

# JuliaPro

(v1.5.4-1)

## Installation Manual and Quickstart Guide

### Contents

1. *Objective*
2. *Prerequisites*
3. *Installing JuliaPro*
  - 3.1. *Installing JuliaPro on legacy Windows systems like Windows 7*
  - 3.2. *Installing JuliaPro through commandline*
4. *Using JuliaPro*
  - 4.1. *Launching JuliaPro*
  - 4.2. *Getting Started with JuliaPro*
5. *Cloud computing with JuliaPro*
6. *Uninstalling JuliaPro*
7. *Trademark Usage*

## 1. Objective

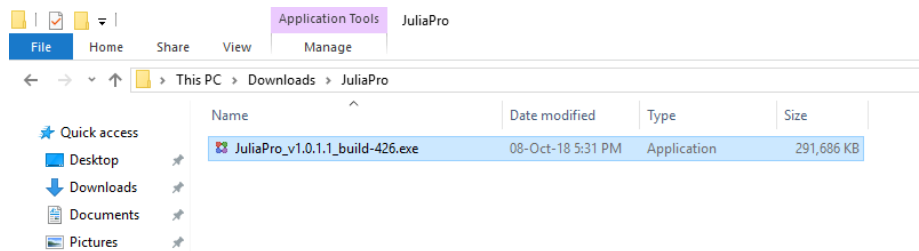
This guide details the installation procedure and usage of the base JuliaPro package and JuliaPro's Juno IDE.

## 2. Prerequisites

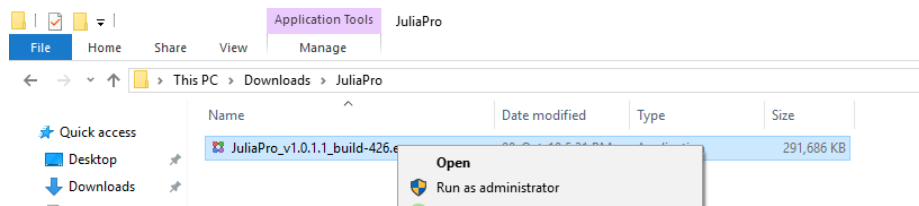
- An appropriate version of Microsoft® Windows®
  - Windows 8.1, Windows 10
  - Windows Server 2012, Windows Server 2012 R2, Windows Server 2016
- 5 GB of disk space
- Active Internet connection
- LinkedIn or Gmail or GitHub or JuliaComputing account (Either one of these accounts are required for authentication)

### 3. Installing JuliaPro

Once the requirements are met, you can start the JuliaPro installation using the executable provided.

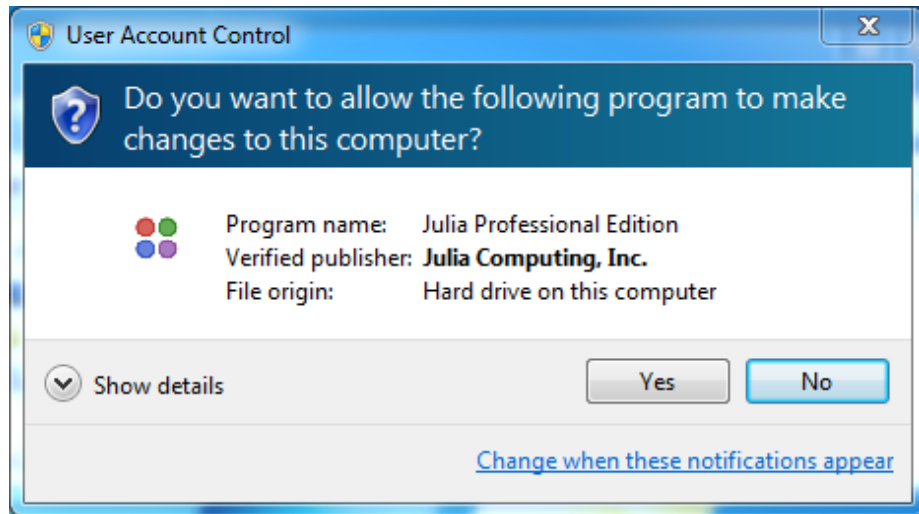


To execute the installer with Administrative privileges, right click on the JuliaPro\_v1.5.4-1.exe and select “Run As Administrator”. To execute the installer as the current user, just double-click on the executable.

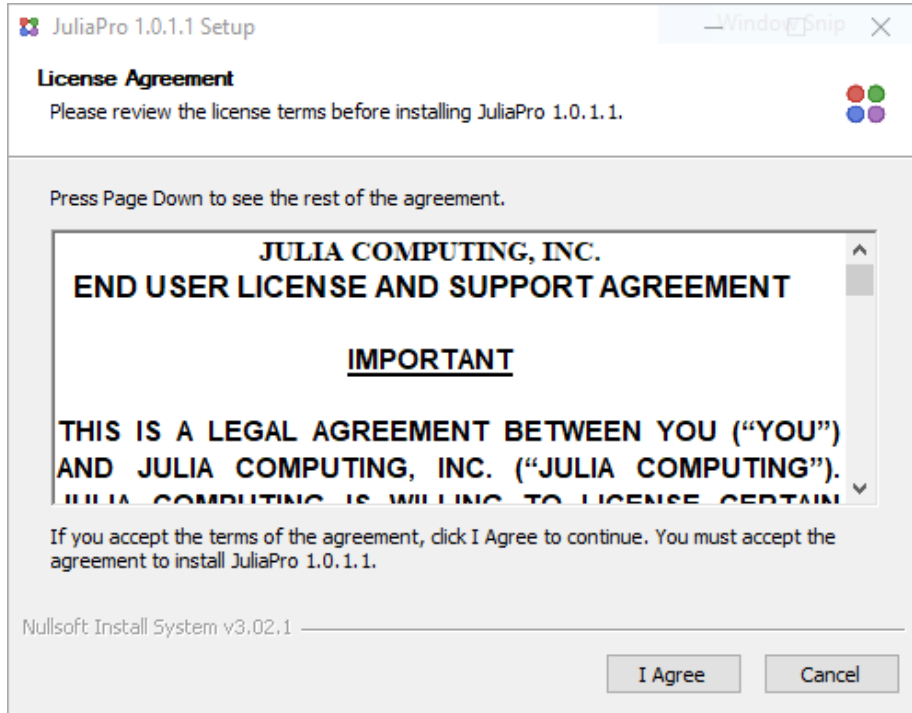


Upon launching the executable, you might be presented with a User Account Control permissions window.

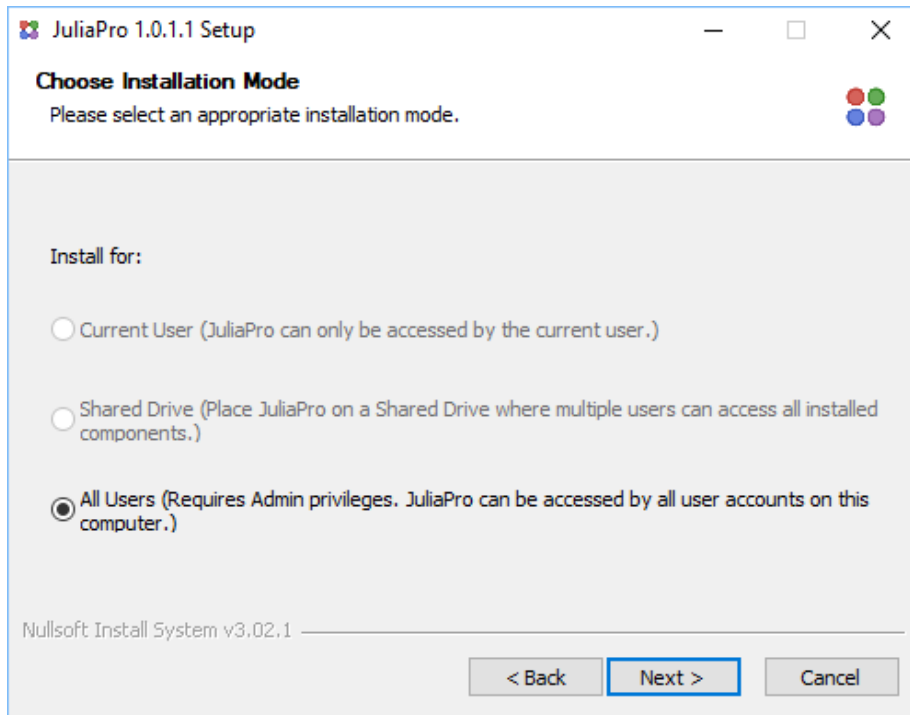
If the installer is being executed with Administrative privileges, then click “Yes” to proceed with the installation.



Upon starting the installation, you will be presented with the JuliaPro Software License Agreement. After reading through the terms mentioned in the agreement, click “I Agree” if you accept the terms of the license and proceed with the installation.

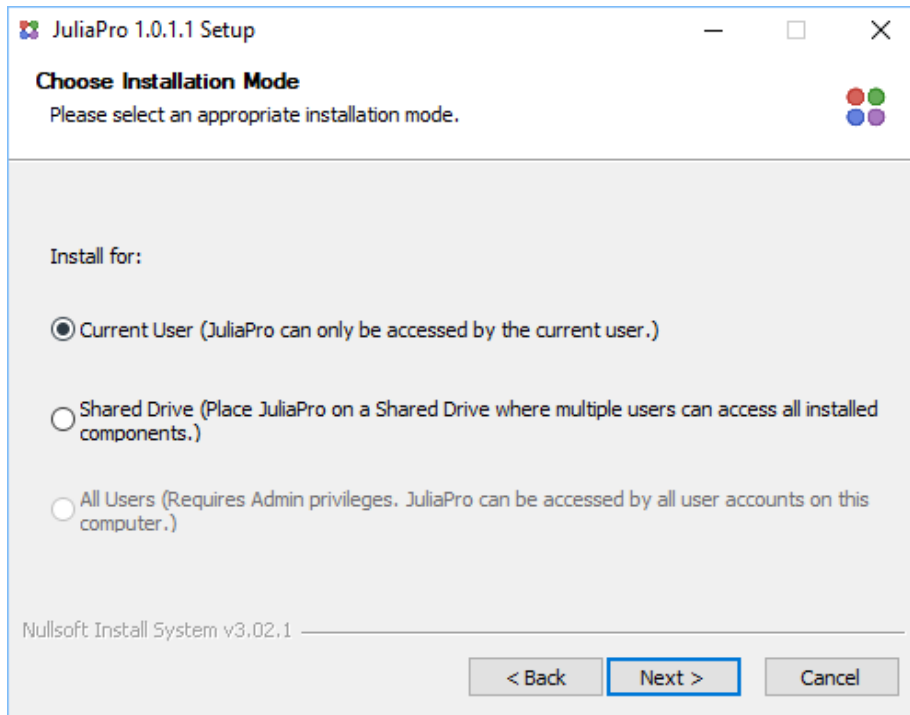


If the installer is being executed with Administrative privileges, then you will be presented with the following installation option:



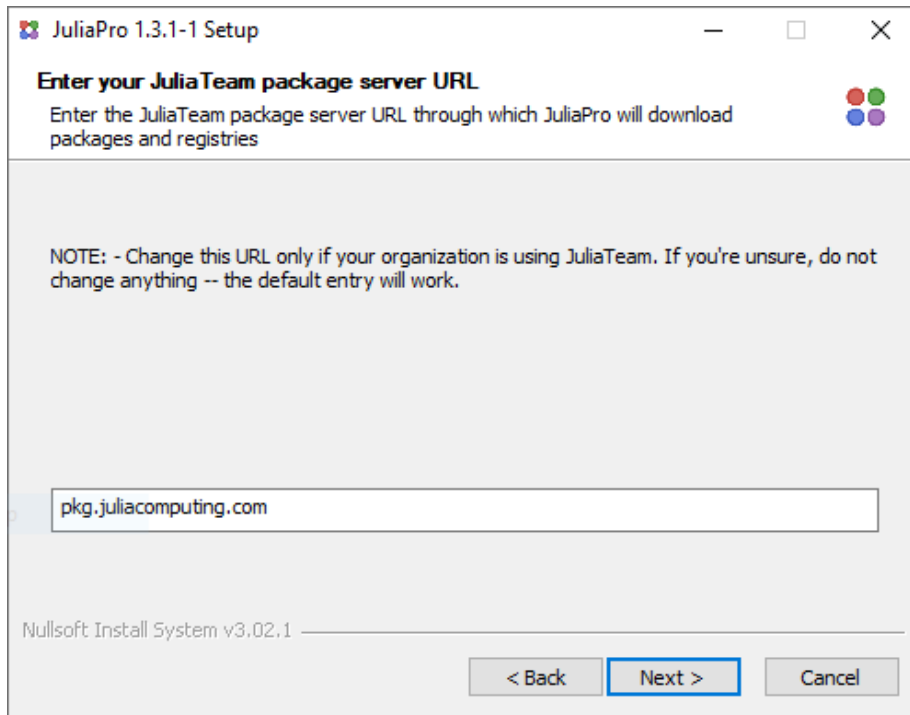
- **All Users:** This installation option requires the current user to have Administrator privileges and will default to installing JuliaPro to a location available to all system users. The default location is a folder created at the location of the current %HOMEDRIVE% environment variable. Selecting this option also allows for a user with Administrative privileges to install Julia Professional to a shared network drive location.

If the installer is being executed without Administrative privileges, then you will be presented with the following two active installation options:



- **Current User (No Admin Privileges Required):** With this installation option, the selected directory into which JuliaPro is installed must be a directory for which the current user has read/write privileges. The default directory chosen is within the “AppData” directory of the current user’s account.
- **Shared Drive (No Admin Privileges Required):** With this installation option, the selected directory into which JuliaPro is installed can be a shared network drive location for which the current user has read/write privileges. While the default directory chosen is within the “AppData” directory of the current user’s account, this option is useful when the current user intends to install Julia Professional to a shared location, but does not have administrative privileges.

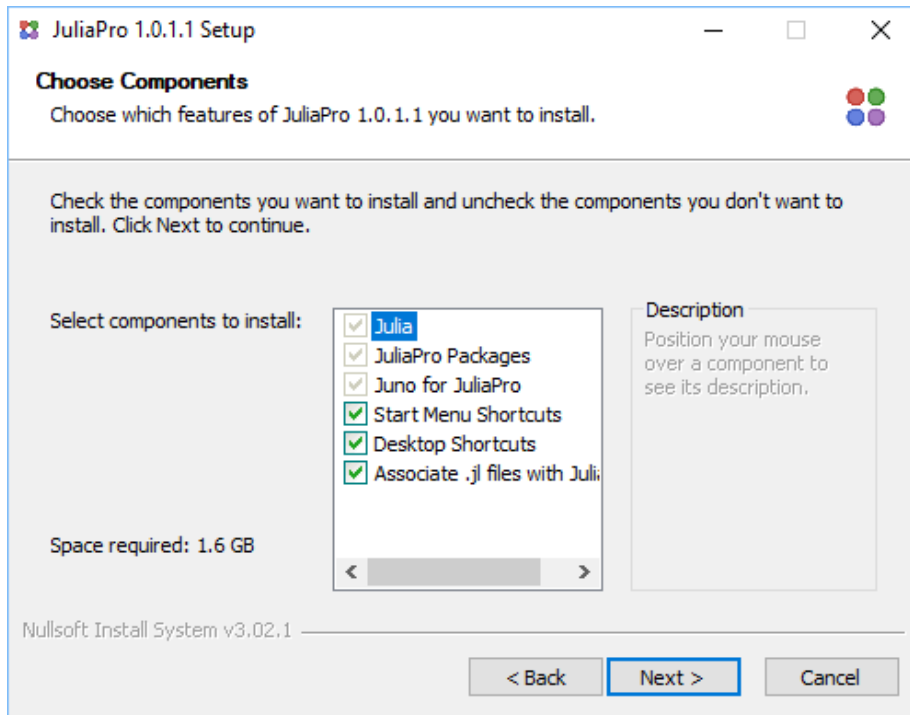
Once you select the mode of installation, you will be presented with JuliaTeam configuration page



This page is applicable only for JuliaTeam customers, if you're an individual or your company or organization is not using JuliaTeam, then you can click on "Next" and continue with the installation. The installer will configure JuliaPro with default package server `pkg.juliahub.com`. You can download (Or update) packages and registries from this server even if you're not a JuliaTeam customer.

If your organization is using JuliaTeam, then you can configure JuliaPro to work with your private package server (JuliaTeam enables you to create private package server) by changing the URL in the text box. Once you enter your private JuliaTeam package server URL in the text box and click on "Next", the installer will configure JuliaPro to download all the packages and registries from your private JuliaTeam server.

Once you're done with your JuliaTeam configuration, you will be presented with options related to: which components of the JuliaPro distribution you would like to install.



The first listed component is the Julia compiler, runtime and terminal application. This component is required for installation.

The next listed component is the set of all “JuliaPro packages” to be installed as part of JuliaPro. This component is also required for installation.

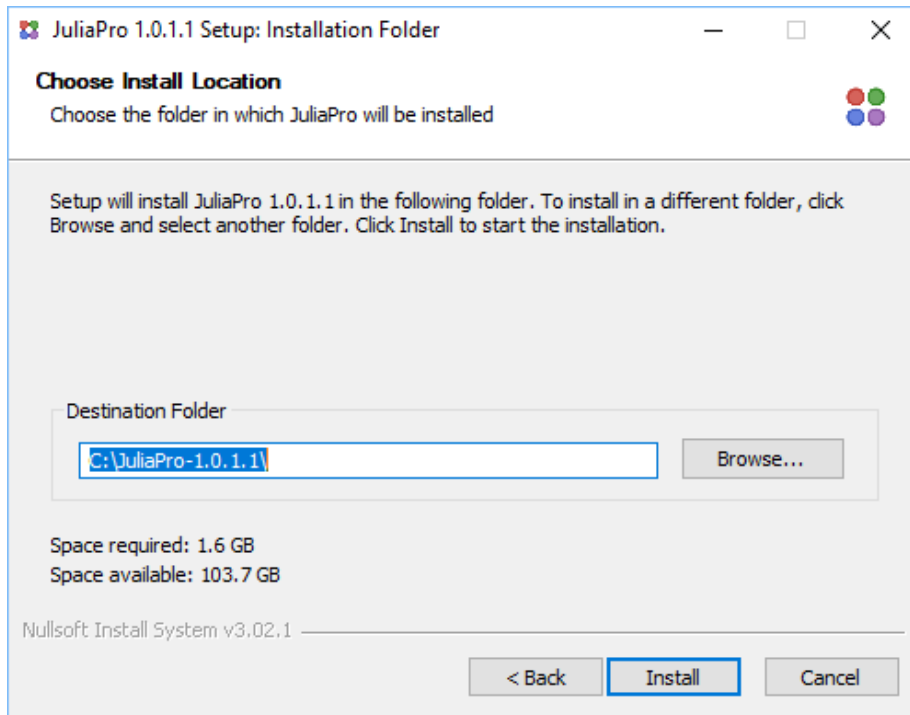
The next listed component includes “Juno for JuliaPro”. Selection of this component will install a copy of the Atom editor onto which various JuliaPro specific enhancements are continually being added.

The subsequent component is the entry for “Start Menu Icons”. By unselecting this entry, no entries for JuliaPro will be added to the Windows Start Menu.

The subsequent component is the entry for “Desktop Shortcuts”. By unselecting this entry, no desktop shortcut icons linking to various JuliaPro components are created.

The final component is the entry for “Associate .jl files with JuliaPro”. By unselecting this entry, “.jl” will no longer be associated with JuliaPro.

Upon selecting your desired components, click “Next” to proceed.

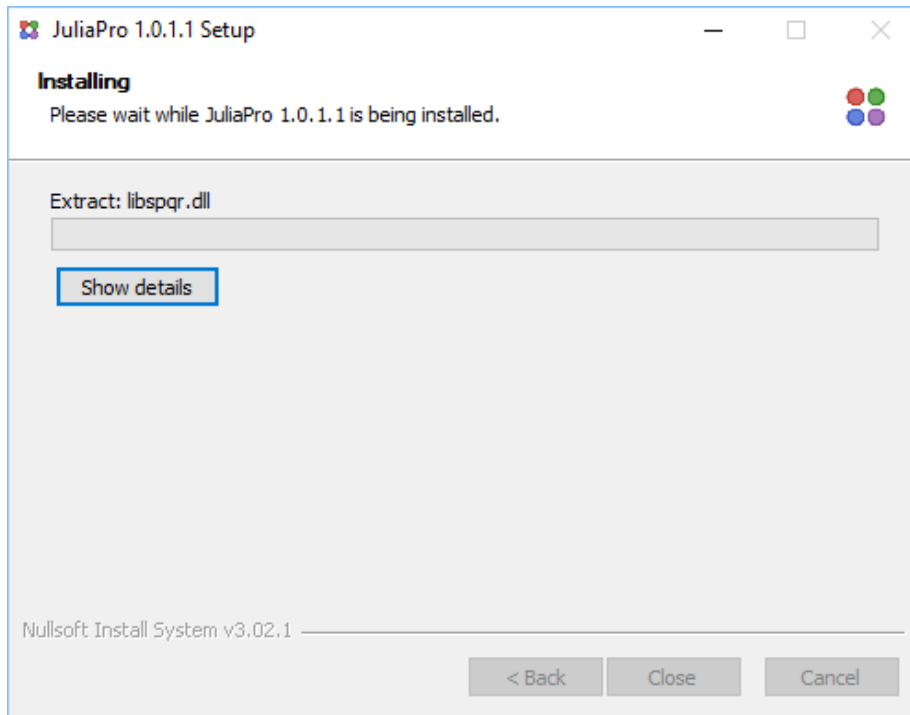


Next, select the folder into which you wish to install JuliaPro. The selection must be a directory into which the current user has read/write privileges. The installer provides an appropriate default location when either the “All Users”, “Current User” options were selected.

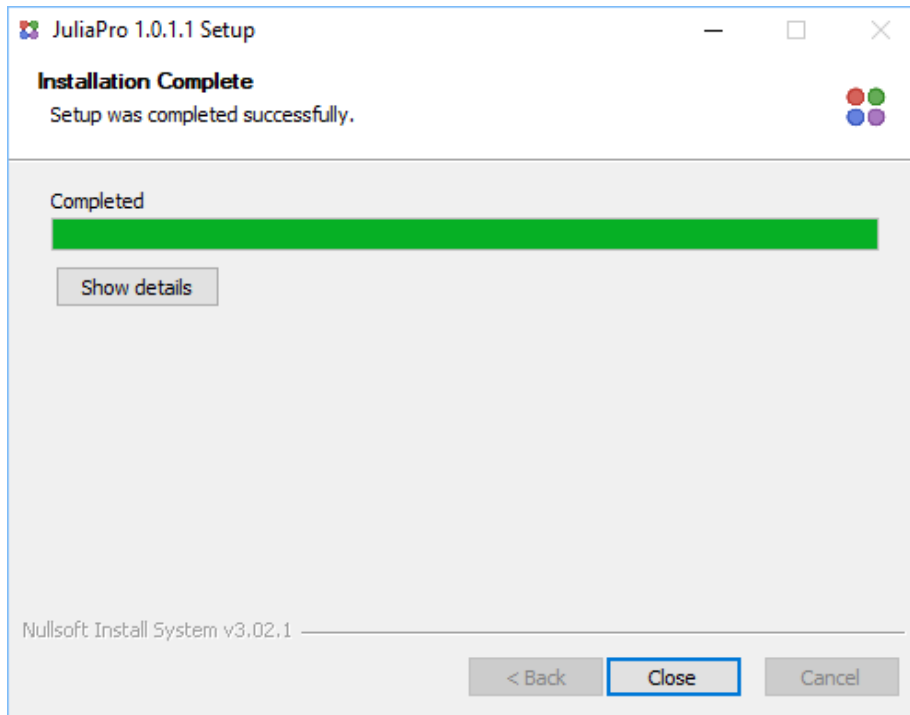
If the “Shared Drive” option was selected by a non-Administrative user, or an Administrative user selects the “All Users” option, then you will likely wish to click the “Browse...” button in order to select a shared network drive location into which to install JuliaPro. Installing JuliaPro to a shared network drive location is useful for teams that plan to share access to a single Julia Professional installation among multiple user accounts on separate machines in the same Active Directory domain.

The following screenshot shows the JuliaPro installer during its installation phase.





Upon completion of the installer, press close to exit the installer.



**NOTE: Setting access permissions on your JuliaPro installation for shared installations.**

For shared network drive installations, if the directory into which JuliaPro was installed is currently accessible by multiple users, then an Administrator may desire to set access permissions for their JuliaPro installation to be read-only for non-Administrative users.

By setting the JuliaPro installation to be read-only, only administrative users will be able to perform operations via Julia's built-in Pkg package manager. Operations such as `Pkg.add`, `Pkg.update`, and `Pkg.rm` can only be performed by users with write access to the `pkgs-1.5.4-1` directory of the Julia Professional installation.

Restricting write access to these directories to a single administrative user will ensure a consistent state of packages for all users of a shared installation.

To prevent unnecessary precompilation of Julia packages from being triggered by separate users, all users accessing a shared drive installation of JuliaPro from a mapped network drive must have the shared drive mapped to the same network drive letter and absolute path.

**NOTE: Copying of Atom Configuration Information on First Launch of Juno.**

For shared network drive installation, the first time a user launches Juno from their client machine, a `.atom` configuration folder will be copied from the Julia Professional installation into the directory `%HOMEDRIVE%\%HOMEPATH%\JuliaPro_Juno_1.5.4-1`. This directory stores user specific configuration information for use of Juno for JuliaPro.

If you're using Windows Server 2012; Then you need to install following additional software's inorder to perform Pkg operations in JuliaPro

- The TLS `easy_fix` for the package manager to work, see this Discourse thread for more details.
- Windows Management Framework 3.0 or later to include PowerShell 3.0 or later.

### 3.1 Installing JuliaPro on legacy Windows systems like Windows 7

NOTE:- Legacy Windows systems like Windows 7, Windows Server 2008 R2 SP1, etc are no longer supported by JuliaPro, following steps can be tried at your own risk. Julia Computing will not provide any support or assistance in installing these software's, nor does it provide any guarantee about the work-ability of JuliaPro after installing on these unsupported operating systems.

If you're using Windows 7 or any other unsupported Windows operating systems; Then you need to install following additional software's inorder to perform Pkg operations in JuliaPro

- The TLS `easy_fix` for the package manager to work, see this Discourse thread for more details.
- Windows Management Framework 3.0 or later to include PowerShell 3.0 or later.

It is recommended that users on these legacy Windows systems install and use a terminal besides `cmd.exe` since the default terminal application has known issues which affect its usability with Julia and other libuv-based cross-platform software.

### 3.2 Installing JuliaPro through commandline

JuliaPro installer supports silent or non-GUI installation, this is especially useful if you would like to install JuliaPro pragmatically (Using a script). You can invoke the silent mode installation by passing `/S` flag to the JuliaPro binary. Following is the shortest possible command that's required to install JuliaPro, after executing following command, JuliaPro binary will create an asynchronous process for the JuliaPro installer and returns back the control of your command prompt (Or PowerShell prompt), you can exit the command or PowerShell

prompt (If required) from which you initiated the JuliaPro installation, the asynchronous process will continue to run in the background and will exit automatically once the installation is complete.

```
JuliaPro_v1.5.0-2.exe /S
```

With above command, JuliaPro will be installed with default values.

Following flags are applicable only when used with /S, all these flags are optional

- /D= Specify the installation directory using this option, it must be the last parameter used in the command line argument and must not contain any quotes, even if the path contains spaces. Only absolute paths are supported.
- /JuliaTeam\_server= Specify your JuliaTeam package server domain name.

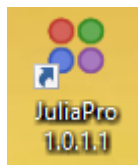
By default all the components are enabled, use following options to disable components during installation.

- /SSHORT Disables creation of shortcuts in StartMenu
- /DSHORT Disables creation of shortcuts on your Desktop
- /FILEASO Disables file association

## 4. Using JuliaPro

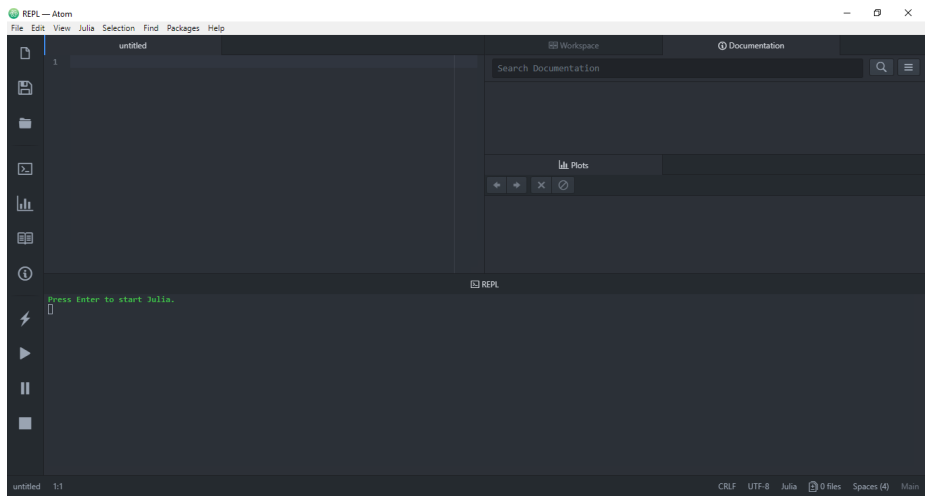
### 4.1 Launching JuliaPro

If the Desktop Shortcuts components were selected during installation of the JuliaPro, then you should have a JuliaPro icon on your desktop. Double-Click on the icon to start IDE.



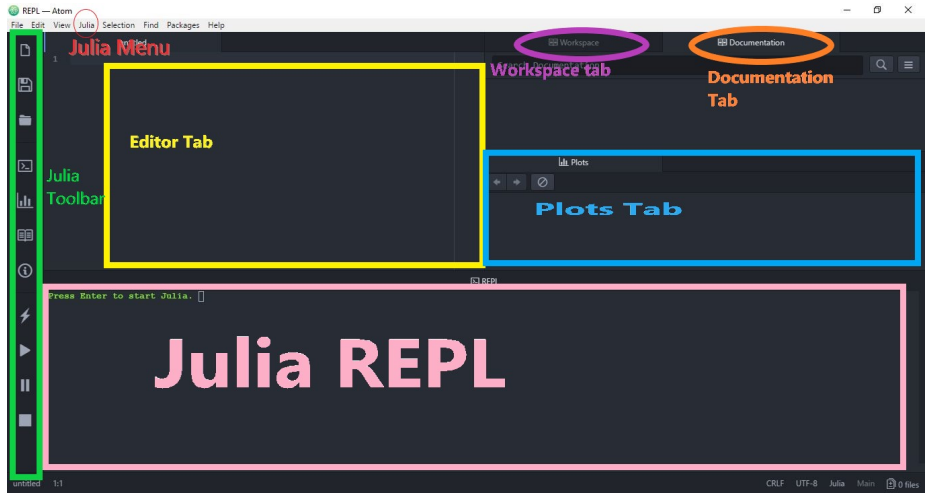
If the Start Menu Shortcuts components were selected during the installation process, then you should have also an entry for the JuliaPro Command Prompt located under “Start -> JuliaPro -> Juno for JuliaPro 1.5.4-1”

Upon initially launching Juno, you will be presented with the following window.

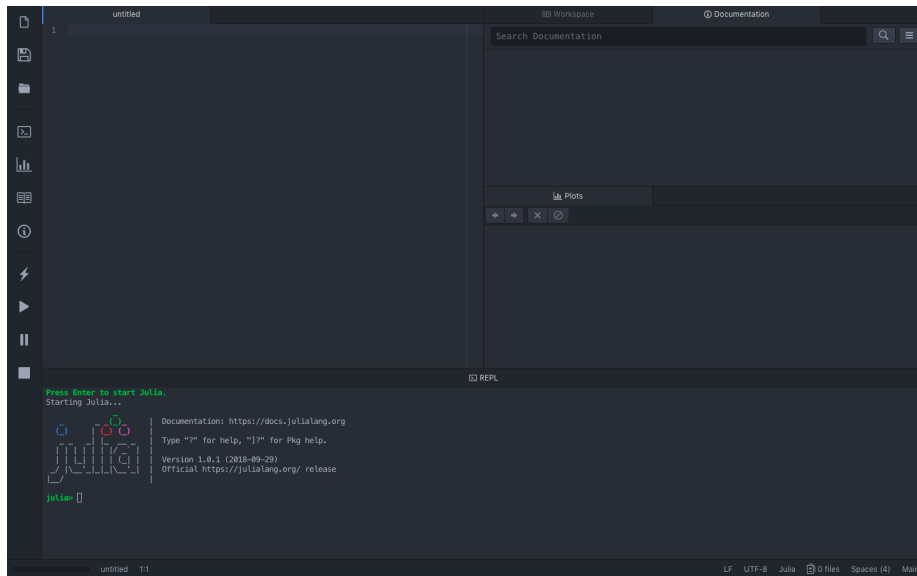


### 4.1 Getting Started with JuliaPro

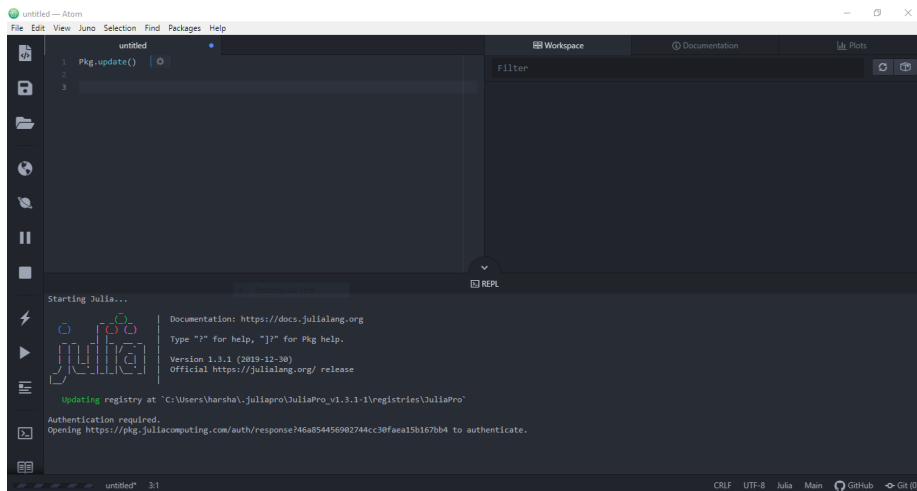
Naming convention for different layout (Panels) in Juno IDE can be seen below



You can start Julia by pressing enter in "Julia REPL" pane



By default, JuliaPro will download all the packages from `pkg.juliahub.com`, this website requires authentication, hence, you will go through the authentication process the first time you perform a `Pkg` operation; JuliaPro will open the authentication link in your default browser, once you successfully authenticate in your browser, JuliaPro will download a unique token for you and place it in the appropriate location. This token will be reused in your subsequent `Pkg` operations.

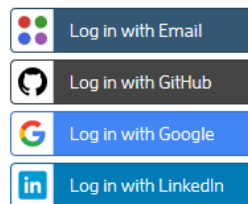


Once the URL link is open in your browser, you will be directed to authenticate by signing into either one of these accounts : LinkedIn or Gmail or GitHub or JuliaComputing account.



## JuliaTeam Authentication

Please proceed to authenticate yourself for package operations.



Once the authentication is done, you will get a message in browser saying “Authenticated. You may close this browser window now.”

**Authenticated. You may close this browser window now.**

Your Pkg operation should continue as is, all your subsequent Pkg operations will use the token that JuliaPro just downloaded.

```
Starting Julia...
Documentation: https://docs.julialang.org
Type "?" for help, "]" for Pkg help.
Version 1.3.1 (2019-12-30)
Official https://julialang.org/ release

Updating registry at `C:\Users\harsha\.juliapro\JuliaPro_v1.3.1-1\registries\JuliaPro`
Authentication required.
Opening https://pkg.juliacomputing.com/auth/response?46a854456902744cc30faea15b167bb4 to authenticate.
Authentication successful.
```

If the automated browser authentication is failing, then please report the issue to JuliaPro@juliacomputing.com and follow below instructions as a fallback to manually download the token and place it in the appropriate location:

NOTE:- YOU DON'T NEED TO FOLLOW THESE INSTRUCTIONS IF YOU SEE FOLLOWING MESSAGE IN YOUR JuliaPro REPL: “Authentication

successful”

1. Open following link in your browser: <https://pkg.juliahub.com/auth>
2. Once you open the URL link, you will be directed to authenticate by signing into either one of these accounts : LinkedIn or Gmail or GitHub or JuliaComputing account.

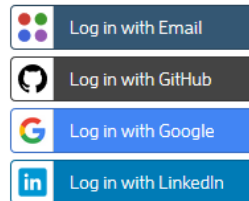


---

## JuliaTeam Authentication

---

Please proceed to authenticate yourself for package operations.



Once the authentication is done, your `token.toml` file download should begin immediately , if your download doesn't begin automatically, you can always click on “here” URL link to download the token

← → ↻ <https://pkg.juliacomputing.com/auth/download/token.toml>

---

### Authenticated to JuliaTeam

---

Your token file should begin downloading now.  
Click [here](#) if it does not.

Once you have `token.toml` , rename this file as `auth.toml` and then you can move this file to the machine where you have installed JuliaPro (If you're downloading token from a different machine), the default location to place this file is `<HOME-FOLDER>\.julia\servers\pkg.juliahub.com\auth.toml` i.e create following path `.julia\servers\pkg.juliahub.com\` in your home directory and place `auth.toml` in the path you just created. Once you place the file in this location, all subsequent Pkg operations will use this token to authenticate



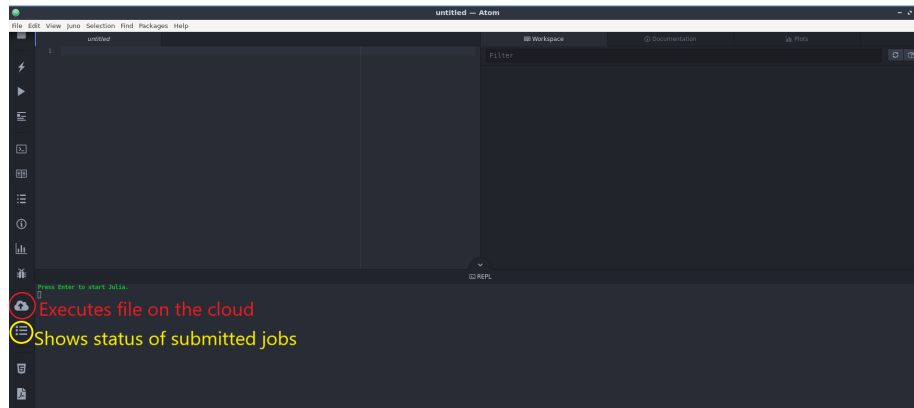
the download from `pkg.juliahub.com`

## 5. Cloud computing with JuliaPro

JuliaPro has built-in support to submit jobs to JuliaHub or your private JuliaRun instance, you can bring up cloud compute UI; either by executing following commands in Atom Command Palette (Press `Ctrl+Shift+P` while focused in an editor pane to see command palette pop up) or by using the cloud compute buttons in the Julia toolbar.

- Julia Run: Run File
- Julia Run: Show Jobs

The cloud compute buttons are located at the bottom of the Julia toolbar, so you need to place the mouse pointer on the toolbar and scroll down to the bottom to locate the cloud compute buttons, relevant cloud compute buttons are shown in below screenshot (Circled in Red and Yellow along with the description).



Following prerequisites need to be satisfied before using the cloud compute feature in JuliaPro

1. You need a valid token for the server to which you're submitting the job, if a valid token is not found; then cloud compute module will trigger the authentication mechanism to download a valid token from the server. In case of JuliaHub, a browser Window will open up requesting you to login using one of the following authentication mechanisms.

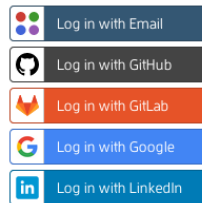


---

## JuliaTeam Authentication

---

Please proceed to authenticate yourself for package operations.



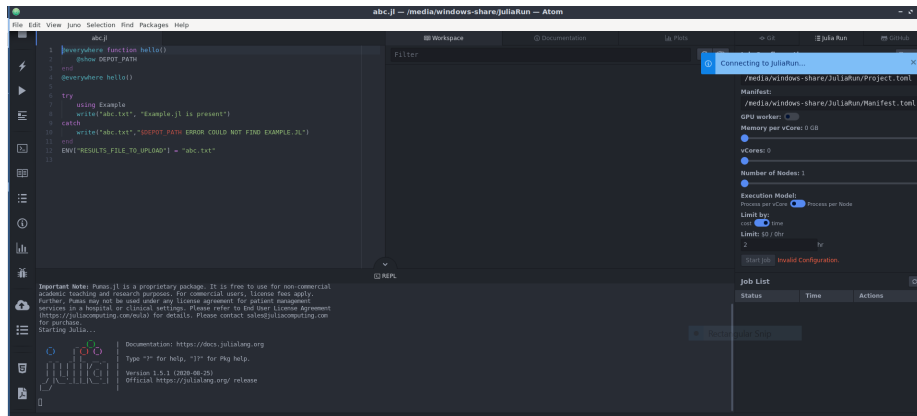
Once you successfully login, you should see following message in your browser, this message indicates the token download was successful, you can return back to the IDE and wait until the cloud compute module connects to the server.

**Authenticated. You may close this browser window now.**

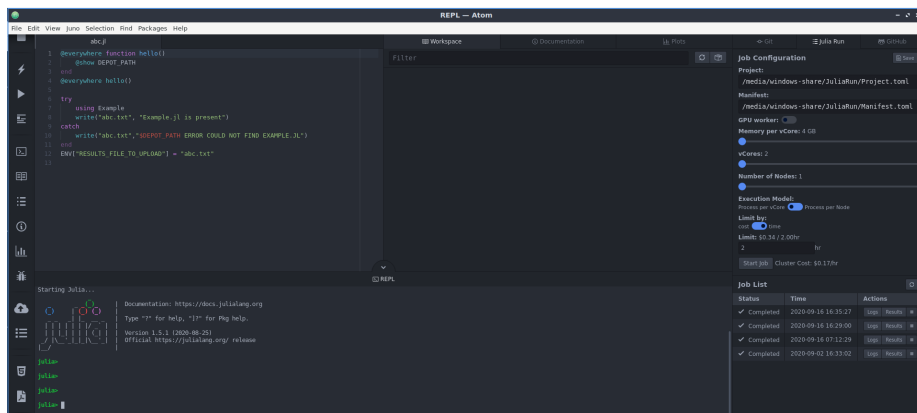
2. If you're submitting jobs to JuliaHub; then make sure you have enough credits to run the job, if you don't have any JuliaHub cloud compute credits; then you can always login to JuliaHub website <https://juliahub.com> and provide payment details to which the cloud compute charges will be billed. If you're submitting jobs to your private JuliaRun instance; then make sure the account associated with `%USERPROFILE%\julia\servers\auth.toml` is authorized to run the jobs on your JuliaRun instance.
3. You should be using JuliaPro v1.5.4-1 or higher

Once all the prerequisites are satisfied, you can follow these steps to start submitting jobs to your server

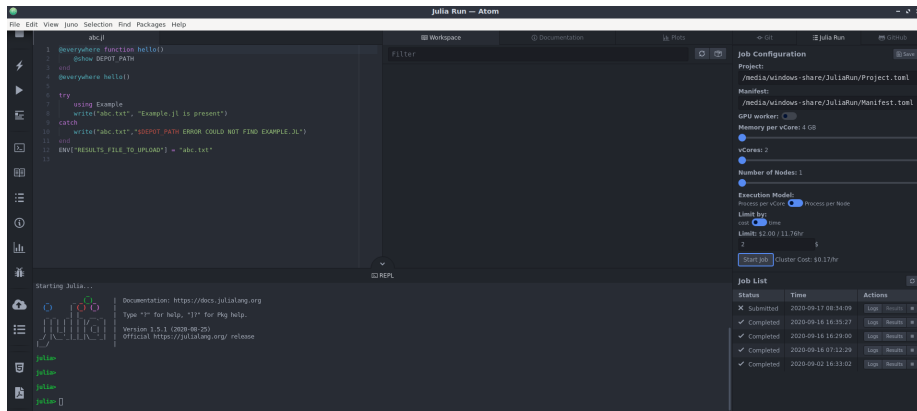
1. Open the project (In JuliaPro IDE) that requires to be ran on the cloud instance
2. Click on the "Show JuliaRun jobs" button from the Julia toolbar or execute following command in Command Palette: **Julia Run: Show Jobs**, this should open up the cloud compute UI, you should also see a message saying "Connecting to JuliaRun"



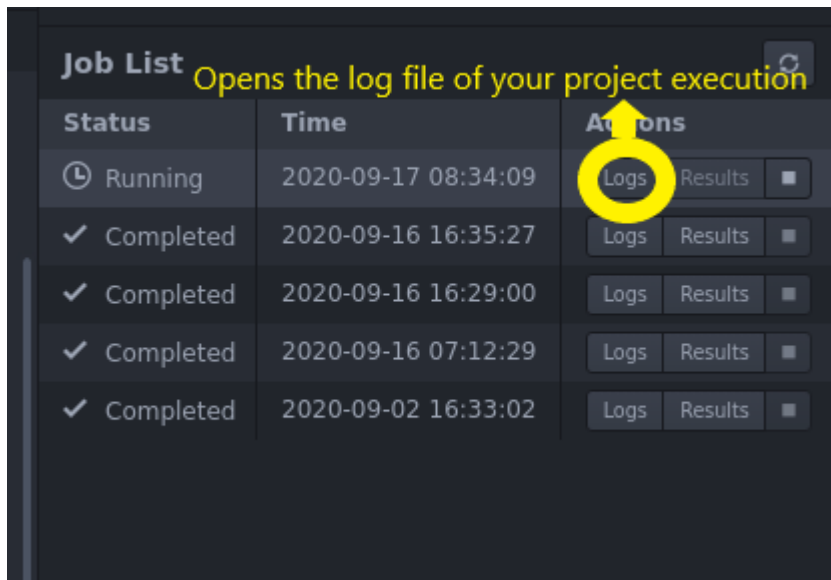
3. Wait until the cloud compute module connects to your server, if all the prerequisites are satisfied, you should see the “Start job” button enabled once the connection is established, you should also be able to see all the past jobs that you submitted (If you have any).



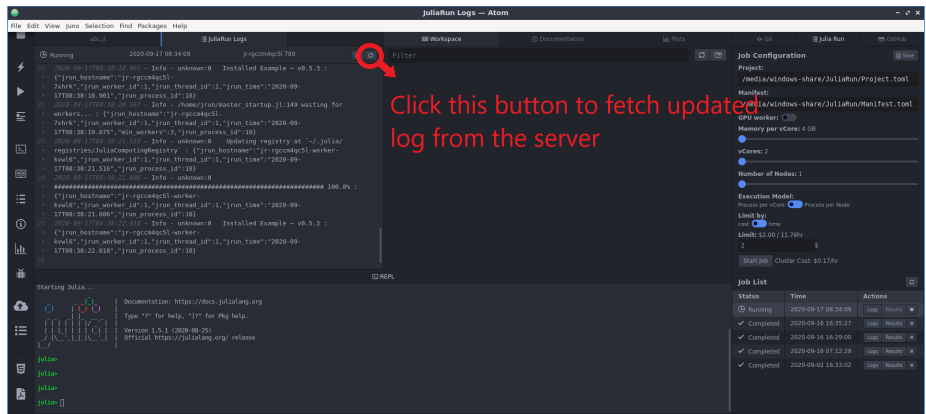
4. Select the appropriate configuration for the workers over which your project will be ran, if you’re using JuliaHub; then make sure to limit your expenditure for the current job, either based on time or by Dollars using “Limit by” toggle button, the estimate cost per hour is displayed in the same configuration window, this cost is relative to the worker configuration you selected. For more details regarding CPU and GPU configurations, please refer to following documentation: <https://docs.juliahub.com/>
5. Click on “Start job” button to execute your project on the cloud instance, you should see a popup in your IDE saying “Job submitted”, your job should also appear in the “Job list”



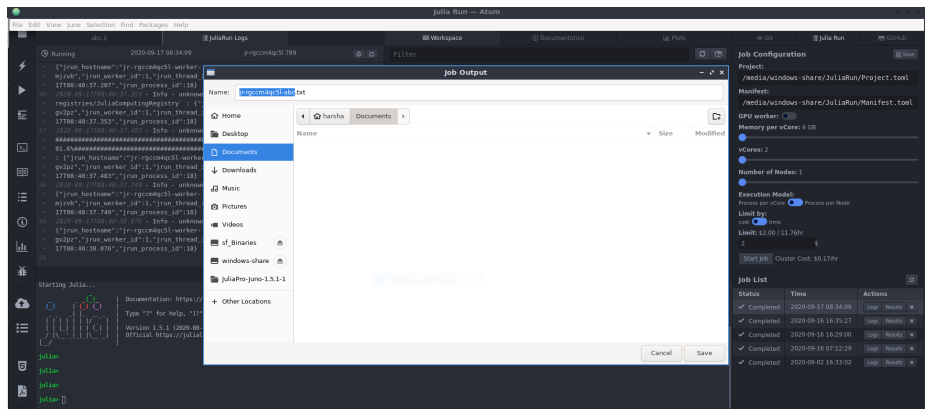
- Once the status of your job changes from “Submitted” to “Running”, you can click on “Logs” button next to your job to see the progress of your project execution.



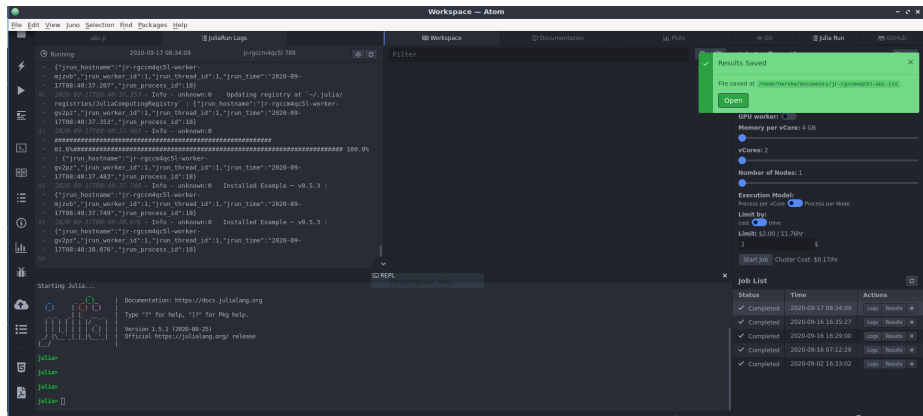
- A log file should open up in log viewer, these logs are not “live”, you need to hit the refresh button in order to fetch the latest logs from the server.



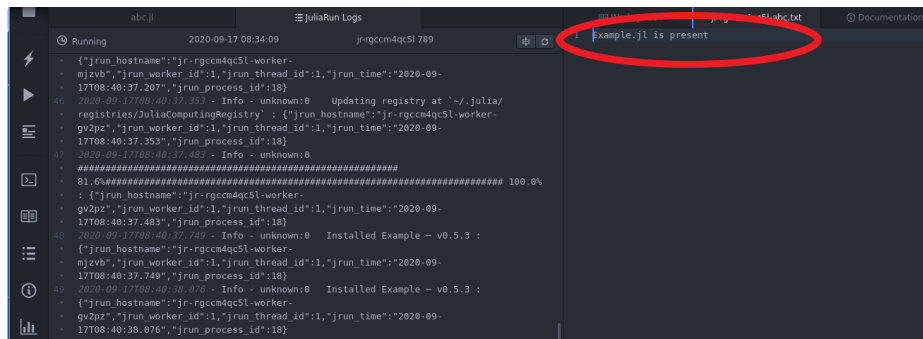
- Once your project execution completes, the results button will get enabled (If your script stored results in a file) and the job status should now say “Completed”, you can click on the “Results” button to download your results from the server. An explorer Window should open up as soon as you click on the “Results” button, you can navigate and store the results file in an appropriate location.



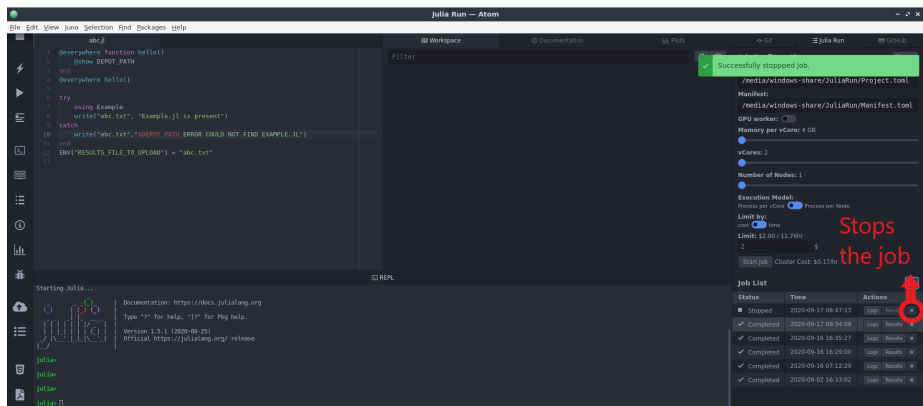
- Once the result file download completes, you should see a popup confirming the same



10. You can choose to open the results file in Atom IDE or any other software to validate it.



11. You can interrupt a “Running” job by clicking on the “Stop” button, this button is enabled only if the job status is “Running”, you should also see a popup that confirms your job was interrupted successfully and job status on interrupted jobs should be “Stopped”



Below source code was used in above example

```

@everywhere function hello()
    @show DEPOT_PATH
end
@everywhere hello()

try
    using Example
    write("abc.txt", "Example.jl is present")
catch
    write("abc.txt", "$DEPOT_PATH ERROR COULD NOT FIND EXAMPLE.JL")
end
ENV["RESULTS_FILE_TO_UPLOAD"] = "abc.txt"

Contents of Project.toml and Manifest.toml in the example project

[deps]
Example = "7876af07-990d-54b4-ab0e-23690620f79a"

# This file is machine-generated - editing it directly is not advised

[[Example]]
git-tree-sha1 = "46e44e869b4d90b96bd8ed1fdcf32244fddfb6cc"
uuid = "7876af07-990d-54b4-ab0e-23690620f79a"
version = "0.5.3"

```

## 6. Uninstalling JuliaPro

Please use `Add or Remove Programs` to uninstall JuliaPro on Windows, this will make sure the uninstaller will get executed with appropriate privileges required to remove JuliaPro installation.

JuliaPro stores end-user profiles outside the installation directory to ease the upgrade process, if you're not planning on reusing the contents of these folders, then you can remove following directories manually:

- Julia packages that are located in following path: `<Home-Directory>\.julia`
- IJulia kernel located in following path:  
`<Home-Directory>\AppData\Roaming\jupyter\kernels\juliapro_v1.5.4-1-1.5`  
 If IJulia kernels were created for system wide usage then:  
`%PROGRAMDATA%\jupyter\kernels\juliapro_v1.5.4-1-1.5`
- Atom cache folder in following location: `<Home-Directory>\AppData\Roaming\Atom`
- Atom package directory: `<Home-Directory>\.atom`

NOTE:- Other applications might be using above directories, please be cautious while removing above folders.

## **7. Trademark Usage**

Microsoft® and Windows® are registered trademarks of Microsoft Corporation.

Other names may be trademarks of their respective owners.