Data Exploration

April 30, 2023

SUBMITTED TO:-Ms.A.SOFIA MADAM.
SUBMITTED BY:-
NAME:Shaik Chinthapalli.parveen.
ROLL NUMBER:21VV1A1258.

1 Aim:-

Write a python program to import pandas.

1.1 Description:-

Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data. The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008. Pandas allows us to analyze big data and make conclusions based on statistical theories. Pandas can clean messy data sets, and make them readable and relevant. Relevant data is very important in data science.

1.2 PROGRAM:-

```
import pandas as pd
data = {'Name': ['John', 'Jane', 'Bob', 'Anna'],
'Age': [25, 30, 40, 20],
'Gender': ['Male', 'Female', 'Male', 'Female']}
df = pd.DataFrame(data)
print(df)
```

1.3 Expected output:-

```
>>> df = pd.DataFrame(data)
>>> print(df)
   Name  Age  Gender
0  John   25    Male
1  Jane   30  Female
2   Bob   40    Male
3  Anna   20  Female
```

1.4 Observed output:-

```
>>> df = pd.DataFrame(data)
>>> print(df)
   Name  Age  Gender
0  John   25    Male
1  Jane   30  Female
2   Bob   40    Male
3  Anna   20  Female
```

2 Aim:-

Write a pyhon program to loading the data various format(.XLS,.TXT,.CSV,JSON)using pandas.

2.1 Description:-

Ah, the good old CSV format. A CSV (or Comma Separated Value) file is the most common type of file that a data scientist will ever work with. These files use a "," as a delimiter to separate the values and each row in a CSV file is a data record. These are useful to transfer data from one application to another and is probably the reason why they are so commonplace in the world of data science. A file format is a standard way in which information is encoded for storage in a file. First, the file format specifies whether the file is a binary or ASCII file. Second, it shows how the information is organized. For example, comma-separated values (CSV) file format stores tabular data in plain text.

To identify a file format, you can usually look at the file extension to get an idea. For example, a file saved with name "Data" in "CSV" format will appear as "Data.csv". By noticing ".csv" extension we can clearly identify that it is a "CSV" file and data is stored in a tabular format.

2.2 PROGRAM:-

1.
import pandas
data ={'CHN': {'COUNTRY': 'China', 'POP': 1_398.72, 'AREA': 9_596.96, 'GDP': 12_234.78, 'CONT': 'Asia'},
'IND': {'COUNTRY': 'India', 'POP': 1_351.16, 'AREA': 3_287.26, 'GDP': 2_575.67, 'CONT': 'Asia', 'IND_DAY': '19470815'},
'USA': {'COUNTRY': 'US', 'POP': 329.74, 'AREA': 9_833.52, 'GDP': 19_485.39, 'CONT': 'N.America', 'IND_DAY': '17760704'},}
columns = ('COUNTRY', 'POP', 'AREA', 'GDP', 'CONT', 'IND_')
import pandas as pd
df = pd.DataFrame(data=data).T
df.to_csv('data.csv')
df = pd.read_csv('data.csv',index_col=0)
df
2.
import pandas as pd
df = pd.read_csv("C:311.csv")
df

2.3 Expected output:-

```
>>> df
    COUNTRY      POP     AREA       GDP       CONT     IND_DAY
CHN   China  1398.72  9596.96  12234.78       Asia         NaN
IND   India  1351.16  3287.26   2575.67       Asia  1947-08-15
USA      US   329.74  9833.52  19485.39  N.America  1776-07-04
```

2.4 Observed output:-

```
>>> df
     COUNTRY      POP     AREA       GDP       CONT     IND_DAY
CHN    China  1398.72  9596.96  12234.78       Asia         NaN
IND    India  1351.16  3287.26   2575.67       Asia  1947-08-15
USA       US   329.74  9833.52  19485.39  N.America  1776-07-04
```

3 Aim:-

describe data, modify data,grouping data,filtering data in python using pandas

3.1 Description:-

Filtering data from a data frame is one of the most common operations when cleaning the data. Pandas provides a wide range of methods for selecting data according to the position and label of the rows and columns. In addition, Pandas also allows you to obtain a subset of data based on column types and to filter rows with boolean indexing.

In this article, we will cover the most common operations for selecting a subset of data from a Pandas data frame: (1) selecting a single column by label, (2) selecting multiple columns by label, (3) selecting columns by data type, (4) selecting a single row by label, (5) selecting multiple rows by label, (6) selecting a single row by position, (7) selecting multiple rows by position, (8) selecting rows and columns simultaneously, (9) selecting a scalar value, and (10) selecting rows using Boolean selection.

3.2 PROGRAM:-

```
import pandas as pd
data = {'Name': ['John', 'Mary', 'Peter', 'Lisa', 'David'], 'Age': [25, 30, 27, 32, 28], 'Gender': ['M', 'F', 'M', 'F', 'M'], 'City': ['New York', 'Paris', 'Sydney', 'Tokyo', 'London']}
df = pd.DataFrame(data)
summary = df.describe()
print(summary)
df['Age'] = df['Age'] + 1
df.loc[2, 'City'] = 'Melbourne'
del df['Gender']
grouped_data = df.groupby('City')
summary = grouped_data.agg({'Age': 'mean'})
print(summary)
filtered_data = df.loc[df['Age'] ¿ 28]
print(filtered_data)
```

3.4 Observed output:-

```
             Age
count   5.000000
mean   28.400000
std     2.701851
min    25.000000
25%    27.000000
50%    28.000000
75%    30.000000
max    32.000000
            Age
City
London     29.0
Melbourne  28.0
New York   26.0
Paris      31.0
Tokyo      33.0
    Name  Age    City
1   Mary   31   Paris
3   Lisa   33   Tokyo
4  David   29  London
>>>
```

3.3 Expected output:-

```
             Age
count   5.000000
mean   28.400000
std     2.701851
min    25.000000
25%    27.000000
50%    28.000000
75%    30.000000
max    32.000000
            Age
City
London     29.0
Melbourne  28.0
New York   26.0
Paris      31.0
Tokyo      33.0
    Name  Age    City
1   Mary   31   Paris
3   Lisa   33   Tokyo
4  David   29  London
>>>
```

4 Aim:-

Converting a variable to adifferent data type back to a CSV,JSON,or SQL in python using pandas.

4.1 Description:-

Convert JSON to CSV using pandas in python? pandas is a library in python that can be used to convert JSON (String or file) to CSV file, all you need is first read the JSON into a pandas DataFrame and then write pandas DataFrame to CSV file.

The JSON stands for JavaScript Object Notation that is used to store and transfer the data between two applications. To use JSON in python you have to use Python supports JSON through a built-in package called JSON. To use this feature, we import the JSON package in Python script. The text in JSON is done through quoted-string which contains the value in key-value mapping within . It is similar to the dictionary in Python.

4.2 PROGRAM:-

```
import pandas as pd
import io
import sqlite3
sample_data = {
'Name': ['John', 'Jane', 'Alice', 'Bob'],
'Age': [25, 30, 35, 40],
'Salary': [50000, 60000, 70000, 80000]
}
df = pd.DataFrame(sample_data)
json_data = df.to_json()
df_from_json = pd.read_json(json_data)
csv_data = df.to_csv(index=False)
df_from_csv = pd.read_csv(io.StringIO(csv_data))
conn = sqlite3.connect('example.db')
df.to_sql('employee', conn, if_exists='replace', index=False)
df_from_sql = pd.read_sql('SELECT * FROM employee', conn)
print('\n Original DataFrame:\n', df)
print('\n DataFrame from JSON:\n', df_from_json)
print('\n DataFrame from CSV:\n', df_from_csv)
print('\n DataFrame from SQL:\n', df_from_sql)
```