Setting Up Julia, JupyterLab, and Pluto on Windows

Department of Physics, University of Southern Maine

September 2026

These instructions walk you through installing Julia on a Windows computer and setting it up so that you can open JupyterLab or Pluto notebooks with a single command in the terminal. Wherever you see a `PS>` prompt, type the command into PowerShell; wherever you see `julia>` or `pkg>`, type it into the Julia REPL.

1 Install Julia

Open PowerShell (search for "PowerShell" or "Windows Terminal" in the Start menu) and run:

```
PS> winget install julia -s msstore
```

This installs `juliaup`, the official Julia version manager, along with the current release of Julia. Close and reopen PowerShell, then confirm the installation:

```
PS> julia --version
julia version 1.13.0
```

If you already have Julia installed through `juliaup` and want to update, run `juliaup update`.

2 Add the packages

Start Julia by typing `julia` in PowerShell. At the `julia>` prompt, press `]` to enter package mode and add the two packages you need:

```
pkg> add IJulia Pluto
```

`IJulia` lets Jupyter run Julia code; `Pluto` is a self-contained notebook system that needs nothing else. Press Backspace to return to the `julia>` prompt, then load `IJulia` once so that it registers the "Julia 1.13" kernel with Jupyter:

```
julia> using IJulia
```

Packages you install this way go into the default environment, located at `C:\Users\`*yourname*`\.julia\environments\v1.`. Any other packages you need for coursework (for example `CairoMakie`, `DifferentialEquations`, or `LsqFit`) can be added the same way.

3 Install JupyterLab

The easiest option is to let `IJulia` install and manage its own copy of JupyterLab, so there is no separate Python installation to keep in sync. Still in the Julia REPL, run:

```
julia> using IJulia
julia> jupyterlab()
```

The first time you run this, Julia will ask whether it may install JupyterLab through Conda; answer `y`. The download takes a few minutes. When it finishes, JupyterLab opens in your web browser. To stop it, close the browser tab and press Ctrl-C in the terminal, then type `exit()` to leave Julia.

4 One-word launchers for the terminal

To avoid typing the commands above every time, define two shortcuts in your PowerShell profile. The profile is a script that runs each time PowerShell starts.

First, allow PowerShell to run local scripts (this only needs to be done once):

```
PS> Set-ExecutionPolicy -Scope CurrentUser RemoteSigned
```

Then open the profile in Notepad:

```
PS> notepad $PROFILE
```

If Notepad asks whether to create the file, click Yes. Paste in the following two functions, save, and close Notepad:

```
function jlab  { julia -e "using IJulia; jupyterlab(dir=pwd(), detached=true)" }
function pluto { julia -e "using Pluto; Pluto.run()" }
```

Close and reopen PowerShell. From now on:

- `jlab` opens JupyterLab in your browser, rooted in whatever folder you are currently in, and gives you your prompt back right away. (Remove `detached=true` if you would rather keep it attached so that Ctrl-C shuts it down.)
- `pluto` starts Pluto and opens it in your browser. Press Ctrl-C in the terminal to stop it.

A typical workflow is to `cd` into the folder for a homework assignment and then type `jlab` or `pluto`.

5 If you already have Jupyter installed

If you have JupyterLab installed through Anaconda or a Python virtual environment, you can keep using it: just run `jupyter lab` as usual. `IJulia` writes its kernel definition to the standard location, `%APPDATA%\jupyter\kernels\julia-1.13`, so your existing Jupyter will list "Julia 1.13" in its kernel menu. In that case you can skip Section 3 and change the `jlab` function to simply `function jlab { jupyter lab }`.

6 Troubleshooting

- **`julia` is not recognized.** Close and reopen PowerShell after installing; the installer adds Julia to your PATH but open windows do not see the change.
- **The Julia kernel does not appear in Jupyter.** In the Julia REPL, run `using IJulia; IJulia.installkernel("J` to re-register it.
- **`jlab` or `pluto` is not recognized.** Check that the profile saved correctly by running `notepad $PROFILE` again, and make sure you set the execution policy as described above.
- **Antivirus or firewall prompts.** JupyterLab and Pluto run a small local web server on your own computer; it is safe to allow access on private networks.