

Installing Julia

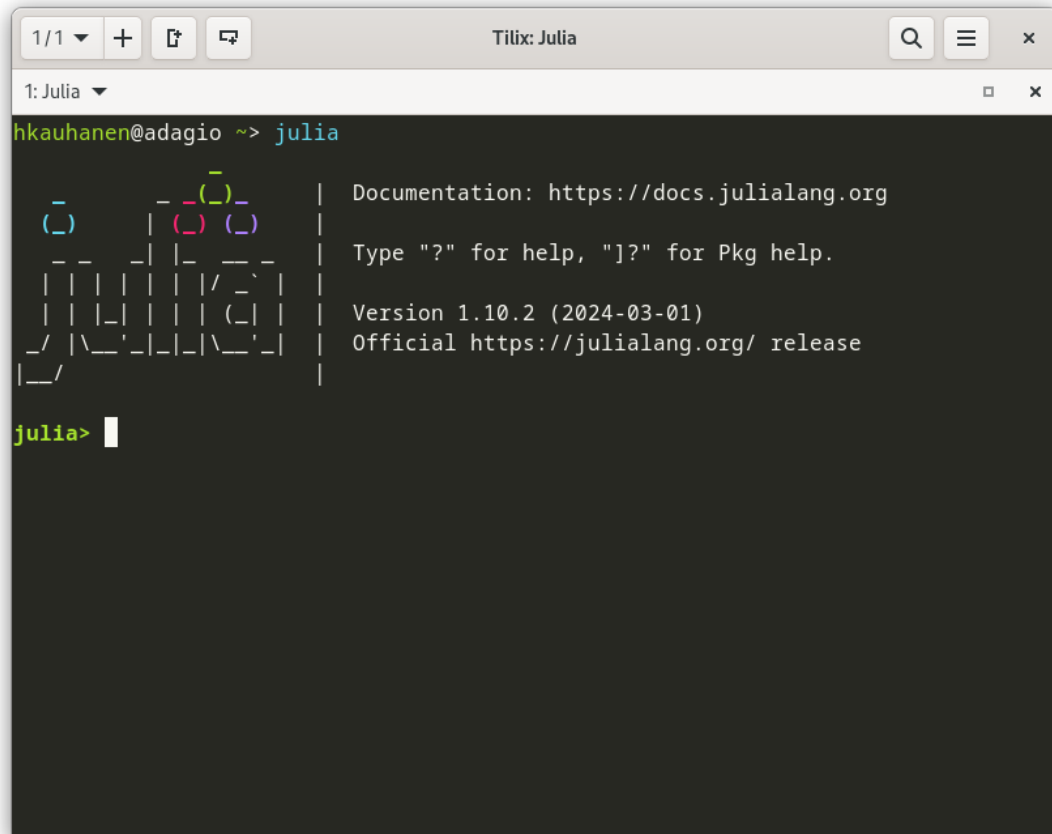
Homework | Agent-based modelling, Konstanz, 2024

Henri Kauhanen

9 April 2024

In this course, we will conduct all our programming work in the Julia programming language. This homework will guide you through the process of installing Julia on your computer. You will also be installing a few important packages (think of them as “add-ons” to the base Julia installation) that we will need later on. Finally, we will install an editor which will make dealing with source code a bit more convenient compared to just running Julia in a terminal window.

1. Navigate to <https://julialang.org>, download the proper installation files for your operating system, and install Julia. **Make sure to download version 1.10.2 or newer.** (Older versions may not be fully compatible with the code we will be working with in this course.)
2. With installation complete, launch Julia. You should be greeted by a screen such as this:



```
1/1 + [ ] [ ] [ ]
Tilix: Julia
1: Julia
hkauhanen@adagio ~> julia

      _       _       _
     ( )     ( )     ( )
    _ _ _ _ _ _ _ _ _
   | | | | | | | | | |
   | | | | | | | | | |
  _/ | \ _ ' | _ | \ _ ' |
 |_/  |_/

Documentation: https://docs.julialang.org
Type "?" for help, "]"? for Pkg help.
Version 1.10.2 (2024-03-01)
Official https://julialang.org/ release

julia> 
```

This is known as the Julia **REPL**, which stands for read-eval(uate)-print loop. Here, you can give Julia commands, it will execute them, and you get the output back. For example, try the following commands:

```
2+2
sqrt(2)
[1, 2, 3, 4]
sum([1, 2, 3, 4])
```

3. The REPL can also be used to install packages i.e. extensions. We will now install two packages, *Agents* and *CairoMakie*, along with all their dependencies (i.e. further packages required to run those two packages). To do this, execute the following commands in the REPL:

```
using Pkg
Pkg.add("Agents")
Pkg.add("CairoMakie")
```

This will take a long while, as the two packages have many dependencies. Wait patiently.

When everything is complete, you may exit the REPL:

```
exit()
```

4. It would be very inconvenient to do all our coding in the REPL. It is much better to use a text editor or an IDE (integrated development environment). Here, you are free to make your own choice, if you already have a favourite editor for coding. For the purposes of this course, however, I recommend Visual Studio Code – this is the editor I will be using in the lectures, and so if you choose to use the same editor, things will be a bit smoother. To get started with VS Code, navigate to <https://code.visualstudio.com/download>, download the version for your operating system, and follow the instructions to install it.
5. Once VS Code has been successfully installed, we need to install a Julia extension for it. This will allow the editor to interface with your Julia installation, so that you will be able to execute code directly from the editor. Follow the instructions [here](#), in bullet point number 3, to do this.

If all went well, you now have an up-to-date, working Julia installation, with a number of packages installed, plus a source code editor that interfaces seamlessly with that installation.

If something went wrong, try again. **Google your problem** to see if other users have already run into the same problem and have a solution. If it still refuses to work, bring your problem to the next lecture – we will discuss how to solve it and get you a working system.

Bonus

If you are already familiar with another programming language such as Python, I recommend you to watch this short video from IBM Technology detailing some of the major differences between Python and Julia. It does go into some technical details which may seem obscure for now (“dynamic typing”, “multiple dispatch”), but worry not – I will explain those concepts next week.

<https://www.youtube.com/watch?v=NF5InxVP3ZQ>