231501134
RESHMATHI G

Experiment No: 1

Date: 16-07-2025

Setting up the Python environment and libraries Jupyter Notebook

Aim: To set up the Python environment using Jupyter Notebook, create and execute Python code cells, Markdown cells, and demonstrate the use of Jupyter Widgets and Jupyter AI for interactive programming.

Code:

# Step 1: Basic Python Execution print("Hello,

Jupyter Notebook!")

# Step 2: Using Markdown

# Experiment: Setting up the Python environment and libraries in Jupyter Notebook

## Aim

To set up the Python environment using Jupyter Notebook, create and execute Python code cells, Markdown cells, and demonstrate the use of Jupyter Widgets and Jupyter AI for interactive programming.

## Steps

1. Create a new Jupyter Notebook.
2. Add and run Python code cells.
3. Create Markdown cells for formatted text and documentation.
4. Import and use Python libraries like NumPy and Matplotlib.
5. Use Jupyter Widgets for interactivity.
6. Demonstrate Jupyter AI for AI-assisted queries.

## Result

- Successfully created and executed Python and Markdown cells.
- Plotted a sine wave using NumPy and Matplotlib.
- Used Jupyter Widgets to create an interactive slider.
- Demonstrated how Jupyter AI generates responses for AI-based queries.

# Step 3: Using Libraries

```
import numpy as np import
matplotlib.pyplot as plt

x = np.linspace(0, 10, 100) y
= np.sin(x)

plt.plot(x, y, label="sin(x)")
plt.title("Plot using Matplotlib in Jupyter")
plt.xlabel("X-axis") plt.ylabel("Y-axis")
plt.legend() plt.show()
```

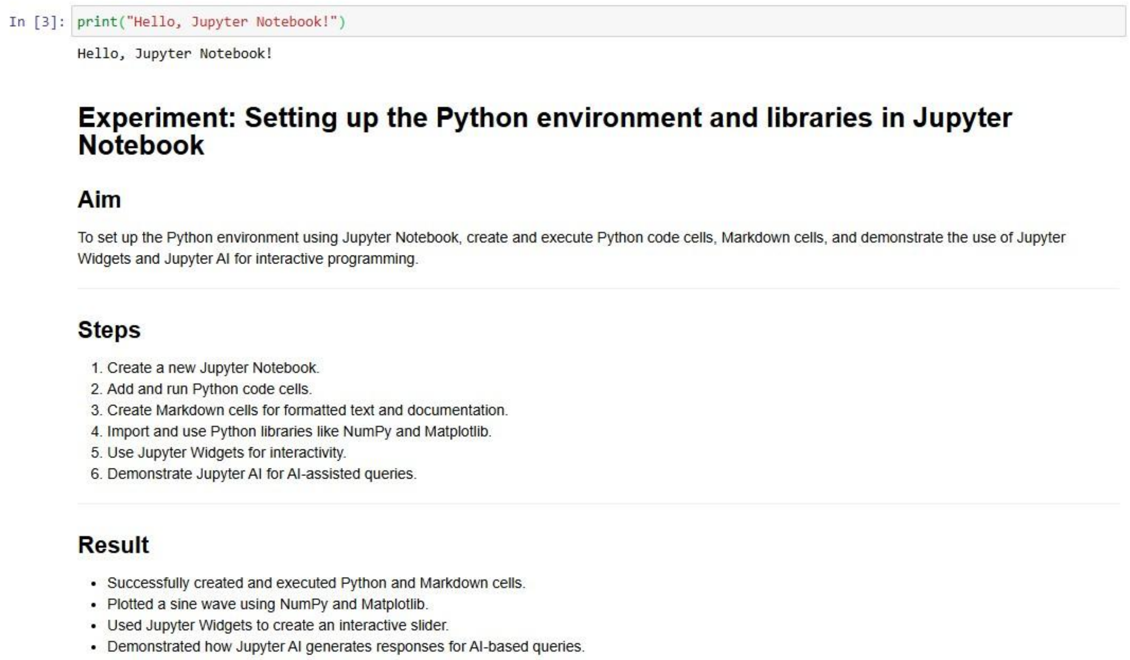
```
# Step 4: Jupyter Widgets from
ipywidgets import interact


def square(n):
    return f"The square of {n} is {n*n}"


interact(square, n=(1, 20));
```

231501134
RESHMATHI G

OUTPUT:



In [3]:
```
print("Hello, Jupyter Notebook!")
```

```
Hello, Jupyter Notebook!
```

Experiment: Setting up the Python environment and libraries in Jupyter Notebook

Aim

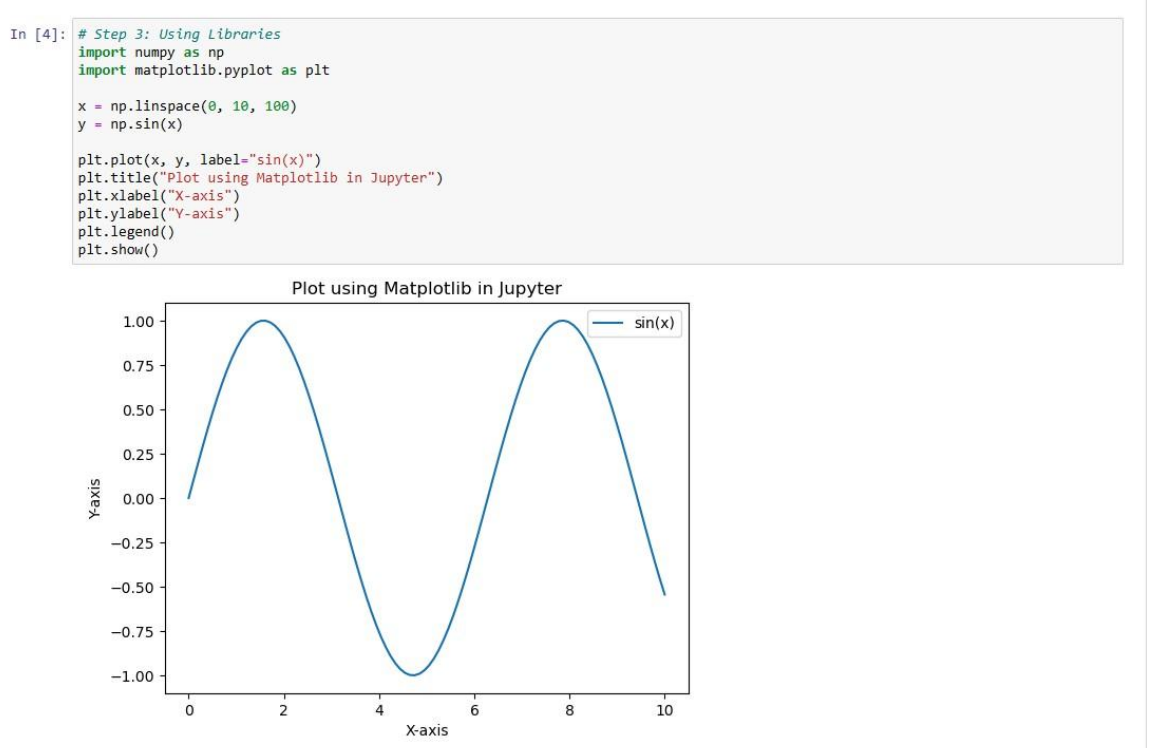
To set up the Python environment using Jupyter Notebook, create and execute Python code cells, Markdown cells, and demonstrate the use of Jupyter Widgets and Jupyter AI for interactive programming.

Steps

1. Create a new Jupyter Notebook.
2. Add and run Python code cells.
3. Create Markdown cells for formatted text and documentation.
4. Import and use Python libraries like NumPy and Matplotlib.
5. Use Jupyter Widgets for interactivity.
6. Demonstrate Jupyter AI for AI-assisted queries.

Result

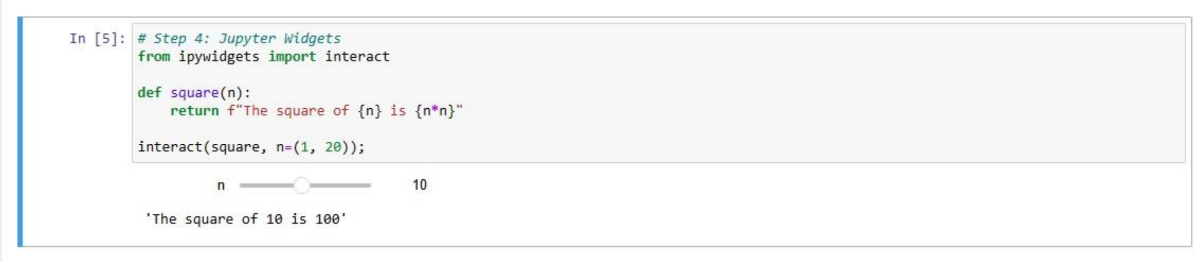
- Successfully created and executed Python and Markdown cells.
- Plotted a sine wave using NumPy and Matplotlib.
- Used Jupyter Widgets to create an interactive slider.
- Demonstrated how Jupyter AI generates responses for AI-based queries.



In [4]:
```
# Step 3: Using Libraries
import numpy as np
import matplotlib.pyplot as plt

x = np.linspace(0, 10, 100)
y = np.sin(x)

plt.plot(x, y, label="sin(x)")
plt.title("Plot using Matplotlib in Jupyter")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.legend()
plt.show()
```



```
In [5]: # Step 4: Jupyter Widgets
        from ipywidgets import interact

        def square(n):
            return f"The square of {n} is {n*n}"

        interact(square, n=(1, 20));
```

```
'The square of 10 is 100'
```

Result: Successfully set up the Python environment in Jupyter Notebook, executed code and Markdown cells, and demonstrated the use of libraries, widgets, and Jupyter AI.