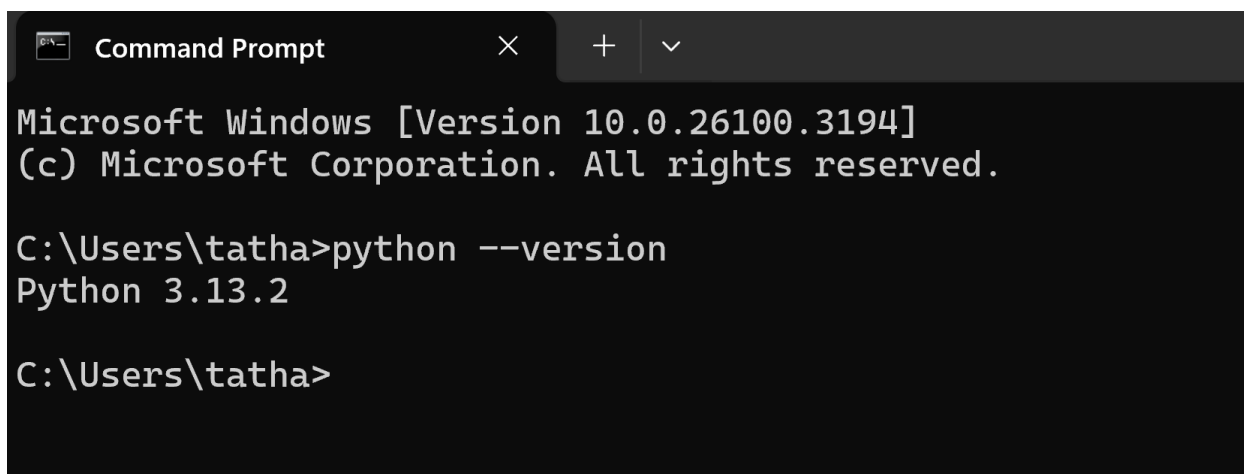


To install Jupyter Notebook on your Windows 11 computer, follow these steps:

Install Python (if not already installed)

1. Download the latest version of Python from [python.org](https://www.python.org/downloads/).
 1. <https://www.python.org/downloads/>
2. Run the installer, select **Add Python to PATH** (very important), and then click **Install Now**.
3. Once installed, verify by opening **Command Prompt** (search for cmd in the Start menu) and typing:

A screenshot of a Windows Command Prompt window. The title bar says "Command Prompt". The text inside shows the Windows version "Microsoft Windows [Version 10.0.26100.3194]" and copyright "(c) Microsoft Corporation. All rights reserved." The command prompt shows the user is in the directory "C:\Users\tatha" and has typed "python --version", which returned "Python 3.13.2". The prompt is now waiting for another command.

```
Microsoft Windows [Version 10.0.26100.3194]
(c) Microsoft Corporation. All rights reserved.

C:\Users\tatha>python --version
Python 3.13.2

C:\Users\tatha>
```

Install Jupyter Notebook using pip

1. Open **Command Prompt** or **Windows PowerShell**.

```
pip install notebook
```

To Update pip

```
python.exe -m pip install --upgrade pip
```

Ensure Python and Scripts Folder Are in PATH

1. Check if Python and the Scripts folder (where Jupyter installs its executables) are in your PATH environment variable.
2. To add them to the PATH:

- Open **Settings > System > About > Advanced system settings**.
- In the **System Properties** window, click on **Environment Variables**.
- Under **System variables**, find and select the **Path** variable, then click **Edit**.
- Add the following paths (adjust if Python is installed in a different directory):

C:\Users\tatha\AppData\Roaming\Python\Python313\

C:\Users\tatha\AppData\Roaming\Python\Python313\Scripts

- Click **OK** to save and close all windows.

Restart Command Prompt and Run Jupyter Notebook

1. Close and reopen **Command Prompt** (to reload the updated PATH).

jupyter notebook

This should open Jupyter Notebook in your browser. Let me know if the issue persists after these steps!

Create a New Notebook

- In the Jupyter Notebook interface, navigate to the folder where you want to save your notebook.
- Click on **New** (top-right corner) and select **Python 3**. This will open a new notebook with an empty cell.

Write and Run Your First Code

- In the first cell, type some Python code. For example:

```
print("Hello, world!")
```

- To run the cell:
 - Press **Shift + Enter** on your keyboard, or
 - Click the **Run** button in the toolbar (it looks like a play button).

View the Output

After running the cell, you should see the output directly below the cell:

Hello, world!

Additional Tips

- You can create more cells by clicking **Insert > Insert Cell Below** or pressing **B** on your keyboard (when in command mode).
 - To switch between command and edit modes, use **Esc** (command mode) and **Enter** (edit mode).
-

Example:

Create New Cells

You can write code in different cells for better organization. Here's how:

- **Insert New Cells:**
 - Click **Insert** in the menu and select **Insert Cell Below** (or **Insert Cell Above**).
 - Alternatively, while in command mode (blue border), press **B** to insert a cell below or **A** to insert a cell above.

Write Your Code

- **In Each Cell:** You can write a complete function, class, or any block of code. For example:

```
def greet(name):  
    return f"Hello, {name}!"  
  
print(greet("Alice"))
```

- **Long Scripts:** If you have a long script, you can split it across multiple cells to logically separate sections. For example:

- **Cell 1:** Import necessary libraries.

```
import numpy as np  
  
import pandas as pd
```
- **Cell 2:** Load and preprocess data.

```
data = pd.read_csv('data.csv')
```

```
cleaned_data = data.dropna()
```

- **Cell 3:** Perform analysis.

```
mean_values = cleaned_data.mean()
```

```
print(mean_values)
```

Run Your Code

- **Run Cells:** You can run a cell by selecting it and:
 - Pressing **Shift + Enter** to run the cell and move to the next one.
 - Pressing **Ctrl + Enter** to run the cell and stay in the same cell.

Markdown Cells for Comments

- If you want to add comments or explanations, you can use Markdown cells:
 - Click on a cell and change its type to Markdown from the drop-down menu in the toolbar or by pressing **M** in command mode.
 - Write your text, then run the cell to render the Markdown. For example:

```
## Data Analysis
```

This section performs data analysis on the cleaned dataset.

Save Your Work

- Remember to save your notebook frequently by clicking the **Save** icon (disk icon) in the toolbar or by pressing **Ctrl + S**.

Example Workflow

Here's a simple example workflow:

1. **Cell 1:** Import Libraries

```
import matplotlib.pyplot as plt
```

2. **Cell 2:** Define Functions

```
def plot_data(data):
```

```
    plt.plot(data)
```

```
    plt.show()
```

3. **Cell 3:** Load Data

```
data = [1, 2, 3, 4, 5]
```

4. **Cell 4:** Call Function to Plot

```
plot_data(data)
```

By organizing your code this way, you can write complex scripts while maintaining clarity and making debugging and understanding your work easier.

Install Matplotlib

1. Press Win + R, type cmd, and hit Enter to open Command Prompt.
2. Install Matplotlib:
 - In the Command Prompt, run the following command:

```
pip install matplotlib
```

3. **Verify the Installation:**

After installation, you can verify it by starting Python and trying to import Matplotlib:

```
import matplotlib.pyplot as plt  
  
print("Matplotlib installed successfully!")
```

1. Saving Your Notebook

Using the Toolbar

- **Save Icon:** Look for the save icon (a disk icon) located in the toolbar at the top of the notebook interface.
- **Click the Save Icon:** Clicking this icon will save your notebook.

Using Keyboard Shortcuts

- **Keyboard Shortcut:** You can quickly save your notebook by pressing **Ctrl + S** (Windows/Linux) or **Cmd + S** (macOS).

2. Saving as a Different File

If you want to save the notebook under a different name or format:

1. **Click on File:** In the menu at the top, click on **File**.
2. **Select Download as:** Hover over **Download as** in the dropdown menu.
3. **Choose a Format:** You can choose to download your notebook in various formats, such as:
 - **Notebook (.ipynb):** This is the default format for Jupyter notebooks.
 - **Python (.py):** Saves your code as a Python script.
 - **HTML (.html):** Saves the notebook as a static HTML file.
 - **Markdown (.md):** Saves as a markdown file.

3. Closing Your Notebook

- When you're finished working, make sure to save your notebook before closing it to avoid losing any unsaved changes.
- If you attempt to close the browser tab or navigate away without saving, Jupyter Notebook will typically prompt you to save your changes.

4. Auto-Save Feature

Jupyter Notebook has an auto-save feature that automatically saves your work at regular intervals. However, it's always a good practice to manually save before making significant changes or when you're done working.

5. Save and Check Output

- After saving, you should see a message at the top right corner indicating when the notebook was last saved (e.g., "Last saved a few seconds ago").