

Install Streamlit using command line

This page will walk you through creating an environment with `venv` and installing Streamlit with `pip`. These are our recommended tools, but if you are familiar with others you can use your favorite ones too. At the end, you'll build a simple "Hello world" app and run it. If you prefer to have a graphical interface to manage your Python environments, check out how to [Install Streamlit using Anaconda Distribution](#).

Prerequisites

As with any programming tool, in order to install Streamlit you first need to make sure your computer is properly set up. More specifically, you'll need:

1. Python

We support [version 3.8 to 3.12](#).

2. A Python environment manager (recommended)

Environment managers create virtual environments to isolate Python package installations between projects.

We recommend using virtual environments because installing or upgrading a Python package may cause unintentional effects on another package. For a detailed introduction to Python environments, check out [Python Virtual Environments: A Primer](#).

For this guide, we'll be using `venv`, which comes with Python.

3. A Python package manager

Package managers handle installing each of your Python packages, including Streamlit.

For this guide, we'll be using `pip`, which comes with Python.

4. Only on MacOS: Xcode command line tools

Download Xcode command line tools using [these instructions](#) in order to let the package manager install some of Streamlit's dependencies.

5. A code editor

Our favorite editor is [VS Code](#), which is also what we use in all our tutorials.

Create an environment using venv

1. Open a terminal and navigate to your project folder.

```
cd myproject
```

2. In your terminal, type:

```
python -m venv .venv
```

3. A folder named ".venv" will appear in your project. This directory is where your virtual environment and its dependencies are installed.

Activate your environment

4. In your terminal, activate your environment with one of the following commands, depending on your operating system.

```
# Windows command prompt
.venv\Scripts\activate.bat

# Windows PowerShell
.venv\Scripts\Activate.ps1

# macOS and Linux
source .venv/bin/activate
```

5. Once activated, you will see your environment name in parentheses before your prompt. "(.venv)"

Install Streamlit in your environment

6. In the terminal with your environment activated, type:

```
pip install streamlit
```

7. Test that the installation worked by launching the Streamlit Hello example app:

```
streamlit hello
```

If this doesn't work, use the long-form command:

```
python -m streamlit hello
```

8. Streamlit's Hello app should appear in a new tab in your web browser!

Built with Streamlit 

Fullscreen 

9. Close your terminal when you are done.

Create a "Hello World" app and run it

10. Create a file named `app.py` in your project folder.

```
import streamlit as st

st.write("Hello world")
```

11. Any time you want to use your new environment, you first need to go to your project folder (where the `.venv` directory lives) and run the command to activate it:

```
# Windows command prompt
.venv\Scripts\activate.bat

# Windows PowerShell
.venv\Scripts\Activate.ps1
```

```
# macOS and Linux
```

```
source .venv/bin/activate
```

12. Once activated, you will see your environment's name in parentheses at the beginning of your terminal prompt. "(.venv)"

13. Run your Streamlit app.

```
streamlit run app.py
```

If this doesn't work, use the long-form command:

```
python -m streamlit run app.py
```

14. To stop the Streamlit server, press `Ctrl+C` in the terminal.

15. When you're done using this environment, return to your normal shell by typing:

```
deactivate
```

What's next?

Read about our [Basic concepts](#) to understand Streamlit's dataflow model.

← **Previous:** Installation

Next: Use Anaconda Distribution →



Still have questions?

Our [forums](#) are full of helpful information and Streamlit experts.