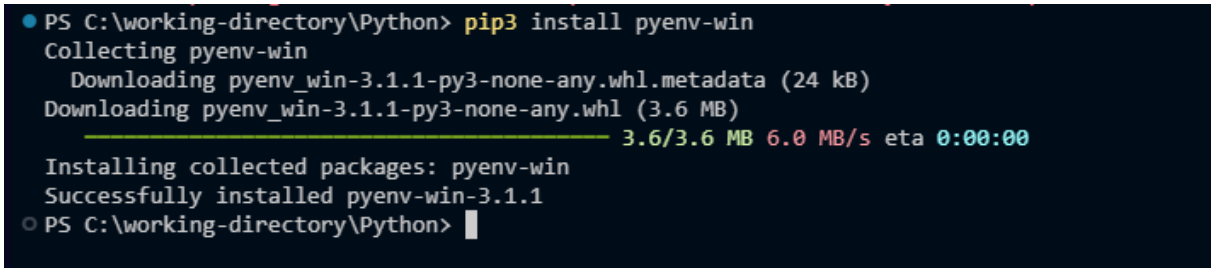
2024-07-11#Pyenv-python-enviromental

Python

---

pyenv

- ```
  ![[Pasted image 20240711140817.png]]
  ```




```
pyenv install 3.12.4
```

.

```
pyenv global 3.12.4
```

.

```
pyenv local 3.12.4
```

Conda

Miniconda3

- Since you're running the command in a Miniconda environment
- (indicated by the `(base)` prefix in your prompt),
- the Python 3.12.4 version is **local** to that environment.

In other words, this version of Python is installed and managed by Miniconda, and it's not the system's global Python installation. This means that:

- The Python 3.12.4 version is only available within this specific Miniconda environment.
- It doesn't affect the system's global Python installation (if you have one).
```

- You can have multiple Miniconda environments with different Python versions, and they won't interfere with each other or the system's global Python installation.

To confirm, you can try running

- `where python` (on Windows) or
- `which python` (on Linux/macOS)

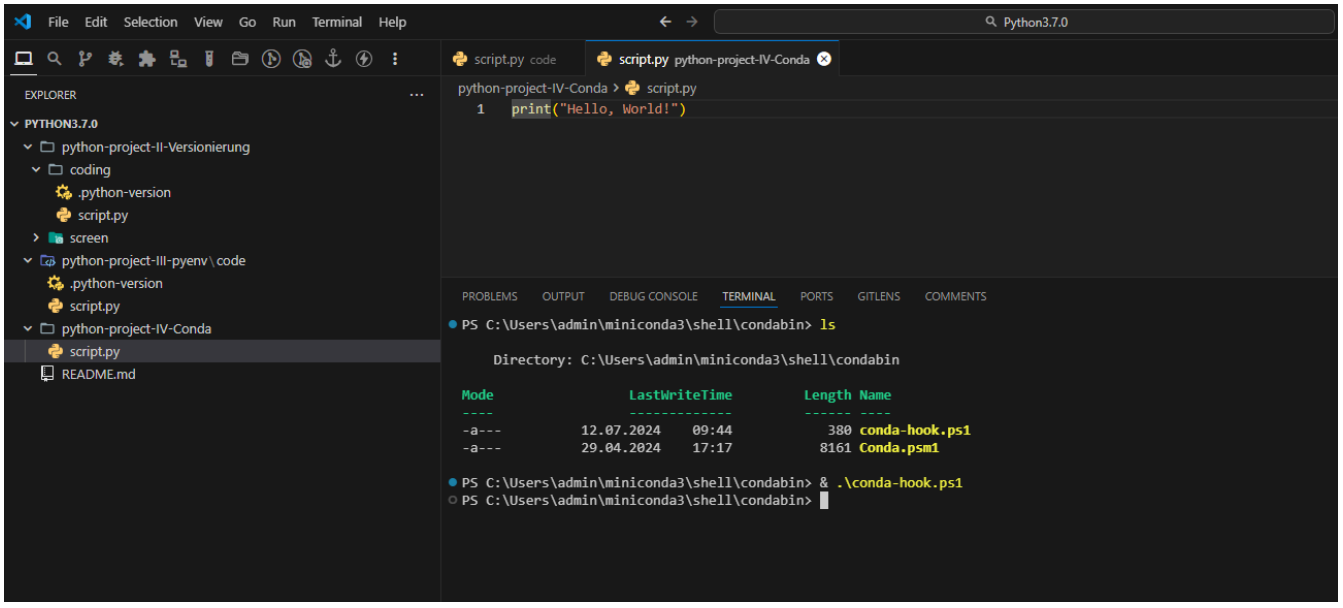
to see the path of the Python executable.
You should see a path related to your Miniconda environment,
indicating that it's a local installation.

activate Miniconda

- **Step 1: Activate the Miniconda environment**
- wir springen in den Pfad `C:\Users\admin\miniconda3\shell\condabin`
- führen folgenden Befehl aus

```
conda-hook.ps1
```

.



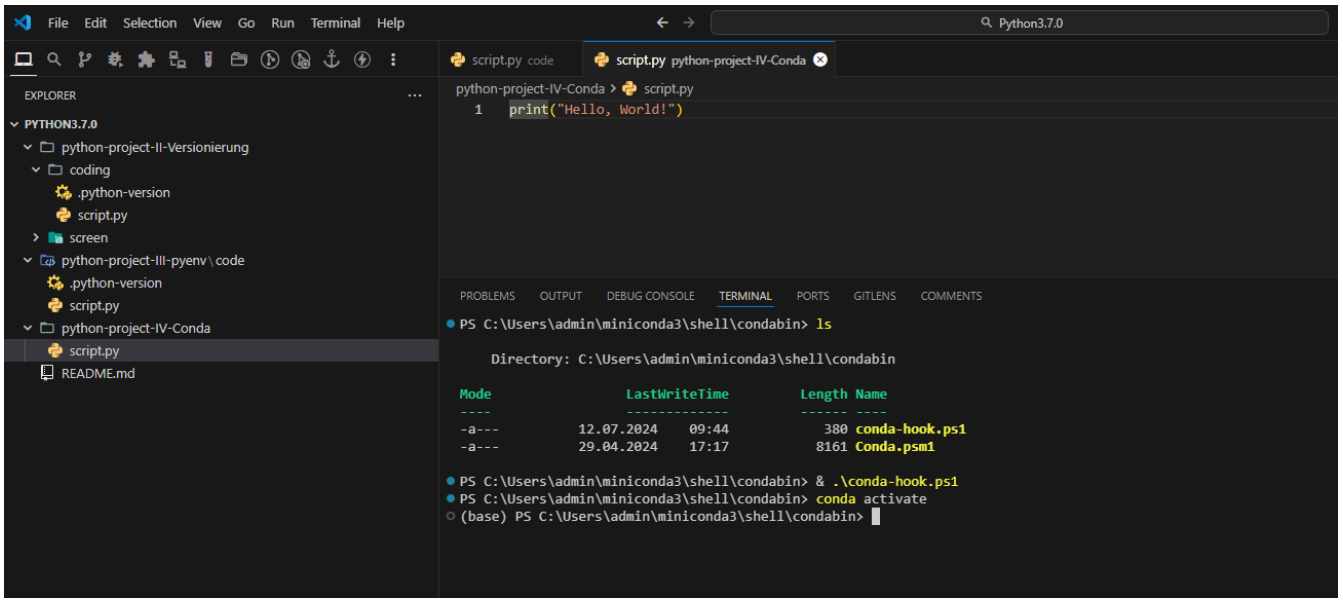
.

activate base environmental

- **Step 2a: Activate the base environment**
- wir führen ebenfalls aus:

```
ls
        conda activate
```

.



deactivate base environmental

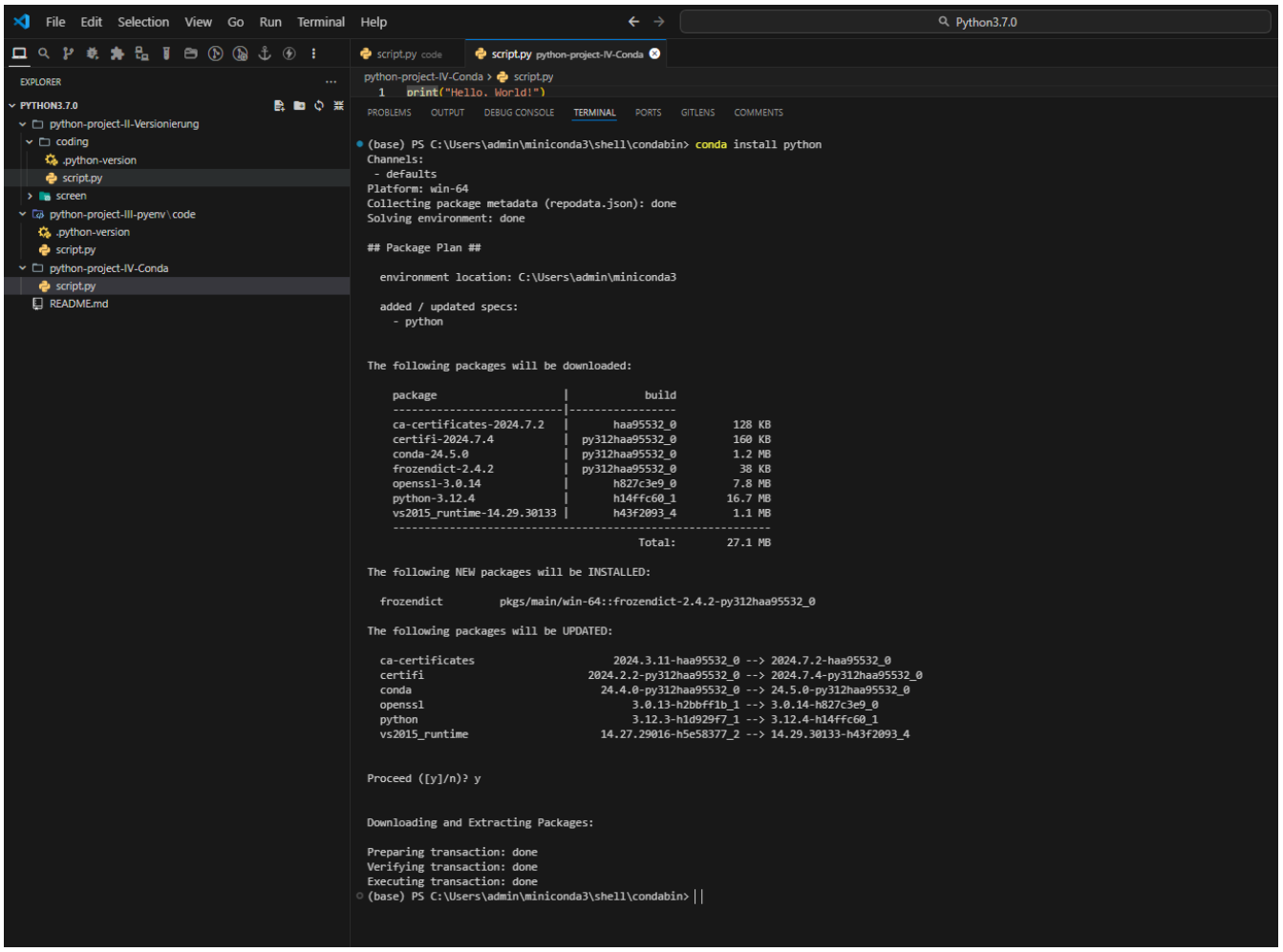
- **Step 2b: Deactivate the environment (if needed !)**

```
conda deactivate
```

.

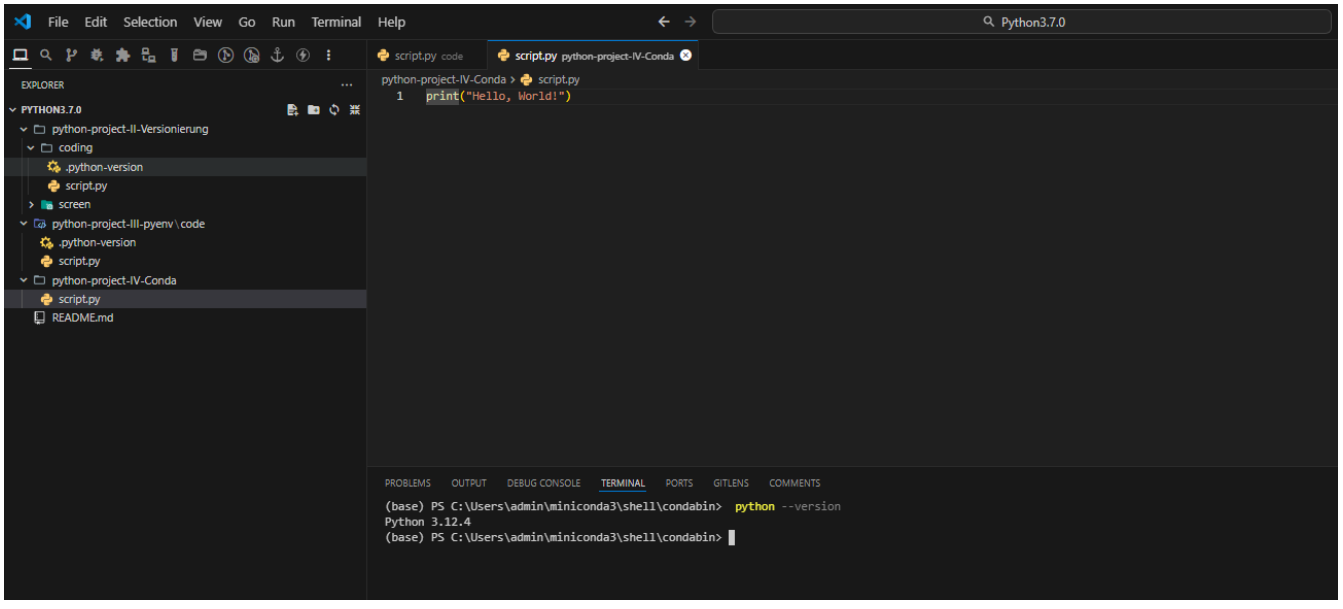
install package python

- **Step 3:** install packages



- You should now be able to use Python 3.12.4 in your Miniconda environment. You can verify this by running a Python command, such as:
- goTo: `(base) PS C:\Users\admin\miniconda3\shell\condabin>`

```
python --version
```



create Script

- To create a Python script in a Miniconda environment, you can follow these steps:

Step 1: Activate the Miniconda environment

- Open a new terminal or command prompt and activate the Miniconda environment where you want to create the script. You can do this by running:

```
conda activate base
```

Step 2: Create a new file with a `.py` extension

Use a text editor or an Integrated Development Environment (IDE) like

- PyCharm,
- Visual Studio Code, or
- Spyder

to create a new file with a `.py` extension. For example, you can create a file called `my_script.py`.

Step 3: Write your Python code

Write your Python code in the `my_script.py` file. For example:

```
# my_script.py
print("Hello, World!")
```

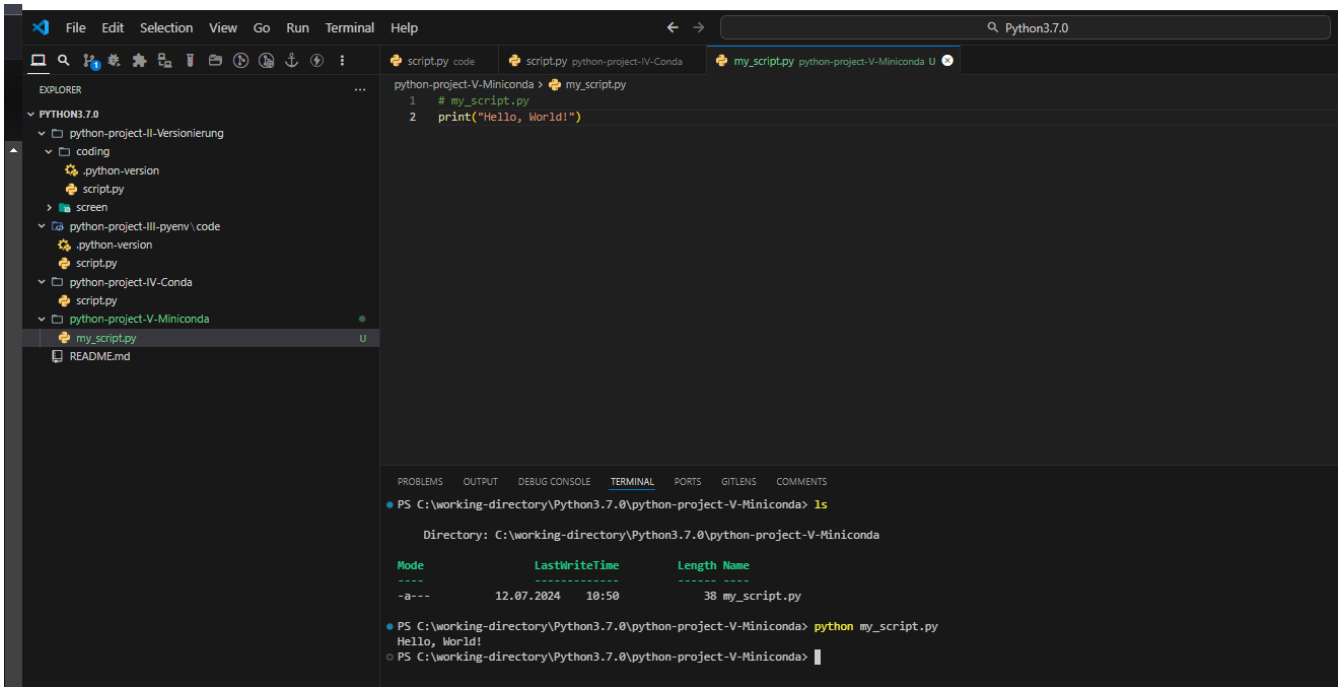
Step 4: Save the file

Save the file in a location of your choice, such as your desktop or a project folder.
`C:\working-directory\Python3.7.0\python-project-V-Miniconda`

Step 5: Run the script

To run the script, navigate to the directory where you saved the file using the terminal or command prompt. Then, run the script using Python:

```
# C:\working-directory\Python3.7.0\python-project-V-Miniconda\
python my_script.py
```



create ".python-version"

To create a `.python-version` file with Miniconda, you can use the `conda` command to specify the Python version you want to use.

Here's an example:

```
conda activate
```

.

```
conda env config vars set PYTHON_VERSION=3.7
```

.

```
conda env config --prune
```

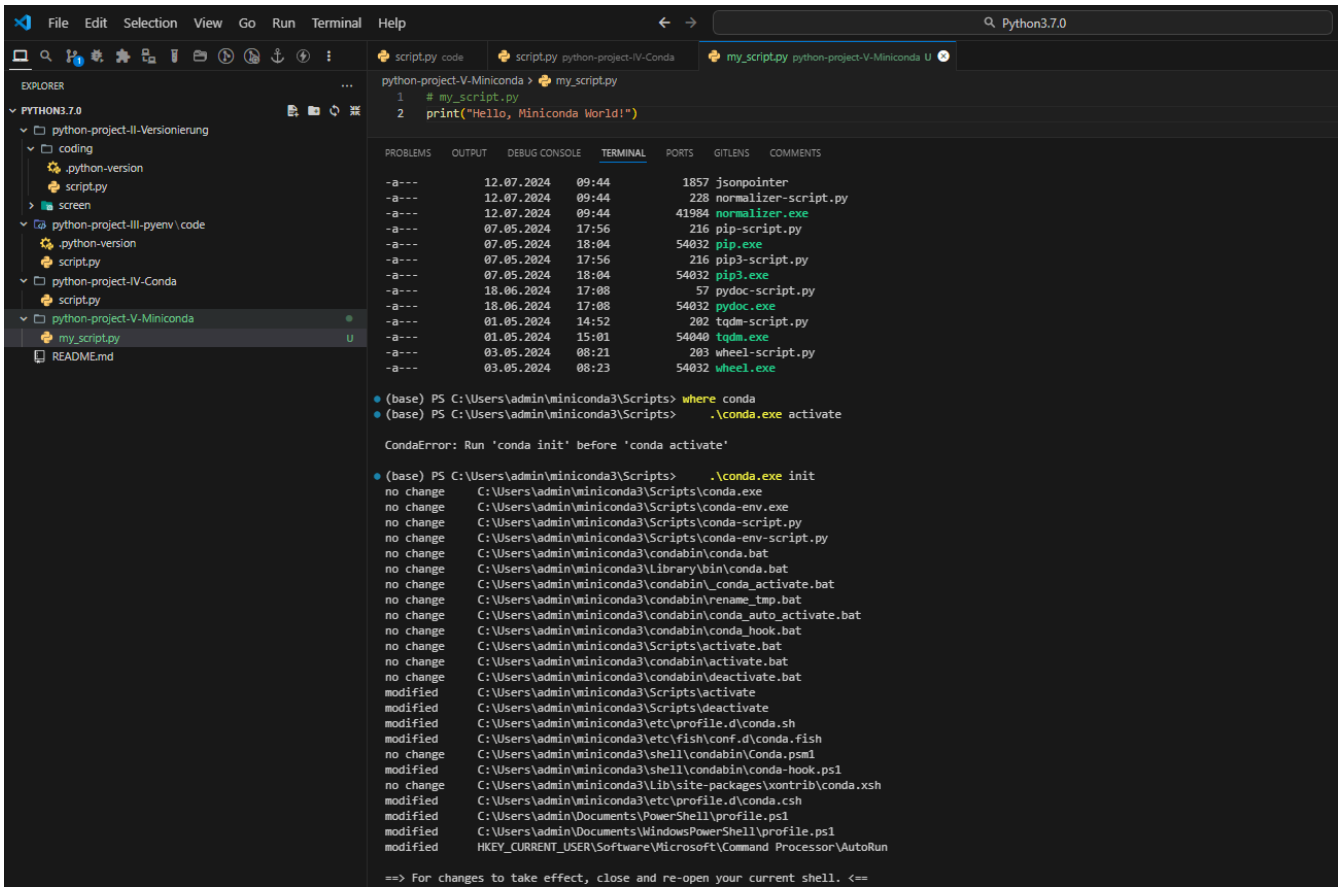
activate Conda

wenn conda nicht aktiviert - dann gehen wir in den Pfad:

- `C:\Users\admin\miniconda3\Scripts`

und führen den Befehl aus:

```
.\conda.exe init
```



.

```
        .\conda.exe activate
```

.

conda init

- To fix this, try the following:

1. **Close and reopen your PowerShell session**. This will ensure that the changes made by `conda init` take effect.

```
(base) PS C:\working-directory\Python3.7.0\python-project-V-Miniconda> conda init
no change     C:\Users\admin\miniconda3\Scripts\conda.exe
no change     C:\Users\admin\miniconda3\Scripts\conda-env.exe
no change     C:\Users\admin\miniconda3\Scripts\conda-script.py
no change     C:\Users\admin\miniconda3\Scripts\conda-env-script.py
no change     C:\Users\admin\miniconda3\condabin\conda.bat
no change     C:\Users\admin\miniconda3\Library\bin\conda.bat
no change     C:\Users\admin\miniconda3\condabin\_conda_activate.bat
no change     C:\Users\admin\miniconda3\condabin\rename_tmp.bat
no change     C:\Users\admin\miniconda3\condabin\conda_auto_activate.bat
no change     C:\Users\admin\miniconda3\condabin\conda_hook.bat
no change     C:\Users\admin\miniconda3\Scripts\activate.bat
no change     C:\Users\admin\miniconda3\condabin\activate.bat
no change     C:\Users\admin\miniconda3\condabin\deactivate.bat
no change     C:\Users\admin\miniconda3\Scripts\activate
no change     C:\Users\admin\miniconda3\Scripts\deactivate
no change     C:\Users\admin\miniconda3\etc\profile.d\conda.sh
```

```
no change     C:\Users\admin\miniconda3\etc\fish\conf.d\conda.fish
no change     C:\Users\admin\miniconda3\shell\condabin\Conda.psm1
no change     C:\Users\admin\miniconda3\shell\condabin\conda-hook.ps1
no change     C:\Users\admin\miniconda3\Lib\site-packages\xontrib\conda.xsh
no change     C:\Users\admin\miniconda3\etc\profile.d\conda.csh
no change     C:\Users\admin\Documents\WindowsPowerShell\profile.ps1
no change     C:\Users\admin\Documents\PowerShell\profile.ps1
no change     HKEY_CURRENT_USER\Software\Microsoft\Command Processor\AutoRun
No action taken.
```

.

2. **Run `conda info --envs` to list your environments**. This will verify that your environments are set up correctly.

```
(base) PS C:\working-directory\Python3.7.0\python-project-V-Miniconda>
         conda info --envs
# conda environments:
#
base                  *  C:\Users\admin\miniconda3
```

3. **Run `conda activate` without any arguments**. This should activate the base environment.

```
conda activate
```

4. RUN `conda env config`

```
conda env config vars set PYTHON_VERSION=3.7
```

.

- It looks like you've successfully set the `PYTHON_VERSION` variable to `3.7` for your Conda environment.

5. Reactivate your Environment for the Changes to take Effect.

   `ps conda deactivate`

   .

```
conda activate
```

6. After reactivating your environment,
   - you can verify that the `PYTHON_VERSION` variable has been set correctly by running:

```
conda env config vars list
```

create environment **myenv**

- If you need to create a new environment with a specific Python version,

8. You can use the `conda create` command with the `--python` option, like this:

```
conda create --name myenv python=3.7
```

.

- Congratulations!
  - You've successfully created a new Conda environment
  - named `myenv` with Python 3.7 as the default Python version.
  - The output shows that Conda has downloaded
  - and installed the necessary packages,
  - including Python 3.7, pip, and other dependencies.
  - The environment is now ready to use.

```
conda activate myenv
```

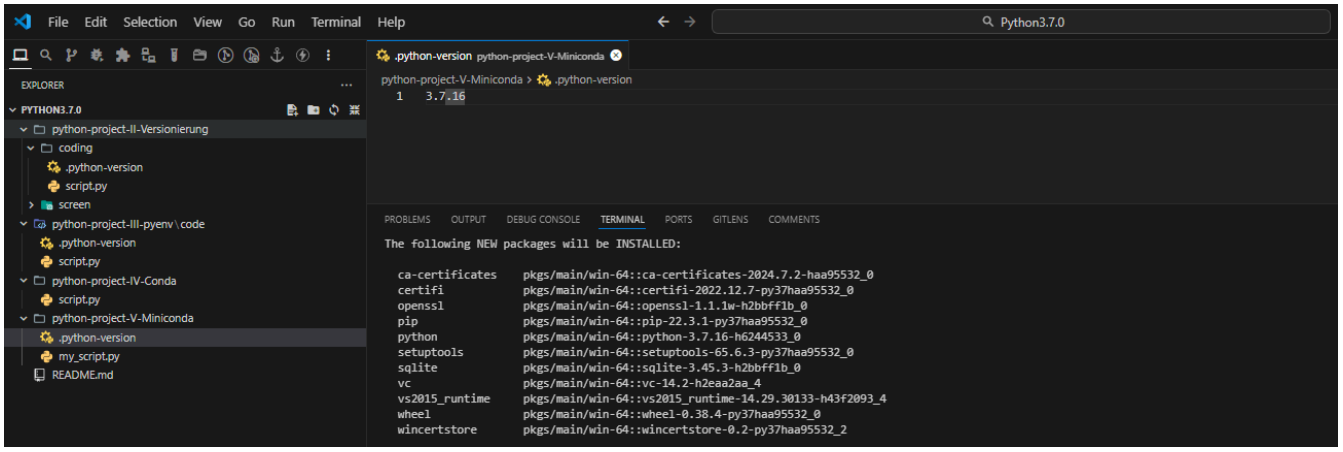
- This will switch you to the `myenv` environment,
- and you can verify this by checking the command prompt,
- which should now display `(myenv)` instead of `(base)`.

9. Run the following command to create the `.python-version` file:

```
python -m python_version > .python-version
```

.

10. open file `.python-version` and write:
    - `3.7.16`
    - This indicates that the project uses Python 3.7.16.



Note: The `python_version` module is not a built-in Python module, but it's a common convention to use this command to create a `.python-version` file.

11.

```
conda activate python-project-V-Miniconda
```