# Jupyter notebook:

**Installation of Jupyter notebook in windows:**

Requirements to install Jupyter notebook are python to be pre installed.

Open windows PowerShell as administrator.

Run the following command to install Jupyter notebook.

```
pip install notebook
```

**Installing in mac or linux:**

Package manager is homebrew for macOS and Linux:

```
brew install jupyterlab
```

Once installed to run the notebook use the command:

```
jupyter notebook
```

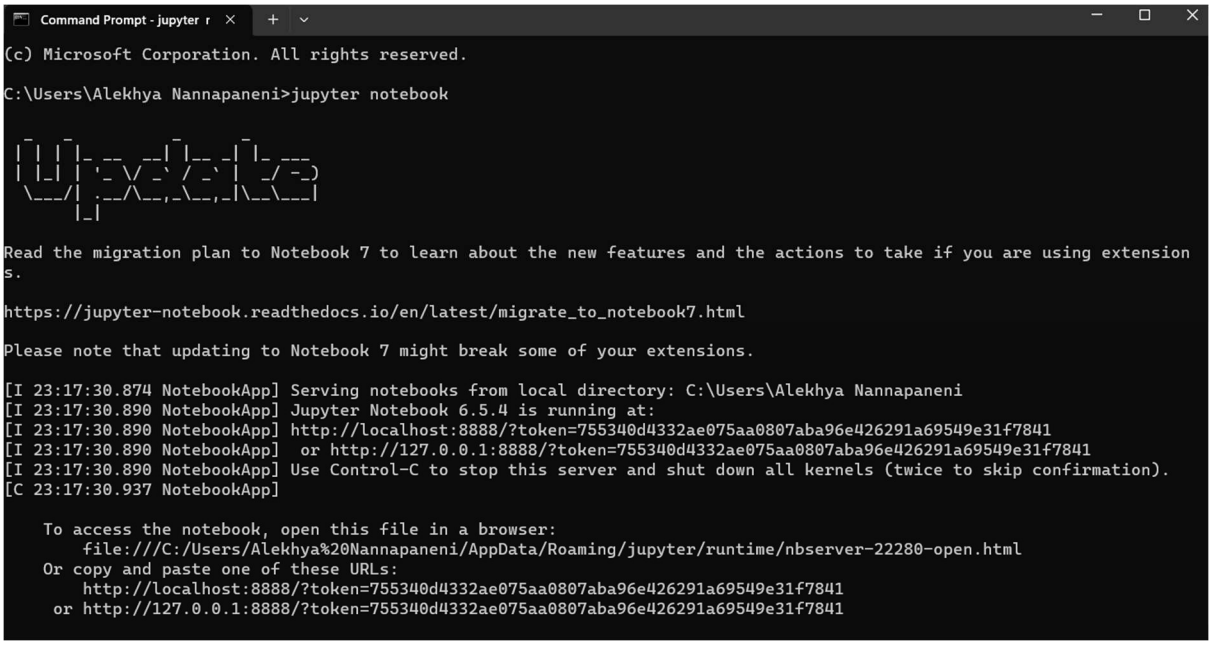


Fig 1: Running Jupyter notebook

This will open a jupyter notebook instance in the browser

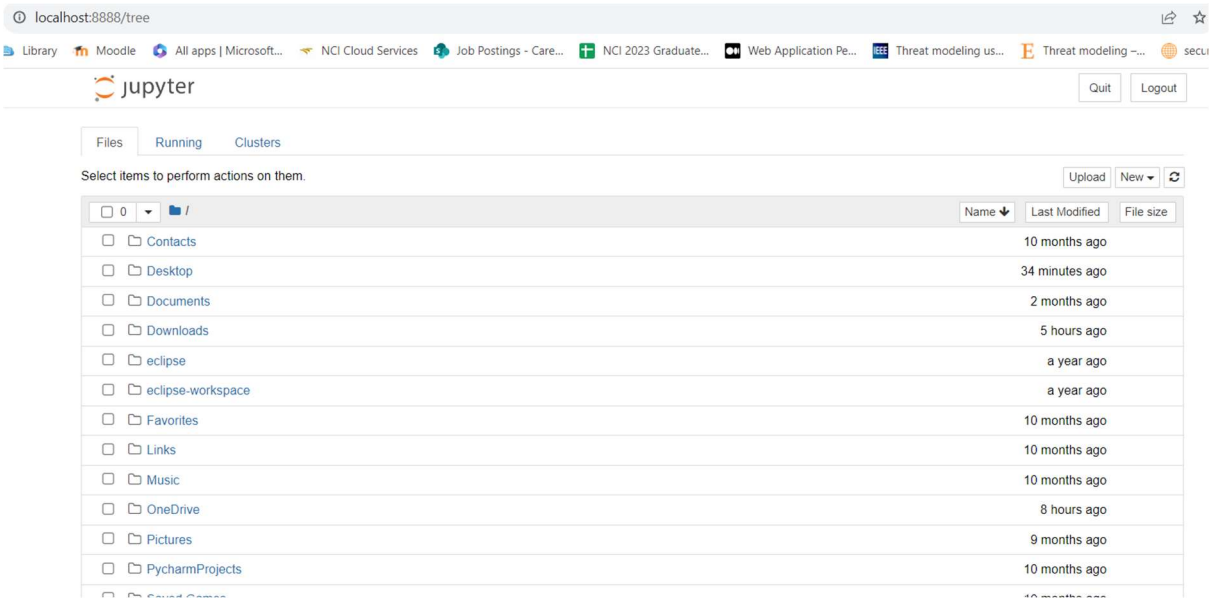


Fig 2: Jupyter instance

Once the instance opens we can open a command shell by using New→Python 3.

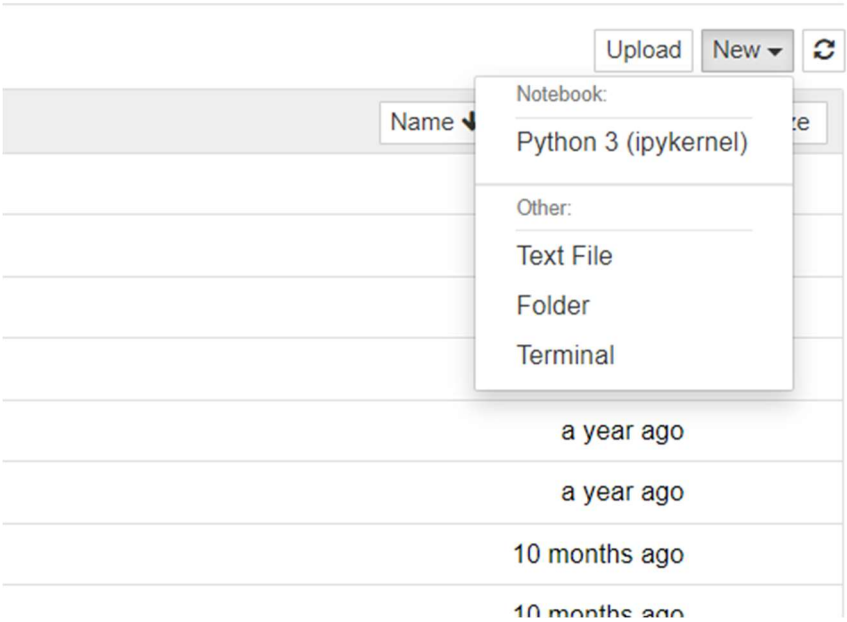


Fig 3: Jupyter command line

The code that is used to perform the data scrapping can be run on this instance or the file attached as thesis.ipynb can be run on this instance by opening the file with Jupyter notebook.

```
import pandas as pd

dataset_file = 'C:\\Users\\Alekhya Nannapaneni\\dataset.csv'

# Read the dataset and drop rows with missing values in specified columns

dataset = pd.read_csv(dataset_file).dropna(subset=['S.No', 'CVSS score', 'Component', 'Remediation',
'Name', 'Description', 'startdate'])

# List of columns to keep in the dataset

columns_to_keep = ['S.No', 'CVSS score', 'Component', 'Remediation']

# Drop all columns except the ones specified in the 'columns_to_keep' list

dataset = dataset[columns_to_keep]

#  filename to save the filtered dataset

save_path = 'C:\\Users\\Alekhya Nannapaneni\\desktop\\dataset.csv'

# Save the filtered dataset to a new CSV file

dataset.to_csv(save_path, index=False)

print("Dataset filtered and saved successfully.")
```

Fig 4: Code snippet

Here the dataset that is taken with name dataset.csv, the file should be placed in the directory where the Jupyter notebook is installed.In my case the path of file is “C:\\Users\\Alekhya Nannapaneni\\dataset.csv”.

And the path where the file saved after performing the operations mentioned is “C:\\Users\\Alekhya Nannapaneni\\desktop\\dataset.csv”.

These paths should be edited to the proper file paths in the local machine where the code will be running.

## Microsoft threat modelling tool:

Microsoft threat modelling tool can be installed from https://www.microsoft.com/en-US/download/details.aspx?id=49168. Downloading this tool is a manual process and easy. We need to follow the instructions provided at the time of installing. We can download the tool for free and is openly available. Once the tool is installed the file with name “New Threat Model” can be opened using it, which is a application architecture. The file with name “TM Final Full report” is the automated report generated from the threat modelling architecture “New Threat Model”.

**Draw.io:** Draw.io is used to prepare the architecture diagram of the application. The application structure is presented in the form of image in the main document. Here I am guiding on how to open the editable version of the application architecture.

Draw.io can be used online or installed on the local machine. We can browse to official page of draw.io https://app.diagrams.net/ and start preparing our diagram. In my case I have attached “TM thesis” file which can be opened using draw.io.

## OWASP ASVS document:

The file with the name “OWASP ASVS” has multiple tabs which are explained as reports and inputs used in the project. Here we can find the explanation of each tab and it’s purpose to have a clear understanding.

**OWASP ASVS Original:** This gives the standard questions given by OWASP foundation.

**Maturity**: this is used to represent the different levels of security requirements satisfied. These maturity levels are used in answering the questions given to validate the application security.

**Security assessment quest**: These tab gives the information of all the questions answered, this is used in preparing the risk assessment report. In the main document we referred this as security assessment questionnaire.

**TM**: This is the initial stage potential threats identified.

**Risk assessment report:** This shows the final stage threats identified using the threat intel dataset. These are explained in the main report.

**Threat categories:** Shows OWASP top 10 vulnerabilities categorization.

**Statistics:** Gives the overall application strength.

**TM vs TMI:** This is used to compare the results from threat modelling vs threat modelling performed using threat intelligence, explanation for this is written in the main document.

**Dataset:** The file with name “dataset” includes the threat intelligence data processed and analysed. This is prepared from open source internet.