Experiment No: 9

9. i)Write a Pandas program to create a date from a given year, month, day and another date from a given string formats.

Aim : To write a Pandas program to create a date from a given year, month, day and another date from a given string formats.

Description :

The **parser** module can parse datetime strings in many more formats. There can be no better library than **dateutil** to parse dates and times in Python. To lookup the timezones, the **tz** module provides everything. When these modules are combined, they make it very easy to parse strings into timezone-aware datetime objects.

All it takes is importing the parser module and calling the parse() function with a datetime string. The parser can return a sensible datetime object, but it cannot parse the string, it will raise a ValueError.

How it works :

- The parser instead of looking for recognizable tokens, guess what those tokens refer to. It doesn't use regular expressions.
- The order of these tokens matters as it uses a date format that looks like Month/Day/Year (the default order), while others use a Day/Month/Year format.
- The parse() function takes an optional keyword argument, dayfirst, which defaults to False to deal with this problem.
- It can correctly parse dates in the latter format if it is set to True.

Program :

```
from datetime import datetime

date1=datetime(year=2022,month=2,day=22)

print("Date from a given year, month, day")

print(date1)

from dateutil import parser

date2=parser.parse("3rd march, 2021")

print("Date from a given string format")

print(date2)
```

Output :

```
Date from a given year, month, day

2022-02-22 00:00:00

Date from a given string format

2021-03-03 00:00:00
```

9. ii)Write a Pandas program to create a time-series with two index labels and random values. Also print the type of the index.

Aim : To write a Pandas program to create a time-series with two index labels and random values. Also print the type of the index.

Description :

Pandas | Basic of Time Series Manipulation

Although time series is also available in scikit-learn but Pandas has some sort of compiled more features. In this module of Pandas, we can include the date and time for every record and can fetch the records of dataframe. We can find out the data within a certain range of date and time by using pandas module named **Time series**. Let's discuss some major objectives to introduce the pandas time series analysis.

Objectives of time series analysis

- Create the series of date
- Work with data timestamp
- Convert string data to timestamp
- Slicing of data using timestamp
- Resample your time series for different time period aggregates/summary statistics
- Working with missing data

Program :

```
import pandas as pd

import numpy as np

import datetime

from datetime import datetime, date

dates = [datetime(2021, 2, 1), datetime(2021, 2, 2)]

print("Time-series with two index labels:")

time_series = pd.Series(np.random.randn(2), dates)

print(time_series)

print("\nType of the index:")

print(type(time_series.index))
```

Output :

```
Time-series with two index labels:

2021-02-01    0.069974

2021-02-02    0.037410

dtype: float64


Type of the index:

<class 'pandas.core.indexes.datetimes.DatetimeIndex'>
```