

HIGH PERFORMANCE RESEARCH COMPUTING

ACES: Introduction to Julia

2025 ACES Summer Workshop

20 July 2025



High Performance
Research Computing
DIVISION OF RESEARCH



High Performance Research Computing | hprc.tamu.edu | NSF Award #2112356

Introduction to Julia: Outline



Part I: A brief overview of Julia



Julia is a high-level general-purpose dynamic programming language primarily designed for **high-performance numerical analysis and computational science**.

- Born in MIT's Computer Science and Artificial Intelligence Lab in 2009
- Combined the best features of Ruby, MatLab, C, Python, R, and others
- First release in 2012
- Latest stable release v1.11.4 as of Mar 10, 2025
- <https://julialang.org/>
- customized for "greedy, unreasonable, demanding programmers"
- [Julia Computing](#) established in 2015 to provide commercial support



Major features of **Julia**:

- **Fast**: designed for high performance
- **General**: supports different programming patterns
- **Dynamic**: dynamically-typed with good support for interactive use
- **Technical**: efficient numerical computing with a mathematics-friendly syntax
- **Optionally typed**: a rich language of descriptive data types
- **Composable**: Julia's packages naturally work well together

*"Julia is as programmable as Python while it is as fast as Fortran for number crunching. It is like **Python on steroids**."*

--an anonymous Julia user on the first impression of Julia.

Where to Run Julia

- Juno IDE - developed for the Julia language (no longer under development)
- VSCode - extensions for Julia are actively being managed
- Jupyter Notebook
- Julia REPL
 - Run, Evaluate, Print, Loop
 - Interactive
 - Searchable history, tab-completion, keybindings, dedicated help and shell modes
- More information: <https://hprc.tamu.edu/kb/Software/Julia/>



Accessing the HPRC ACES Portal

The screenshot shows the HPRC website interface. At the top, the Texas A&M logo is on the left, followed by the text "TEXAS A&M HIGH PERFORMANCE RESEARCH COMPUTING". To the right are social media icons for Twitter, YouTube, and Facebook. Below this is a dark red navigation bar with links: Home, User Services, Resources, Research, Policies, Events, Training, About, and Portal. The "Portal" link is highlighted with a red box. A dropdown menu is open from the "Portal" link, listing: Terra Portal, Grace Portal, FASTER Portal, FASTER Portal (ACCESS), ACES Portal (ACCESS) (highlighted with a red box), and Launch Portal (ACCESS). On the left side of the page, there is a "Quick Links" section with links like "New User Information", "Accounts", "Apply for Accounts", "Manage Accounts", "User Consulting", "Training", "Knowledge Base", "Software", and "FAQ". Below that is a "User Guides" section with links for "ACES", "FASTER", "Grace", and "Terra". The main content area features a large image of server racks and a diagram illustrating a biological process involving "Plasmid DNA" and a "Cell" with a "Nucleus".

HPRC webpage: hprc.tamu.edu

[>_aces Shell Access](#)

OnDemand provides an integrated, single access point for all of your HPC resources.

Message of the Day

ACES Maintenance Status, October 10

The planned maintenance for the PCIe Gen5 composability fabrics has been completed. The PVCs in two Gen5 fabrics will remain unavailable until replacement components arrive tomorrow or next week.

IMPORTANT POLICY INFORMATION

- **Unauthorized use of HPRC resources is prohibited and subject to criminal prosecution.**
- **Use of HPRC resources in violation of United States export control laws and regulations is prohibited.**
- **Sharing HPRC account and password information is in violation of State Law. Any shared accounts will be DISABLED.**
- **Authorized users must also adhere to ALL policies at: <https://hprc.tamu.edu/policies>**

!! WARNING: THERE ARE ONLY NIGHTLY BACKUPS OF USER HOME DIRECTORIES. !!

Run the following commands to install the necessary packages:

```
$ mkdir $SCRATCH/.julia
$ ln -s $SCRATCH/.julia
$ module load Julia/1.11.4-linux-x86_64
$ julia
julia> ]
(v1.11) pkg> add IJulia
(v1.11) pkg> add PrettyTables
(v1.11) pkg> add Plots
(v1.11) pkg> add CSV
(v1.11) pkg> add XLSX
(v1.11) pkg> add DataFrames
(v1.11) pkg> add StatsPlots
```

Press backspace to exit the pkg manager and type `exit()` to quit Julia



OnDemand provides an

Message of the D

ACES Maintenance

The planned maintenance for components arrive tomorrow

IMPORTANT POLIC

- Unauthorized use of H
- Use of HPRC resource
- Sharing HPRC account
- Authorized users must also adhere to ALL policies at: <https://hprc.tamu.edu/policies>

GUI

VNC

NextSilicon VNC

Imaging

CryoSPARC

ImageJ

Jmol

Paraview

cisTEM

Servers

Jupyter Notebook

JupyterLab

RStudio (Short Course)

RStudio

TensorBoard

Tutorials OnDemand

point for all of your HPC resources.

brics has been completed. The PVCs in two Gen5 fabrics will remain unavailable until replacement

and subject to criminal prosecution.

export control laws and regulations is prohibited.

n violation of State Law. Any shared accounts will be DISABLED.

!! WARNING: THERE ARE ONLY NIGHTLY BACKUPS OF USER HOME DIRECTORIES. !!

Tutorials OnDemand version: 6ed5e37

This app will launch Tutorials OnDemand on the [ACES cluster](#).

What tutorial would you like to access?

Introduction to Julia

Number of hours (max 10)

3

Email

This field is optional.

Launch

* The Tutorials OnDemand session data for this session can be accessed under the [data root directory](#).

Select "Introduction to Julia" from the dropdown menu

Set number of hours to 1

Click "Launch"

Tutorials OnDemand (280791)

1 node | 2 cores | Running

Host: >_ac009

Created at: 2024-10-28 09:04:21 CDT

Time Remaining: 2 hours and 57 minutes

Session ID: bc632df9-70c2-43b8-a9a9-6fa03546a5fa

Delete


Connect to Tutorials OnDemand: Introduction to Julia


jupyter

File Edit View Run Kernel Settings Help

Not Trusted

Open in... No Kernel

 High Performance
Research Computing
DIVISION OF RESEARCH



Introduction to Julia

Julia is an open source, general purpose programming language designed to combine the speed and performance of lower-level languages (like C and Fortran) with user-friendly syntax found in dynamic, higher-level programming languages. It is especially well-suited for tasks like visualization, large data analytics, and parallel computing. It can be run dynamically via interactive REPL sessions or written, compiled, and run statically. Julia is an ideal choice for researchers and data scientists that prefer the benefits of dynamic programming but still need to solve large computational calculations or numerical simulations with the speed of statically-typed languages.

Running the Julia REPL on the Command Line

After installation, Julia can be launched on the command line by typing `julia` (this will launch the interactive REPL) and can be quit using `exit()` or by typing `Ctrl+d`. You can switch to **shell mode** by typing a semicolon and **help mode** by typing a question mark.

Running the Julia REPL on the Command Line

KEYBINDING	DESCRIPTION
Ctrl + d	Exit (when buffer is empty)
Ctrl + c	Interrupt or cancel



Julia is an open source, general purpose programming language designed to combine the speed and performance of lower-level languages (like C and Fortran) with user-friendly syntax found in dynamic, higher-level programming languages. It is especially well-suited for tasks like visualization, large data analytics, and parallel computing. It can be run dynamically via interactive REPL sessions or written, compiled, and executed as a standalone application. For those who value the benefits of dynamic programming but still need to solve large computational problems, Julia provides a powerful and flexible environment.

After installation, Julia can be launched on the command line by typing `julia` (this will launch the REPL). You can switch to **shell mode** by typing a semicolon and **help mode** by typing a

KEYBINDING

Ctrl + d

Ctrl + c



High Performance
Research Computing
DIVISION OF RESEARCH



Julia is an open source, general purpose programming language with a user-friendly syntax found in dynamic, higher-level languages like Python and MATLAB, but with the performance of compiled computing. It can be run dynamically via an interpreter, but also compiled that prefer the benefits of dynamic programming like Python and MATLAB to languages like C++.

After installation, Julia can be launched on the command line by typing `Ctrl+d`. You can switch to **shell mode** by typing

Select Kernel

Start Preferred Kernel

- ✓ Python 3 (ipykernel)

Use No Kernel

No Kernel

Start Other Kernel

Julio 1102

Julia 1.11.4

Use Kernel from Preferred Session

Use Kernel from Other Session

IntroductionToJulia.ipynb