**Conda Environment Creation and Jupyter Installation**

This guide walks you through creating a conda environment and installing Jupyter Notebook for data science development.

**Prerequisites**

* Anaconda or Miniconda installed on your system

**Step-by-Step Instructions**

**1. Check Your Base Environment**

Open the Anaconda Prompt. You'll start in the base environment with a default Python version already installed. Check the current Python version:

python --version

**2. Test the Python Prompt (Optional)**

You can enter the Python interactive prompt by typing:

python

While you can execute basic commands here, the default prompt isn't scalable or versatile for development work. For web development, consider using an IDE like Atom or VS Code. For data science, Jupyter Notebook is the preferred choice.

Exit the Python prompt:

exit()

**3. Create a New Conda Environment**

Create a new conda environment with Python 3.13.5:

conda create --name PythonBootcampEnvironment python=3.13.5

This command will:

* Create an environment named "PythonBootcampEnvironment"
* Install Python version 3.13.5
* Download and install all required dependencies from the internet

**Note:** You can create multiple conda environments with different Python versions and names as needed.

**4. Activate the Environment**

Activate your newly created environment:

conda activate PythonBootcampEnvironment

You'll notice the environment name appears in your prompt, indicating it's active. You can:

* Deactivate at any time using conda deactivate
* Switch between environments by using conda activate [environment\_name]

**5. Install Jupyter Notebook**

With your environment activated, install Jupyter Notebook using pip (Python's package installer):

pip install notebook

This will install Jupyter Notebook and all its dependencies in your active environment.

**6. Navigate to DemoAppCodes Directory**

Open a new command prompt or terminal and navigate to your specific project directory:

cd E:\CompletePython3Bootcamp\Projects\GitDemo\DemoAppCodes

**7. Launch Jupyter Notebook from DemoAppCodes Directory**

With your conda environment still activated, launch Jupyter Notebook:

jupyter notebook

This will open Jupyter Notebook in your browser, starting directly from the DemoAppCodes directory.

**8. Create a New Jupyter Notebook**

1. Click the **"New"** button in the top right corner
2. Select **"Python 3"** from the dropdown menu
3. A new untitled notebook will open in a new tab and will be saved in the DemoAppCodes directory

**9. Rename Your Notebook**

1. Click on **"Untitled"** at the top of the notebook
2. Rename it to DemoAppNotebook
3. The file will automatically be saved as DemoAppNotebook.ipynb in the DemoAppCodes directory

**10. Add Content to Your Notebook**

In the first cell, add some basic Python code to test your setup:

print("Hello, World!")

print("Welcome to Python Bootcamp!")

print("Jupyter Notebook is working perfectly!")

Run the cell by pressing **Shift + Enter** or clicking the **"Run"** button.

**11. Save Your Notebook**

Save your work using **Ctrl + S** or by clicking **File > Save and Checkpoint**.

**12. Synchronize with GitHub**

Now we'll sync our project directory with GitHub. Navigate back to the main GitDemo directory in your command prompt:

cd E:\CompletePython3Bootcamp\Projects\GitDemo

Execute the following Git commands to sync with GitHub (assuming the repository is already initialized and connected):

1. **Stage all changes**:
2. git add .
3. **Commit your changes**:
4. git commit -m "Install IPython and Jupyter Notebook, create DemoAppNotebook"
5. **Push to remote repository**:
6. git push origin main

**13. Verify Your Setup**

In this way, you have successfully:

* Created a conda environment with Python 3.13.5
* Installed and configured Jupyter Notebook
* Created your first notebook with basic Python code
* Synchronized your project with GitHub for version control
* Established a workflow for managing your Python bootcamp projects

Your development environment is now ready for the complete Python bootcamp journey!

**Key Points to Remember**

* **pip** is Python's package installer and downloader
* Each conda environment is isolated, allowing you to maintain different Python versions and package sets
* Always ensure you're in the correct environment before installing packages
* Jupyter Notebook is ideal for data science, machine learning, and exploratory data analysis

**Useful Commands**

* conda list - View installed packages in current environment
* conda env list - List all available environments
* conda deactivate - Deactivate current environment
* jupyter notebook --help - View Jupyter Notebook options