



Source: <https://julialang.org/>

安裝及建置 Julia 開發環境

下載

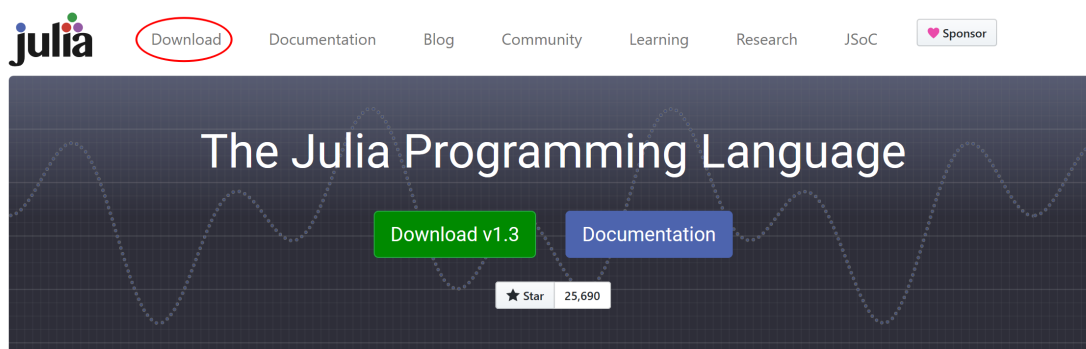
JuliaLang.org 提供的 open source Julia 可以從官網下載，選擇自己適合的版本點選下載。本文介紹將以 Windows 10 環境為主要範例。

範例環境設定：

Windows 10 64-bit 1909

Julia 1.2 (Open Source)

進入 JuliaLang 官網 (<https://julialang.org/>) 首頁，可點 Download 連結進入下載頁面。



目前 Julia 最新版本為 1.3.1 版，馬拉松內容會以 1.2.0 為基礎，但範例程式均在 1.1.1 – 1.3.1 版本中測試過。若要下載安裝 1.2.0 版，進入 Download 頁面後，進入 Older releases page 即可下載到 1.2.0 版。

Older Releases

Older releases of Julia for all platforms are available on the [Older releases page](#). Only the LTS and Stable releases are maintained.

進入下載頁面後，依自己的作業系統版本，下載相對應的版本。以下範例以 Windows 10 64 位元版本為例，下載 64 位元版本的自我解壓縮安裝檔案。馬拉松課程採用的穩定版本為 1.2.0 (Aug 20, 2019) 版，下載檔案大小約 57MB。

Older Unmaintained Releases

Old releases are available should you need to use them to run Julia code written for those releases. Note that these are not actively developed nor maintained anymore.

v1.2.0 (Aug 20, 2019)

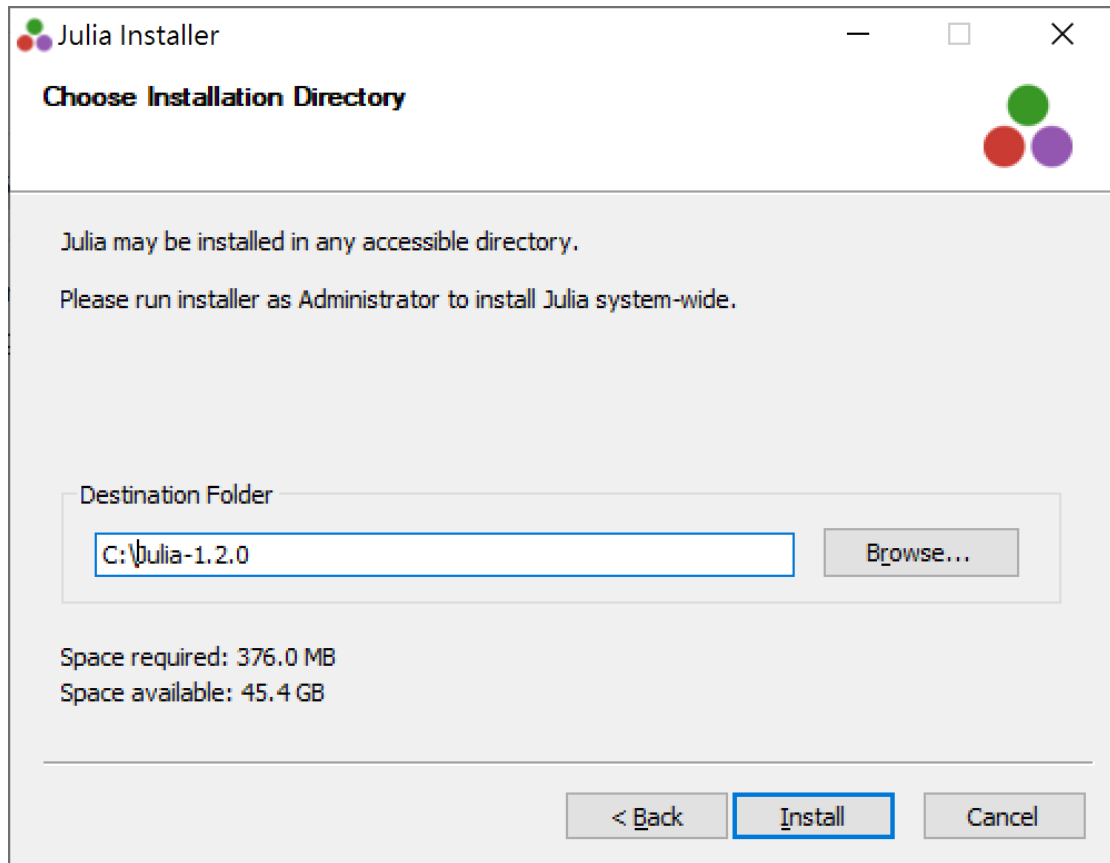
Windows (.exe) [help]	32-bit	64-bit	
macOS 10.8+ (.dmg) [help]		64-bit	
Generic Linux Binaries for x86 [help]	32-bit (GPG)	64-bit (GPG)	
Generic Linux Binaries for ARM [help]	32-bit (ARMv7-a hard float) (GPG)	64-bit (AArch64) (GPG)	
Generic FreeBSD Binaries for x86 [help]		64-bit (GPG)	
Source	Tarball (GPG)	Tarball with dependencies (GPG)	GitHub

下載頁面也提供原始碼下載，也可以透過編譯原始碼的方式進行安裝。

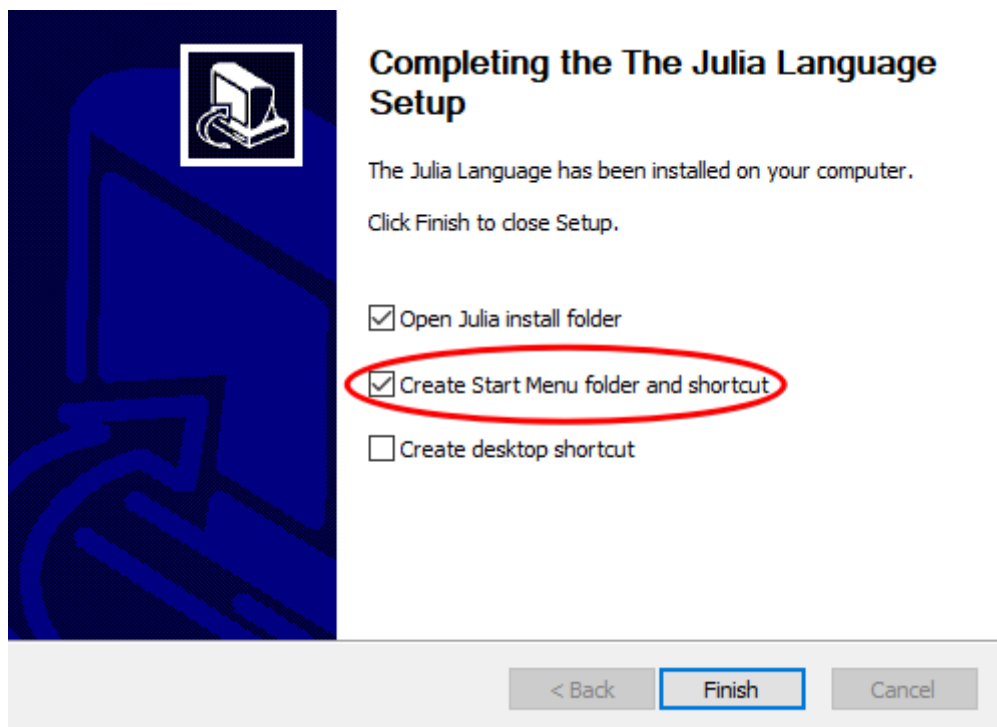
安裝

Windows 安裝很簡單，執行安裝檔即可。

建議安裝完成後，在 Windows 環境變數下將 Julia 執行檔路徑 (e.g. `c:\Julia-1.2.0\bin`) 加到環境變數下，未來若要從命令提示字元下啟動 Julia 的話比較方便。



維持 Create Start Menu folder and shortcut 勾選。



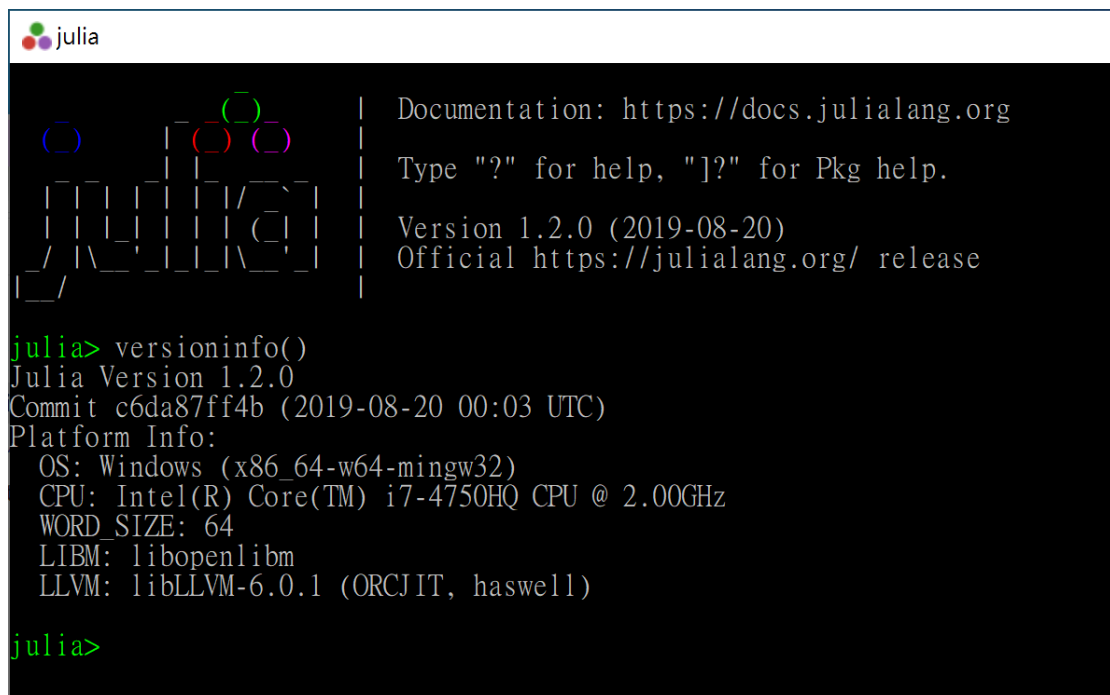
啟動

安裝完成後，可以透過下列三種方式啟動 Julia (以 Windows 為例)：

- 從開始選單
- 從桌面捷徑 (若在安裝時有勾選要新增桌面捷徑)
- 從命令提示字元

啟動進入 Julia REPL 命令列模式，輸入下列指令顯示系統資訊。

`versioninfo()`



```
┌───┐
│   │
│   │
│   │
└───┘
```

```
julia Documentation: https://docs.julialang.org
Type "?" for help, "]" for Pkg help.
Version 1.2.0 (2019-08-20)
Official https://julialang.org/ release

julia> versioninfo()
Julia Version 1.2.0
Commit c6da87ff4b (2019-08-20 00:03 UTC)
Platform Info:
 OS: Windows (x86_64-w64-mingw32)
 CPU: Intel(R) Core(TM) i7-4750HQ CPU @ 2.00GHz
 WORD_SIZE: 64
 LIBM: libopenlibm
 LLVM: libLLVM-6.0.1 (ORCJIT, haswell)

julia>
```

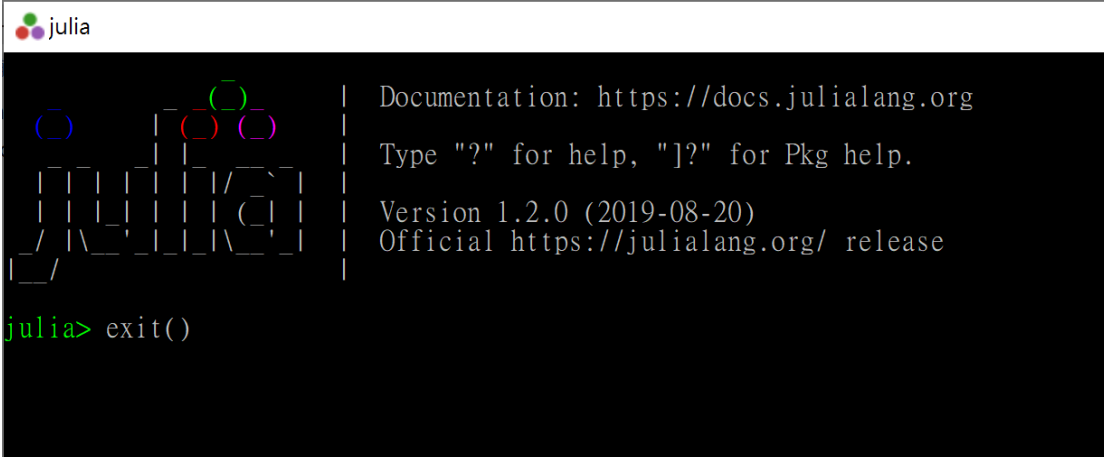
macOS 安裝完成後，可以從應用程式中啟動。

Linux 版本的安裝，將編譯好的下載檔解壓縮，並建立 Symbolic Link (Optional) 即可執行啟動 Julia。

退出

輸入下列指令可退出 Julia REPL 命令提示字元，關閉視窗：

```
exit()
```



```
┌───┐ julia  
└───┘  
┌───┐ | Documentation: https://docs.julialang.org  
└───┘ | Type "?" for help, "]" for Pkg help.  
┌───┐ | Version 1.2.0 (2019-08-20)  
└───┘ | Official https://julialang.org/ release  
julia> exit()
```

Jupyter Notebook 開發環境

下面將簡單介紹如何建置 Julia 開發環境，以 Jupyter Notebook 為範例。

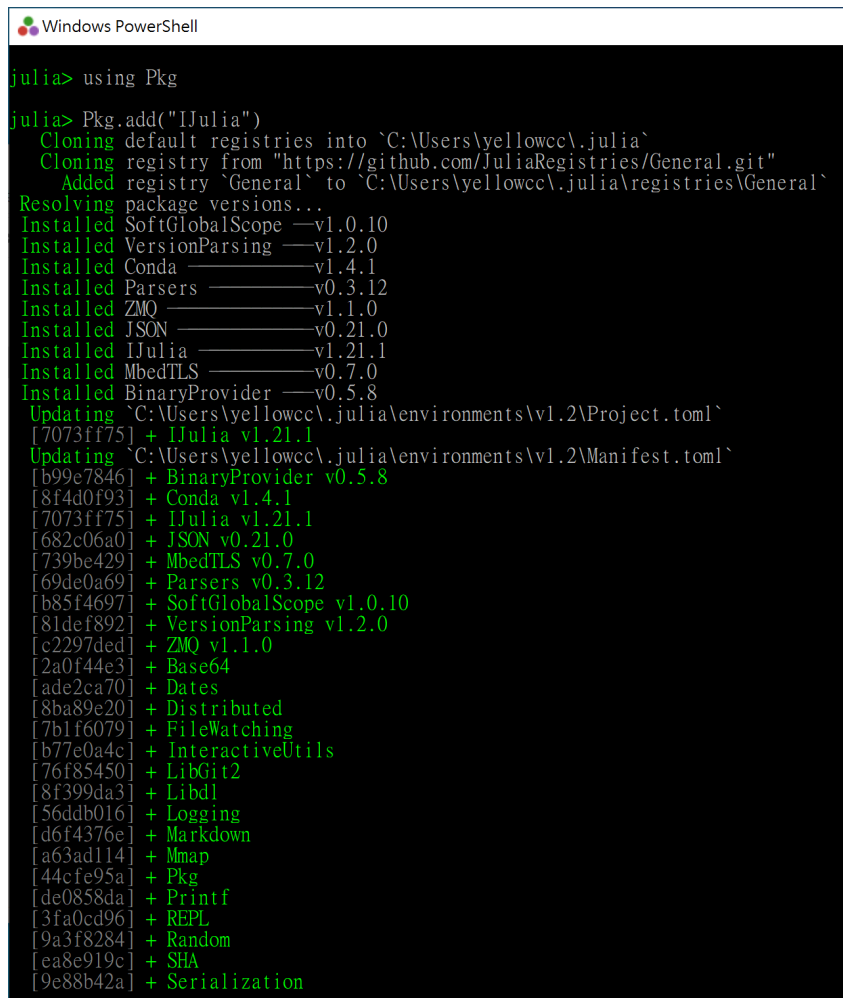
Julia + Jupyter Notebook

下列說明是介紹在 Julia 環境下安裝 Jupyter。如果原先在電腦上就有在 Python 環境安裝 Jupyter，而且想要使用既有安裝版本的話，請參考下段說明。

1. 透過 Pkg (Package Management) 安裝 IJulia。

```
julia> using Pkg
```

```
julia> Pkg.add("IJulia")
```



```
Windows PowerShell

julia> using Pkg
julia> Pkg.add("IJulia")
Cloning default registries into `C:\Users\yellowcc\.julia`
Cloning registry from "https://github.com/JuliaRegistries/General.git"
Added registry `General` to `C:\Users\yellowcc\.julia\registries\General`
Resolving package versions...
Installed SoftGlobalScope v1.0.10
Installed VersionParsing v1.2.0
Installed Conda v1.4.1
Installed Parsers v0.3.12
Installed ZMQ v1.1.0
Installed JSON v0.21.0
Installed IJulia v1.21.1
Installed MbedTLS v0.7.0
Installed BinaryProvider v0.5.8
Updating `C:\Users\yellowcc\.julia\environments\v1.2\Project.toml`
 [7073ff75] + IJulia v1.21.1
Updating `C:\Users\yellowcc\.julia\environments\v1.2\Manifest.toml`
 [b99e7846] + BinaryProvider v0.5.8
 [8f4d0f93] + Conda v1.4.1
 [7073ff75] + IJulia v1.21.1
 [682c06a0] + JSON v0.21.0
 [739be429] + MbedTLS v0.7.0
 [69de0a69] + Parsers v0.3.12
 [b85f4697] + SoftGlobalScope v1.0.10
 [81def892] + VersionParsing v1.2.0
 [c2297ded] + ZMQ v1.1.0
 [2a0f44e3] + Base64
 [ade2ca70] + Dates
 [8ba89e20] + Distributed
 [7b1f6079] + FileWatching
 [b77e0a4c] + InteractiveUtils
 [76f85450] + LibGit2
 [8f399da3] + Libdl
 [56ddb016] + Logging
 [d6f4376e] + Markdown
 [a63ad114] + Mmap
 [44cfe95a] + Pkg
 [de0858da] + Printf
 [3fa0cd96] + REPL
 [9a3f8284] + Random
 [ea8e919c] + SHA
 [9e88b42a] + Serialization
```

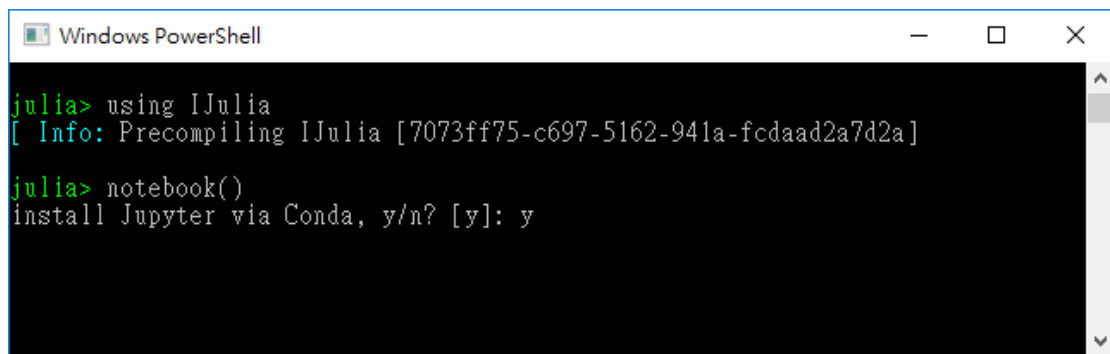
2. 安裝完成後，透過 using 命令使用 IJulia。呼叫 notebook() 啟動 Jupyter

Notebook。

```
julia> using IJulia
[ Info: Precompiling IJulia [7073ff75-c697-5162-941a-fcdaad2a7d2a]
julia> notebook()
install Jupyter via Conda, y/n? [y]:
```

在套件安裝完第一次使用時，套件都需要經過 *precompile* 的過程，需要花一些時間。但下一次使用時若版本相同、或是沒有重新 *build*，就不會再有 *precompile* 過程。

在這邊有一點要注意的地方是，第一次透過呼叫 `notebook()` 啟動 Jupyter Notebook 時，Julia 會詢問是否要透過 Conda 安裝 Jupyter，回答 `y` 的話，Conda 會在 Julia 環境下安裝新的 Jupyter。



```
Windows PowerShell
julia> using IJulia
[ Info: Precompiling IJulia [7073ff75-c697-5162-941a-fcdaad2a7d2a]
julia> notebook()
install Jupyter via Conda, y/n? [y]: y
```

呼叫 `notebook()` 時可指定路徑，將 Jupyter Notebook 在特定的路徑中啟動，語法為 `notebook(dir="<路徑>")`。

若要使用新一代的 Jupyter Lab 而不是 Jupyter Notebook 的話，呼叫 `jupyterlab()` 取代 `notebook()` 即可。

馬拉松課程內容是以上述在 Julia 隨附的 *mini conda* 下安裝

為範例。下面將將 *IJulia* 安裝在既有的 *Jupyter* 上的方式同

學們可以自行參考。

將 *IJulia* 安裝在既有的 *Jupyter* 上

在安裝 *IJulia* 之前，必須先指定環境變數到既有安裝的 *Jupyter* 路徑 (e.g.

D:\Python36\Scripts) 及執行檔。

```
ENV["JUPYTER"]="D:\\Python36\\Scripts\\jupyter.exe"
```

再執行安裝及開啟 *Jupyter Notebook*，就不需再安裝另一份 *Jupyter* 在

Julia 環境中。

```
julia> ENV["JUPYTER"]="D:\\Python36\\Scripts\\jupyter.exe"  
"D:\\Python36\\Scripts\\jupyter.exe"julia> using Pkg  
julia> Pkg.add("IJulia")  
julia> using IJulia  
julia> notebook()
```

Windows PowerShell

```
julia> using Pkg

julia> ENV["JUPYTER"]="D:\\Python36\\Scripts\\jupyter.exe"
"D:\\Python36\\Scripts\\jupyter.exe"

julia> Pkg.add("IJulia")
  Updating registry at `C:\Users\██████████\.julia\registries\General`
  Updating git-repo `https://github.com/JuliaRegistries/General.git`
  Resolving package versions...
  Updating `C:\Users\██████████\.julia\environments\v1.1\Project.toml`
 [7073ff75] + IJulia v1.18.0
  Updating `C:\Users\██████████\.julia\environments\v1.1\Manifest.toml`
 [7073ff75] + IJulia v1.18.0
 [b85f4697] + SoftGlobalScope v1.0.10
 [c2297ded] + ZMQ v1.0.0
 [7b1f6079] + FileWatching

julia> using IJulia
[ Info: Recompiling stale cache file C:\Users\██████████\.julia\compiled\
697-5162-941a-fcdaad2a7d2a]

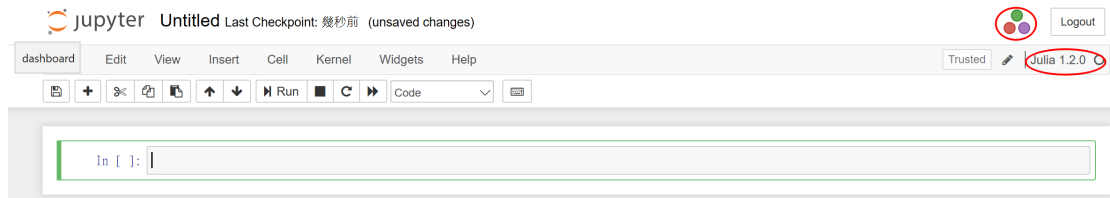
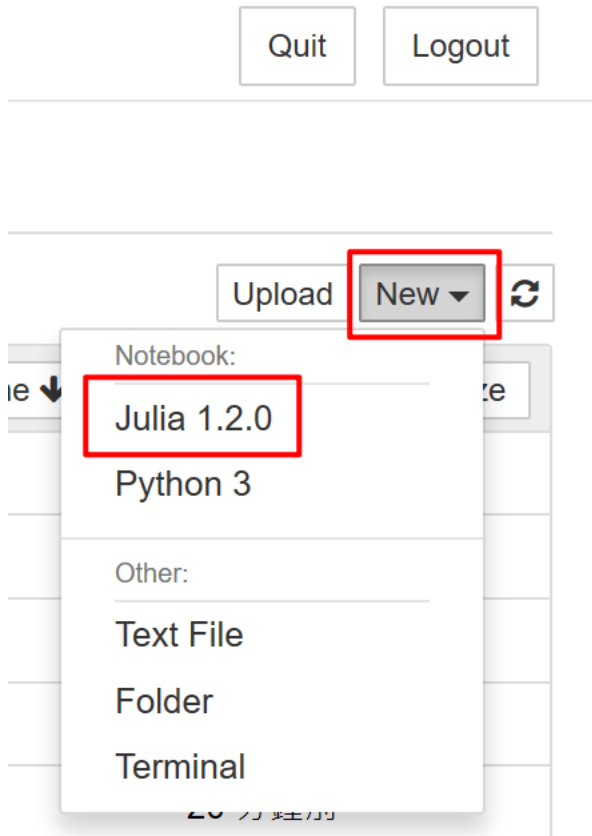
julia> notebook()
[ Info: running setenv(`D:/Python36/Scripts/jupyter.exe notebook`,`["PATH=D:/P
```

開啟 Jupyter 時可以不需要再透過 REPL 開啟，可直接在命令提示字元下執行

jupyter notebook 指令開啟。

開啟 Notebook 檔案開發 Julia

選擇要開啟的 Julia Kernel，新增一個新的 Notebook 檔案。



測試 Julia 正常執行。