This documentation provides a streamlined guide for installing and managing Python projects using **uv**, an extremely fast Python package and project manager written in Rust.

Getting Started with `uv`

`uv` is designed to replace `pip`, `pip-tools`, and `virtualenv` while being 10–100x faster.

1. Installation

Install `uv` on your system using the following platform-specific commands. For more details, visit the official installation documentation.

Windows

Run the following in PowerShell:

```
powershell -ExecutionPolicy ByPass -c "irm https://astral.sh/uv/install.ps1 | iex"
```

macOS / Linux

You can install via `curl` or `wget`:

```
# Using curl
curl -LsSf https://astral.sh/uv/install.sh | sh

# Using wget
wget -qO- https://astral.sh/uv/install.sh | sh
```

---

2. Python Version Management

`uv` can manage Python installations directly, eliminating the need for external tools like `pyenv`.

- **Install a specific version:**

```
uv python install 3.12
```

- **Pin a version to the current project:**

```
uv python pin 3.12
```

---

3. Virtual Environment Setup

Initialize a new virtual environment in your current directory.

```
uv venv
```

> *[!TIP] By default, this creates a `.venv` folder. `uv` will automatically detect this environment for subsequent commands.*

---

4. Dependency Management

`uv` provides a familiar interface for those coming from `pip`, but with significantly improved performance.

Working with Requirements Files

- **Install from file:**

```
uv pip install -r requirements.txt
```

- **Generate/Freeze requirements:**

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

Jupyter Notebook Integration

To use `uv` managed environments within Jupyter, you must install and register the kernel:

1. **Install tools:**

```
uv pip install jupyter ipykernel
```

2. **Register the kernel:**

```
uv run python -m ipykernel install --user --name docscanner_env --display-name "Doc Scanner (uv)"
```

---

5. Execution and Deployment

Use `uv run` to execute scripts or applications. This ensures that the command runs within the context of your environment and project settings.

- **Run a Python script:**

```
uv run python hello_world.py
```

- **Launch a Streamlit application:**

```
uv run streamlit run app.py
```

Quick Summary Table

Task	Command

Install Python	`uv python install <version>`
Create Env	`uv venv`
Install Package	`uv pip install <package>`
Run App	`uv run <command>`