

Getting started with Julia - PSYCH 670

Peter DiBerardino

Jatheesh Srikantharajah

January 24, 2022

1 Practical

Installation is pretty simple. Go to <https://julialang.org/downloads/> and follow the instructions for your operating system.

Like other languages, a Julia console can be opened in your computer's terminal (use command `Julia`). Or you can write script in your favourite text editor and run the entire file from your terminal. For example, if your document is named `test.jl`, you can run it in your terminal with the command `julia test.jl`. You can even pass parameters to this document via `julia test.jl arg1 arg2...`. These parameters will be set in the global environment, and can be accessed through `ARGS`.

Junolab <https://junolab.org/> is a more complex IDE designed for Julia.

Here is a simple “Hello World” program. It will also print any arguments passed the file called from the terminal.

```
print("Hello World")
print(ARGS)
```

2 Theoretical

Julia superficially looks similar to Python, but differs in some important syntax (e.g. string concatenation) and theoretical design. For example, Julia was written with the goal of making the gap between math and code smaller. Consider how simple it is to write a function that multiplies each element in a list by 2:

```
f(x) = [p = p * 2 for p in x]
f([1,2,3])
```

This example is from <https://towardsdatascience.com/julias-most-awesome-features-be51f798f140>.

This document was written with \LaTeX . See the accompanying source-code `julia_intro.tex` and the sample Julia script document `julia_intro.js`.