataFrame(d, index=['a', 'b', 'c', 'd'])
```

Example 46 :

Task:

create a dataframe of list of students dictionaries

```
In [ ]: List_Students_Dict=[
{'name': "Muhammad Umair", 'age':20, 'Department': "BSE", 'Semester':6},
{'name': "Hashim Shakoor", 'age':22, 'Department': "BSE", 'Semester':6},
{'name': "Muhammad Abdullah Tahir", 'age':20, 'Department': "BSE", 'Semester':6},
{'name': "Hamdan Ijaz", 'age':20, 'Department': "BSE", 'Semester':6},
{'name': "Aaqib Munir", 'age':22, 'Department': "BSE", 'Semester':6},
{'name': "Ammar Naveed", 'age':20, 'Department': "BSE", 'Semester':6}
]

In [ ]: pd.DataFrame(List_Students_Dict)

In [ ]: pd.DataFrame(List_Students_Dict,index=['005', '102', '106', '054', '056', '108'])
```

Example 47 :

Task:

Create a pandas Dataframe from list of tuples

```
In [ ]: pd.DataFrame(((('a', 'b'): {('A', 'B'): 1, ('A', 'C'): 2},
('a', 'a'): {('A', 'C'): 3, ('A', 'B'): 4},
('a', 'c'): {('A', 'B'): 5, ('A', 'C'): 6},
('b', 'a'): {('A', 'C'): 7, ('A', 'B'): 8},
('b', 'b'): {('A', 'D'): 9, ('A', 'B'): 10}))

In [ ]:
```