UV Tutorial

For those using Linux or Mac, you can install uv with this command:
curl -LsSf https://astral.sh/uv/install.sh | sh

Alternatively, you can install uv using pip:
pip install uv

To keep uv up to date, simply run:
uv self update

One of the coolest features of uv is its ability to manage different Python versions effortlessly. To see the list of available Python versions, use:
uv python list

Traditional Python Project Setup vs UV Approach
Let's compare the traditional way of setting up a Python project with the uv approach.
Here's how you might traditionally set up a new Python project using pip and virtualenv:

- Create a new directory
- Create a virtual environment: virtualenv venv
- Activate the virtual environment:
- Linux: source .venv/bin/activate
- Windows: .\venv\Scripts\activate

4. Install packages: pip install requests
5. Generate requirements: pip freeze > requirements.txt

Project Migration (Traditional)
When you want to move your project to another machine, you typically:

- Copy the project directory
- Create a new virtual environment
- Install dependencies: pip install -r requirements.txt

Modern Approach with UV
Now, let's see how uv can simplify this process and make it much faster:

- Create a new directory
- Create a virtual environment: uv venv or uv venv --python 3.10 for a specific Python version
- Activate the virtual environment:
- Linux: source .venv/bin/activate
- Windows: .\venv\Scripts\activate

4. Install packages: uv pip install requests
5. Generate requirements: uv pip freeze > requirements.txt

Project Migration with UV

Moving your project to another machine with uv is a breeze:

- Copy the project directory
- Create a virtual environment: uv venv or uv venv --python 3.10
- Install dependencies: uv pip sync requirements.txt

Converting Existing virtualenv to UV

Already using a traditional virtualenv? No problem! Converting it to uv is super simple:

- Navigate to your project directory with the existing virtual environment.
- Run: uv init to adopt the existing environment.
- Convert dependencies: uv add -r requirements.txt to create pyproject.toml and uv.lock files.