Pandas:

Pandas is a Python library for data analysis. Started by McKinney in 2008 out of a need for a powerful and flexible quantitative analysis tool. Pandas is built on top of two core Python libraries—matplotlib for data visualization and NumPy for mathematical operations. Pandas is an open-source library that provides high-performance data manipulation in Python.

Benefits of Pandas :

Through its DataFrame and Series, it presents the data in a manner that is appropriate for data analysis.

1.Series:

A one-dimensional array capable of storing a variety of data types is how it is defined. The term "index" refers to the row labels of a series.

Example:

```
import pandas as pd
import numpy as np
info = np.array(['N','I','K','H','I','L'])
a = pd.Series(info)
print(a)
```

Output

```
0    N
1    I
2    K
3    H
4    I
5    L
```

2.DataFrame :

While series are useful, most analysts work with the majority of their data in DataFrames. DataFrames store data in the familiar table format of rows and columns, much like a spreadsheet or database.

It can be thought of as a series structure dictionary with indexed rows and columns. It is referred to as "columns" for rows and "index" for columns.

>>Example;

```
Import pandas as pd
x = [‘Nikhil’,’Avinash]
df = pd.DataFrame(x)
print(df)
```

Output

```
    0
0                                                   Nikhil
1   Avinash
```

>>functional requirements and non functional requirements:

1. Functional requirements:

- The functional requirements describe the behavior of the system as it correlates to the system's functionality.
- Functional requirements help to understand the functions of the system. Functional requirements are mandatory and easy to define.
- These are represented or stated in the form of input to be given to the system, the operation performed and the output expected.

- It helps us to verify the software's functionality and These requirements are specified by the user.
- Examples of the functional requirements are - Authentication of a user on trying to log in to the system.

2. Non functional requirements:

- Non-functional requirements are not related to the software's functional aspect. They are also called non-behavioral requirements.
- They help to understand the system's performance. It concentrates on the expectation and experience of the user.
- These requirements are specified by the software developers, architects, and technical persons.
- Basic non-functional requirements are - usability, reliability, security, storage, cost, flexibility, configuration, performance, legal or regulatory requirements, etc.

========== >>>> **To see the ports that are currently in use on a Linux system:**

1. Using netstat

-t: Displays TCP ports.
-u: Displays UDP ports.
-l: Shows only listening ports.
-n: Shows numeric addresses instead of resolving hostnames.

2. Using ss

```
ss -tuln
```

This command is similar to netstat but provides more detailed and faster information.

3. Using lsof

```
lsof -i -P -n
```

-i: Selects IPv4 and IPv6 files.
-P: Prevents the conversion of port numbers to port names.
-n: Shows numerical addresses.