pythonenv

August 27, 2024

1 Python Environements

Below, I'll outline step-by-step instructions for initiating a Python environment both with Conda and without (using venv), which are two common methods for Python environment management.

1.1 Conda

First, you'll need to install Conda. You can do this by following the instructions on the Conda website.

1.1.1 Creating a Conda Environment

Once you have Conda installed, you can create a new environment using the following command on the command line in your terminal or Anaconda Prompt:

```
conda create --name myenv python=3.9
```

This will create a new environment named `myenv` with Python 3.9 installed.

1.1.2 Activating the Environment

To activate the environment, use the following command:

```
conda activate myenv
```

1.1.3 Installing Packages

To install packages into the environment, use the following command:

```
conda install package_name1 package_name2
```

1.1.4 Deactivating the Environment

To deactivate the environment, use the following command:

```
conda deactivate
```

1.1.5 Listing Environments

To list all the environments, use the following command:

```
conda info --envs
```

1.1.6 Documentation

For each project and document their environment's dependencies

```
conda env export > environment.yml
```

1.1.7 Removing Env

To remove an environment, use the following command:

```
conda remove --name myenv --all
```

1.2 venv

This is also know as Python Virtual Environments. The venv module is a built-in module in Python that allows you to create isolated Python environments. It is a simple way to create a virtual environment for your Python projects.

1.2.1 Creating a venv Environment

To create a new venv environment, use the following command on the command line in your terminal or Anaconda Prompt:

```
python -m venv myenv
```

This will create a new environment named `myenv` with Python 3.9 installed. ### Activating the Environment To activate the environment, use the following command:

```
source myenv/bin/activate
```

1.2.2 Installing Packages

To install packages into the environment, use the following command:

```
pip install package_name1 package_name2
```

1.2.3 Deactivating the Environment

To deactivate the environment, use the following command:

```
deactivate
```

1.2.4 Listing Environments

To list all the environments, use the following command:

```
find / -name activate 2>/dev/null | grep "/bin/activate$"
```

1.2.5 Documentation

For each project and document their environment's dependencies

```
pip freeze > requirements.txt
```