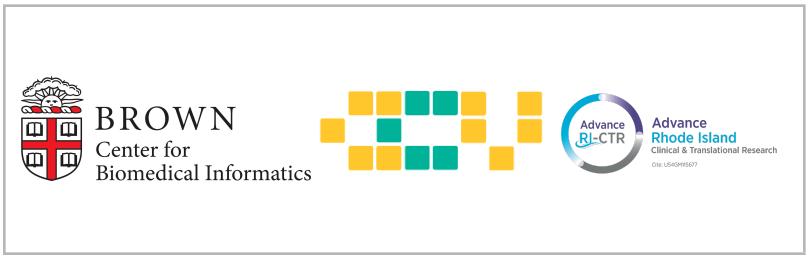


CCV Bootcamp 2023

Health Informatics:
Health Data and Computing Basics

Friday, June 9, 2023 9:00am-12:00pm
Brown AMS Room 270 | Zoom

1. Overview

Instructor/TA Team

- Liz Chen, PhD, FACMI (Contact: liz_chen@brown.edu)
- Neil Sarkar, PhD, MLIS, FACMI, ACHIP
- Karen Crowley, MS, PhD
- Ashlin Harris, PhD
- Farahnaz Maroof, MS
- Mounika Thakkallapally, MS

Description

This hybrid session during CCV Research Computing Bootcamp 2023 will cover basic processing and analysis of an openly-available demo of a de-identified electronic health record (EHR) dataset using Unix commands, SQL, Julia, Python, and R in Oscar.

Note: Brown University provides several computing services that can be used for storage and analysis of electronic health data depending on the Data Risk Classification and requirements of the health data provider. This includes the Oscar high performance computing environment and Stronghold secure computing environment. The Unified Research data Sharing and Access (URSA) Initiative can be leveraged to determine the appropriate environment for a particular health dataset.

Tentative Schedule

9:00am-9:30am	Introduction and Data Review
9:30am-10:00am	Unix Commands
10:00am-10:30am	SQL
10:30am-10:45am	Break
10:45am-11:45am	Julia, Python, and R
11:45am-12:00pm	Wrap Up

2. Prerequisites

Familiarity with skills covered in the following CCV Bootcamp sessions is recommended but not required:

Tuesday, June 6, 2023	Thursday, June 8, 2023
Getting Started on Oscar Using Oscar: Open OnDemand and VSCode Working on the Command Line Basic Bash	Introduction to Julia Python Programming Basics Part I Python Programming Basics Part II An Introduction to R Programming

3. Health Data

Learn more about the Medical Information Mart for Intensive Care (MIMIC) and demo dataset that will be used in this session.

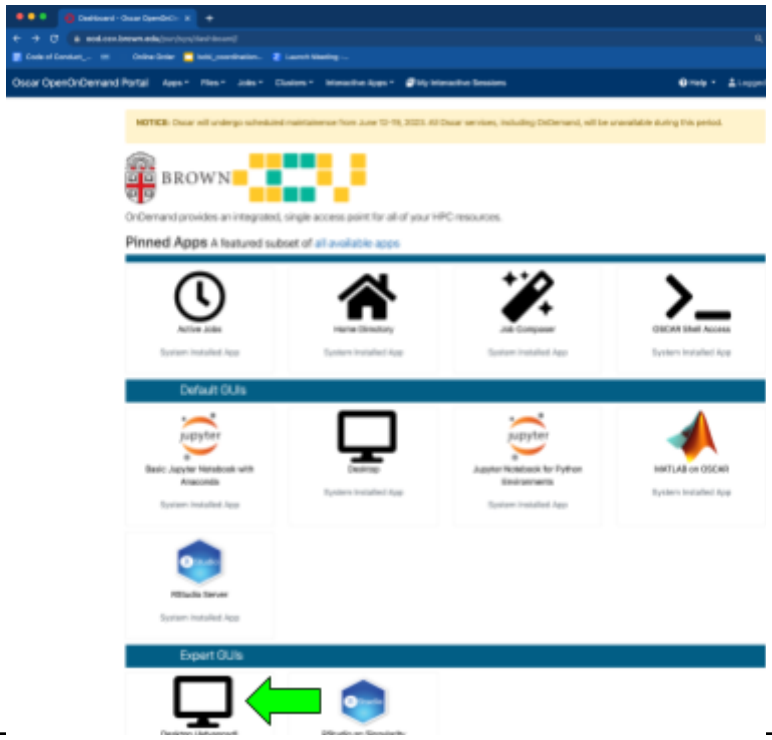
- Johnson AEW, Bulgarelli L, Shen L, Gayles A, Shammout A, Horng S, Pollard TJ, Hao S, Moody B, Gow B, Lehman LH, Celi LA, Mark RG. MIMIC-IV, a freely accessible electronic health record dataset. Sci Data. 2023 Jan 3;10(1):1.
- MIMIC-IV Clinical Database Demo
- MIMIC-IV Documentation

4. Oscar Setup

Get Oscar Account

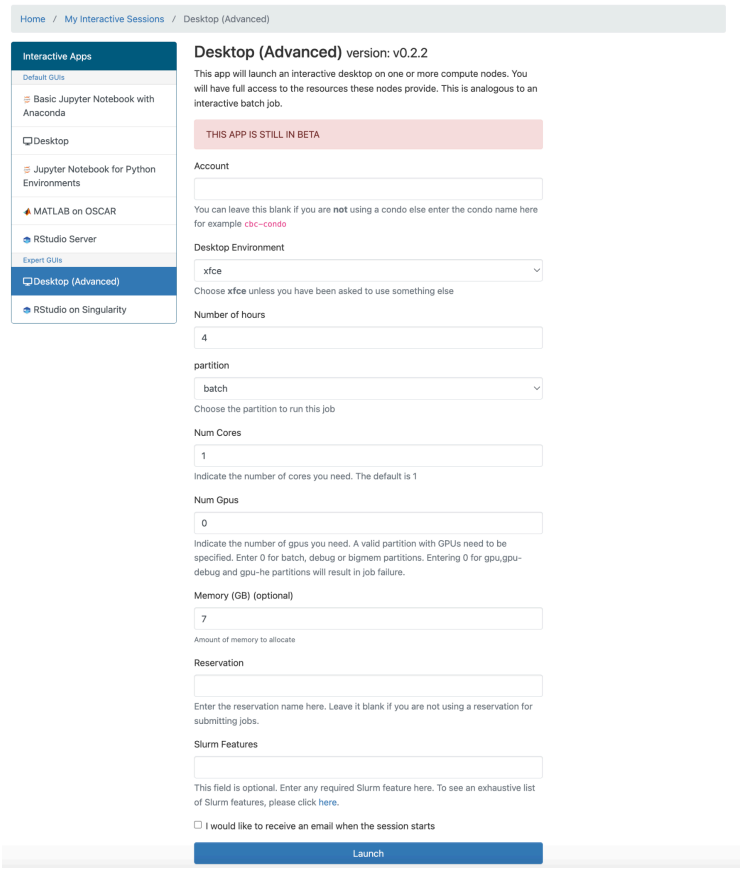
- Register for Oscar Account: https://docs.ccv.brown.edu/oscar/
  - *Note: It may take a few days to process the request.*
- Learn about Oscar: Oscar - Brown University's Cluster
- Adhere to Brown IT Policies: CCV User Policies and OIT Policies
- Review Quickstart Guide
  - OSCAR
  - What username and password should I be using?
  - Connecting to Oscar for the first time
- Watch Short "How to" Videos
- Troubleshoot with Common Problems and FAQ

Login to Oscar using Open on Demand (OOD)

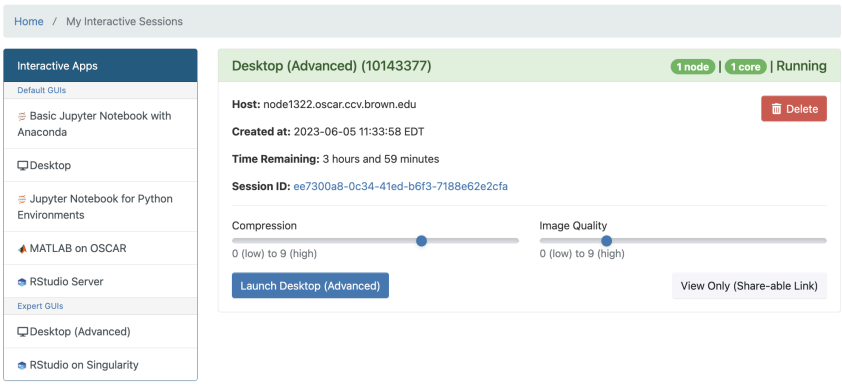
Go to: https://ood.ccv.brown.edu/ Click on "Desktop (Advanced)" under "Expert GUIs"	

Configure interactive desktop (use default values except for the following) and then click “Launch”

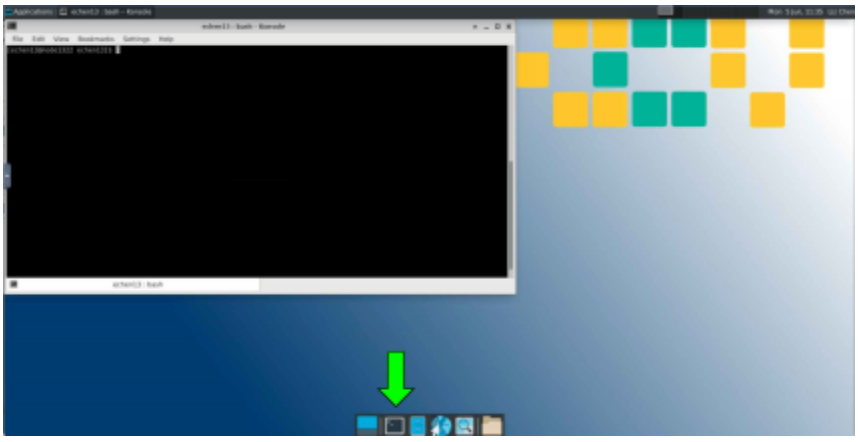
- Number of hours: 4
- Num cores: 1
- Memory: 7 GB



Click “Launch Desktop (Advanced)” once created



Click on “Terminal Emulator” Application in the Launch Bar on the bottom once desktop is launched



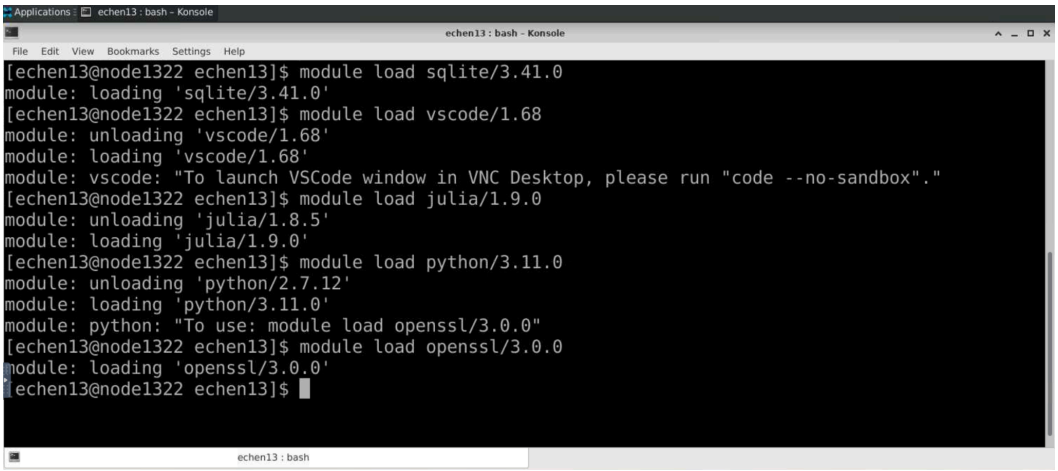
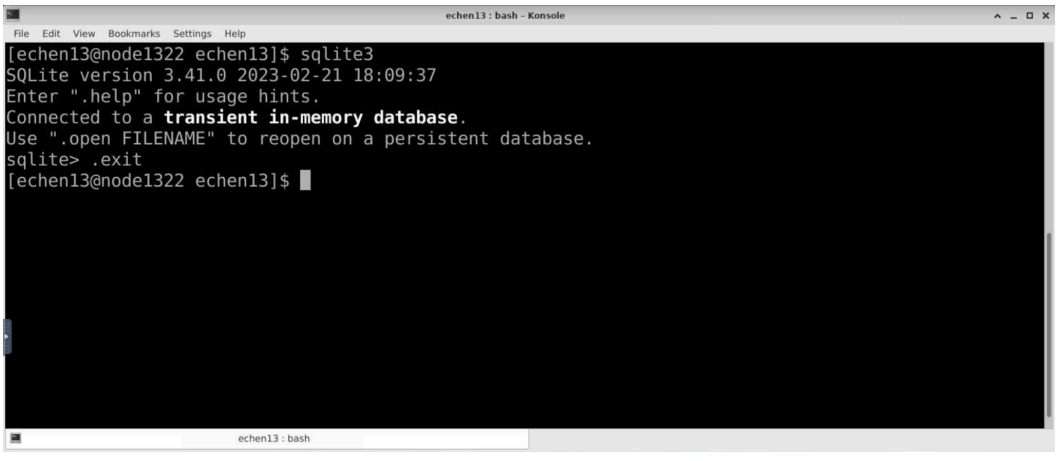
5. Computing Environment Setup

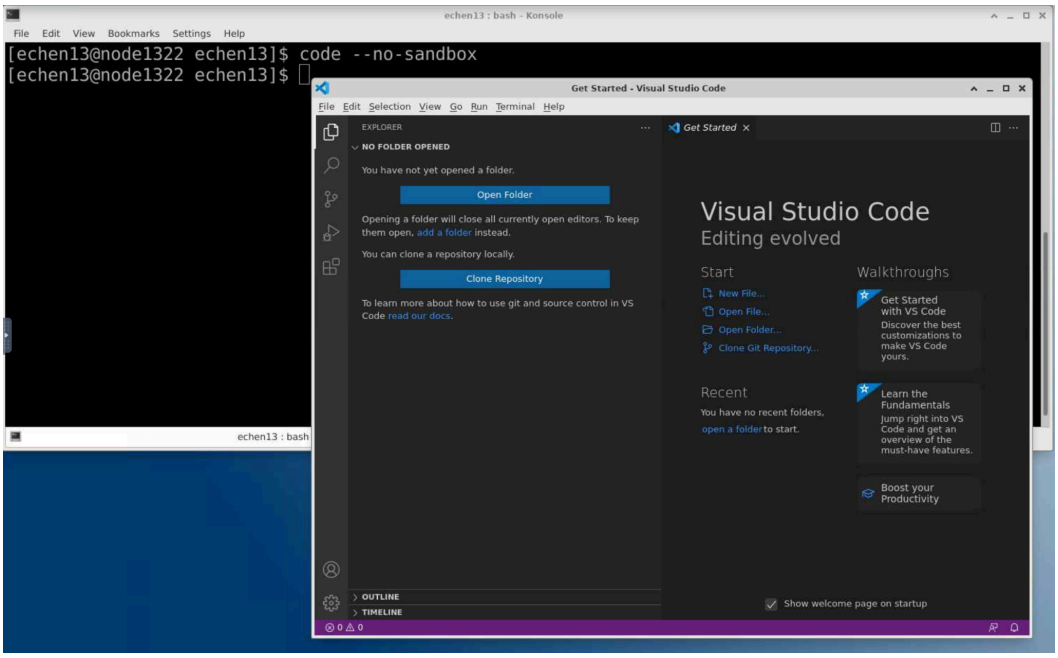
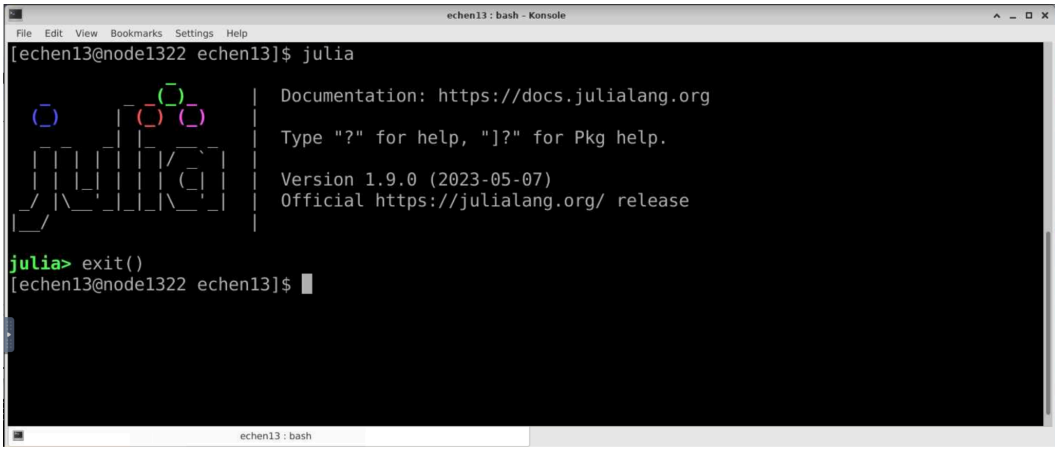
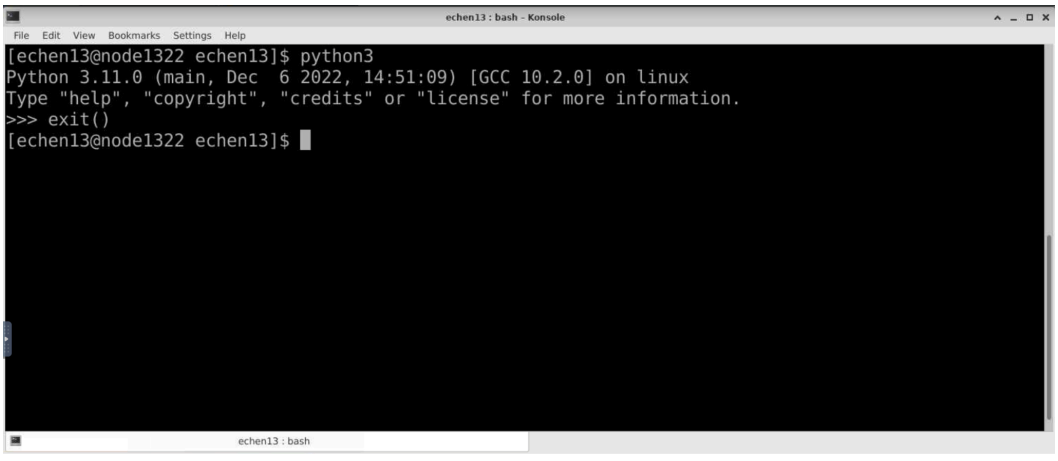
Modules in Oscar

- Help manage the software environment on Oscar
- Can “load” and “unload” modules to configure your computing environment
- Documentation (Using Modules): https://docs.ccv.brown.edu/oscar/software/modules
- Short Video (How to Use Modules in Oscar): https://docs.ccv.brown.edu/oscar/short-videos

Load and Test Modules

- Modules: SQLite, VS Code, Julia, and Python
- Load and test in advance of Friday if possible (more details about each can be found below)

Load Modules	
Test SQLite3	

Test VS Code	
Test Julia REPL	
Test Python	

SQLite

What is SQLite?

- https://www.sqlite.org/index.html
- Written in C programming language
- Small, fast, self-contained, high-reliability, full-featured, SQL database engine

Getting Started with SQLite on Oscar

Find SQLite	**$ which sqlite3** /usr/bin/sqlite3
Check version	**$ sqlite3 --version** 3.7.17 2013-05-20 00:56:22 118a3b35693b134d56ebd780123b7fd6f1497668
Load latest version	**$ module avail sqlite** ~~~~~~~~~~~~~~~~~~~~~~~~ name: sqlite*/* ~~~~~~~~~~~~~~~~~~~~~~~~ sqlite/3.25.2 sqlite/3.41.0 **$ module load sqlite/3.41.0** module: loading 'sqlite/3.41.0'
Check version	**$ sqlite3 --version** 3.41.0 2023-02-21 18:09:37 05941c2a04037fc3ed2ffae11f5d2260706f89431f463518740f72ada350866d
Open SQLite	**$ sqlite3** SQLite version 3.41.0 2023-02-21 18:09:37 Enter ".help" for usage hints. Connected to a transient in-memory database. Use ".open FILENAME" to reopen on a persistent database. sqlite>
Exit SQLite	**sqlite> .exit**

VS Code

What is Visual Studio Code (VS Code)?

- https://code.visualstudio.com/
- Source code editor developed by Microsoft for Linux, MacOS, and Windows
- Support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, etc.

Getting Started with VSCode on Oscar

<table>
<tr><td>Load latest version</td><td>

```
$ module avail vscode
~~~~~~~~~~~~~~~~~~~~~~~~ name: vscode*/* ~~~~~~~~~~~~~~~~~~~~~~~~
vscode/1.22  vscode/1.48  vscode/1.68

$ module load vscode/1.68
module: loading 'vscode/1.68'
module: vscode: "To launch VSCode window in VNC Desktop, please run
"code --no-sandbox"."
```

</td></tr>
<tr><td>Check version</td><td>

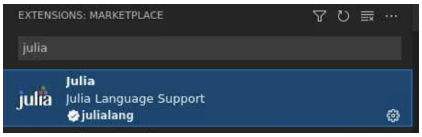
```
$ code --version
3.41.0 2023-02-21 18:09:37
05941c2a04037fc3ed2ffae11f5d2260706f89431f463518740f72ada350866d
```

</td></tr>
<tr><td>Open VSCode</td><td>

```
$ code --no-sandbox
```

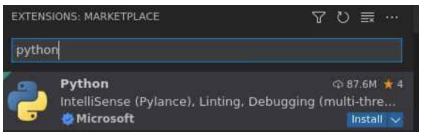
</td></tr>
<tr><td>Turn on Julia Syntax Highlighting</td><td>In VS Code,

Click Extensions ()
Search for Julia
Choose Julia v1.47.2
Click Install



</td></tr>
<tr><td>Turn on Julia Syntax Highlighting</td><td>In VS Code,

Click Extensions ()
Search for Python
Choose Python v2023.8.0
Click Install



</td></tr>
<tr><td>Exit VSCode</td><td>In VSCode, File -> Exit</td></tr>
</table>

Julia

What is Julia?

- https://julialang.org/
- Development started in 2009 at MIT and first appeared in February 2012
- High-level, high-performance dynamic programming language for numerical computing
- Free and open-source
- Provides ease and expressiveness for high-level numerical computing (similar to R, MATLAB, and Python), but also supports general programming

Julia Documentation

- Julia 1.8 Documentation
- Think Julia
- BIDSS Manual - Numbers and Math in Julia
- BIDSS Manual - Control Flow

Getting Started with Julia on Oscar

- Login to Oscar using OOD and open Terminal
- In Terminal

<table>
<tr><td>Load latest version of Julia</td><td>

```
$ module avail julia
~~~~~~~~~~~~~~~~~~~~~~~~~~~ name: julia*/* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
julia/0.5.1  julia/1.0.2  julia/1.5.0  julia/1.6.0  julia/1.7.1  julia/1.8.5
julia/0.6.1  julia/1.1.0  julia/1.5.1  julia/1.6.1  julia/1.7.2  julia/1.9.0
julia/0.6.4  julia/1.2.0  julia/1.5.2  julia/1.6.2  julia/1.7.3
julia/0.7.0  julia/1.4.1  julia/1.5.3  julia/1.6.3  julia/1.8.0
julia/1.0    julia/1.4.2  julia/1.5.4  julia/1.7.0  julia/1.8.2

$ module load julia/1.9.0
module: unloading 'julia/1.8.5'
module: loading 'julia/1.9.0'
```

</td></tr>
<tr><td>Check version</td><td>

```
$ julia --version
julia version 1.9.0
```

</td></tr>
<tr><td>Open Julia REPL</td><td>

```
$ julia

               _
   _       _ _(_)_     |  Documentation: https://docs.julialang.org
  (_)     | (_) (_)    |
   _ _   _| |_  __ _   |  Type "?" for help, "]?" for Pkg help.
  | | | | | | |/ _` |  |
  | | |_| | | | (_| |  |  Version 1.9.0 (2023-05-07)
 _/ |\__'_|_|_|\__'_|  |  Official https://julialang.org/ release
|__/                   |

julia>
```

</td></tr>
<tr><td>Exit Julia REPL</td><td>

```
julia> exit()
```

</td></tr>
</table>

Python

What is Python?

- https://www.python.org/
- High-level, general-purpose programming language
- Emphasizes code readability with use of significant indentation
- Supports multiple paradigms: structured, object-oriented, and functional programming

Python Documentation

- Python 3.11.2 Documentation
- Think Python: How to Think Like a Computer Scientist
- BIDSS Manual - Intro to Python (includes side-by-side comparison with Julia)

Getting Started with Python on Oscar

- Login to Oscar using OOD and open Terminal
- In Terminal

Load latest version of Python3	**$ module avail python** ~~~~~~~~~~~~~~~~~~~~~~~~ name: python*/* ~~~~~~~~~~~~~~~~~~~~~~~~ python/2.7.12 python/3.5.2 python/3.7.4 python/2.7.12_clean python/3.6.6 python/3.8.12_gcc8.3 python/2.7.16 python/3.6.6_test python/3.9.0 **python/3.11.0** python/3.6.8_gcc8.3 python_igraph/0.7.1 **$ module load python/3.11.0** module: unloading 'python/2.7.12' module: loading 'python/3.11.0' module: python: "To use: module load openssl/3.0.0" **$ module load openssl/3.0.0** module: loading 'openssl/3.0.0'
Check version	**$ python3 --version** Python 3.11.0
Open Python	**$ python3** Python 3.11.0 (main, Dec 6 2022, 14:51:09) [GCC 10.2.0] on linux Type "help", "copyright", "credits" or "license" for more information. >>>
Exit Python	**>>> exit()**