

AIR QUALITY ANALYSIS IN TAMIL NADU

DATA ANALYSIS WITH COGNOS



TEAM LEADER - S.YOGESHWARI

TEAM MEMBER - D.ABINAYA

TEAM MEMBER - M.ABINAYA

TEAM MEMBER - M.AARTHI

PHASE-3

DEVELOPMENT PART

1.Import Libraries:

- First, make sure you have Python and pandas installed. You can install pandas using pip if you don't have it already

```
bash
pip install pandas

python
import pandas as pd
```

- A module can be imported into an interactive console environment .

```
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)
In [9]: pip install pandas
Requirement already satisfied: pandas in c:\users\elcot\documents\new folder\lib\site-packages (2.0.3)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\elcot\documents\new folder\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\elcot\documents\new folder\lib\site-packages (from pandas) (2023.3.post 1)
Requirement already satisfied: tzdata>=2022.1 in c:\users\elcot\documents\new folder\lib\site-packages (from pandas) (2023.3)
Requirement already satisfied: numpy>=1.21.0 in c:\users\elcot\documents\new folder\lib\site-packages (from pandas) (1.24.3)
Requirement already satisfied: six>=1.5 in c:\users\elcot\documents\new folder\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
Note: you may need to restart the kernel to use updated packages.

In [ ]: import pandas as pd
```

2.Load the Dataset:

- You need to have your air quality dataset in a compatible format, such as a CSV, Excel, or other tabular format. Let's assume your dataset is in a CSV file. You can load it using the `read_csv` function:

```
python Copy code  
  
# Replace 'your_dataset.csv' with the actual file path of your dataset.  
df = pd.read_csv('your_dataset.csv')
```

- Then click the blue Upload button displayed in the file's row to add the file to the project.

The image shows the JupyterLab interface. At the top, there's a 'jupyter' logo and 'Quit' and 'Logout' buttons. Below that, there are tabs for 'Files', 'Running', and 'Clusters'. Under the 'Files' tab, there are buttons for 'Duplicate', 'Rename', 'Move', 'Download', 'View', 'Edit', and a red trash icon. To the right of these buttons are 'Upload', 'New', and a refresh icon. The main area shows a file browser view of the 'Desktop' directory. It has a table with columns for 'Name', 'Last Modified', and 'File size'. The table lists several files and folders: '..' (seconds ago), 'New folder' (31 minutes ago), 'Data set.xlsx' (11 days ago, 149 kB, selected with a checkmark), 'GitHub Desktop.Ink' (a month ago, 2.12 kB), 'Visual Studio Code.Ink' (a month ago, 1.41 kB), and 'YouTube.Ink' (a month ago, 2.65 kB).

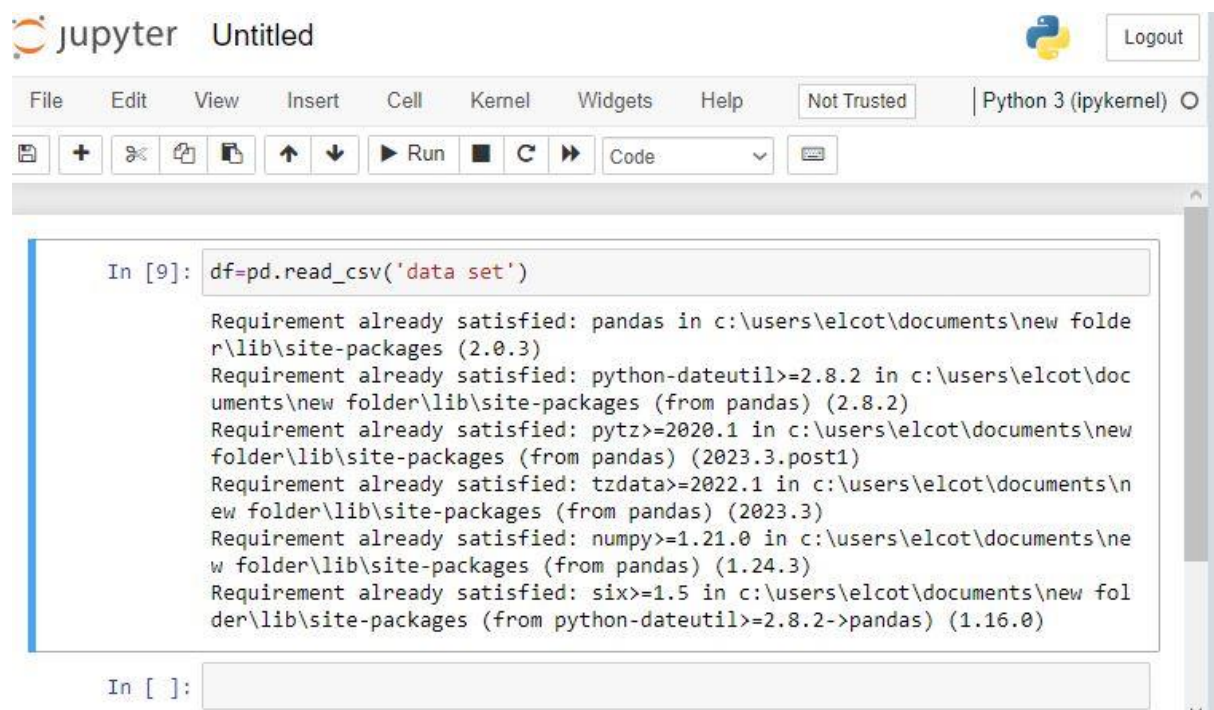
	Name	Last Modified	File size
<input type="checkbox"/>	..	seconds ago	
<input type="checkbox"/>	New folder	31 minutes ago	
<input checked="" type="checkbox"/>	Data set.xlsx	11 days ago	149 kB
<input type="checkbox"/>	GitHub Desktop.Ink	a month ago	2.12 kB
<input type="checkbox"/>	Visual Studio Code.Ink	a month ago	1.41 kB
<input type="checkbox"/>	YouTube.Ink	a month ago	2.65 kB

3.Explore the Dataset:

- Once you've loaded the dataset, you should explore it to understand its structure. You can start by checking the first few rows to get a glimpse of the data.

```
python Copy code  
  
# Display the first 5 rows of the dataset.  
print(df.head())
```

- Explore data set through the Object Storage Explorer and Table Explorer options on the left toolbar of this Lab interface.



4.Data pre-processing:

- Data pre-processing is essential to ensure the data is clean and ready for analysis. Common pre-processing steps include handling missing values, removing duplicates, and dealing with data types.

Handling Missing Value:

- You can check for missing values in your dataset using the '`isna()`' function and then decide how to handle them. For example, to count missing values in each column.

python


 Copy code

```
missing_values = df.isna().sum()
print(missing_values)
```

Date type conversions:

- Ensure that columns have the correct data types. For instance, dates should be converted to datetime objects if they are represented as strings.

python


 Copy code

```
# Convert a date column to datetime.
df['date_column'] = pd.to_datetime(df['date_column'])
```

Handling Duplicates:

Check for and remove duplicates if they exist in the dataset.

python

 Copy code

```
df = df.drop_duplicates()
```

```
File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

In [20]: import pandas as pd
         df=pd.read_csv('data set')
         print(df.head())

In [32]: missing_values = 75
         print(missing_values)

75

In [ ]: df = df.drop_duplicates()

In [49]: # Convert a date column to datetime.
         a=('date-column');
         b=('datetime');
```

```
jupyter Untitled (unsaved changes) Python 3 (ipykernel) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Requirement already satisfied: numpy>=1.21.0 in c:\users\elcot\documents\new fo
lder\lib\site-packages (from pandas) (1.24.3)
Requirement already satisfied: six>=1.5 in c:\users\elcot\documents\new folder
\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)

In [20]: import pandas as pd
         df=pd.read_csv('data set')
         print(df.head())

In [32]: missing_values = 75
         print(missing_values)

75

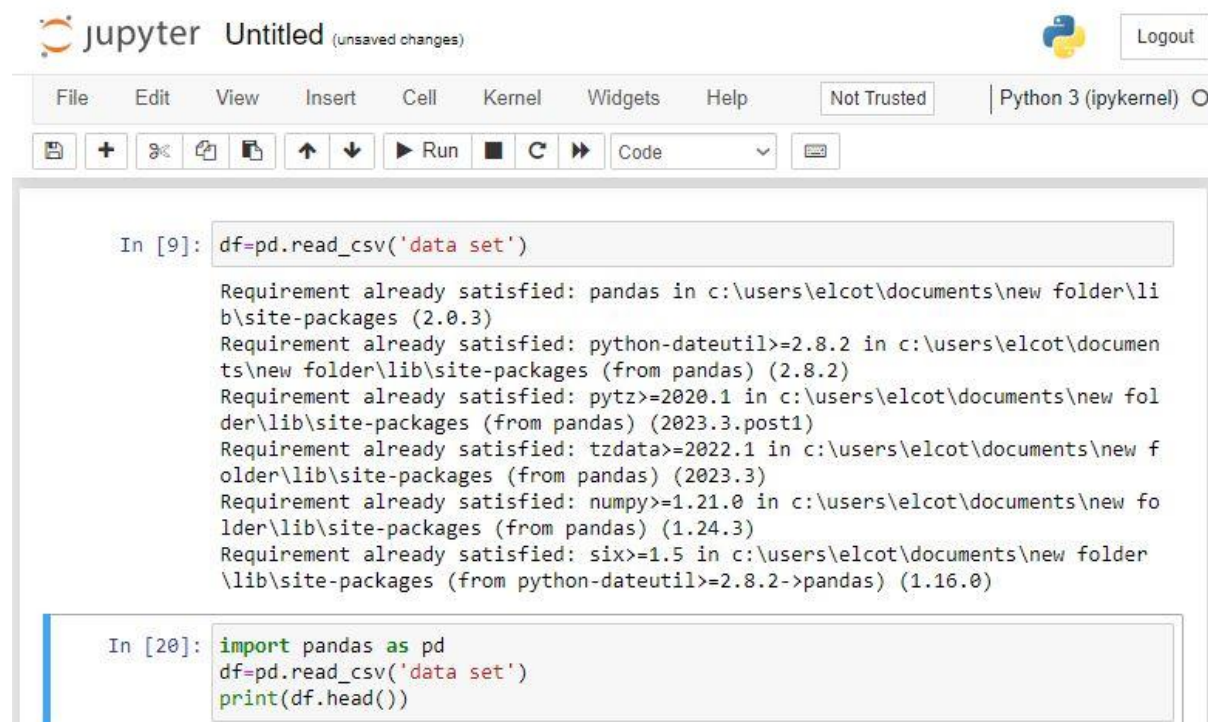
In [ ]: missing_values = df.isna().sum()
         print(missing_values)
```


5. Save pre-processed Data:

- After pre-processing, you may want to save the pre-processed dataset for future use:

```
python Copy code  
  
df.to_csv('preprocessed_data.csv', index=False)
```

Remember to replace 'your_dataset.csv' with the actual path to your air quality dataset and adapt the pre-processing steps to your specific data. If you encounter any issues or need further assistance, feel free to ask for help with specific aspects of pre-processing.



The image shows a Jupyter Notebook interface. At the top, there's a header with the Jupyter logo, the text 'jupyter Untitled (unsaved changes)', a Python logo, and a 'Logout' button. Below this is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. To the right of the menu bar are 'Not Trusted' and 'Python 3 (ipykernel)' labels. Below the menu bar is a toolbar with icons for saving, adding cells, undo, redo, running, and other functions. The main area of the notebook contains two code cells. The first cell, labeled 'In [9]:', contains the code `df=pd.read_csv('data set')`. Below this code, there is a large block of text showing system requirements for various packages: pandas, python-dateutil, pytz, tzdata, and numpy, all indicating they are already satisfied. The second cell, labeled 'In [20]:', contains the code `import pandas as pd`, `df=pd.read_csv('data set')`, and `print(df.head())`.

```
jupyter Untitled (unsaved changes) Python 3 (ipykernel) Logout  
  
File Edit View Insert Cell Kernel Widgets Help Not Trusted  
  
[Icons] Code  
  
In [9]: df=pd.read_csv('data set')  
  
Requirement already satisfied: pandas in c:\users\elcot\documents\new folder\lib\site-packages (2.0.3)  
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\elcot\documents\new folder\lib\site-packages (from pandas) (2.8.2)  
Requirement already satisfied: pytz>=2020.1 in c:\users\elcot\documents\new folder\lib\site-packages (from pandas) (2023.3.post1)  
Requirement already satisfied: tzdata>=2022.1 in c:\users\elcot\documents\new folder\lib\site-packages (from pandas) (2023.3)  
Requirement already satisfied: numpy>=1.21.0 in c:\users\elcot\documents\new folder\lib\site-packages (from pandas) (1.24.3)  
Requirement already satisfied: six>=1.5 in c:\users\elcot\documents\new folder\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)  
  
In [20]: import pandas as pd  
df=pd.read_csv('data set')  
print(df.head())
```

....End of the project Assessment....