**줄리아 학습 노트**

차례

들어가기

1. 준비
   1. 줄리아 설치
   2. 주피터 노트북 환경 설정(IJulia)
   3. 패키지
   4. 데이터 불러오기
2. 선형 대수(행렬 연산)
   1. 데이터 타입(스칼라, 벡터, 행렬, 어레이)
   2. 행렬 연산
   3. LU 분해
   4. QR 분해
   5. 촐스키 분해
   6. 희소 선형 대수(Sparse Linear Algebra)
   7. 특이값 분해(Singular Value Decomposition)
   8. 사진 예제
3. 기초 통계
   1. 기술 통계
   2. 통계 그래프(Statistics Plots)
   3. 커널 밀도 추정
   4. 확률 분포
   5. 가설 검정
   6. 상관계수
   7. AUC와 혼돈 행렬(Confusion Matrix)
4. 차원 축소
   1. PCA
   2. t-SNE
   3. UMAP
5. 클러스터링
   1. K-means clustering
   2. K-medoids clustering
   3. Hierarchical Clustering
   4. DBscan
6. 분류(Clustering)
   1. LASSO
   2. RIDGE
   3. Elastic Net
   4. Decision Trees
   5. Random Forests
   6. Nearest Neighbor method
   7. Support Vector Machines
7. 회귀(Regression)
   1. 선형 회귀(Linear Regression)
   2. 로지스틱 회귀(Logistic Regression)
   3. 비선형 회귀(Non-Linear Regression)
8. 그래프(Graphs)
   1. vlplot
   2. Shortest path problem
   3. Minimum Spanning Tree (MST)
   4. PageRank
   5. Clustering Coefficients
9. 수리적 최적화(Numerical Optimization)
   1. Portfolio investment problem
   2. Image Recovery
   3. Diet optimization problem
   4. Travel optimization
10. Neural Net (인공신경망)
11. 타 언어 불러오기
12. 시각화(Visualization)
    1. Symmetric violin plots and annotations
    2. Bar charts, histograms, and insets
    3. Plots with error bars
    4. Plots with double axes
    5. High-dimensional data in a 2D plot

**들어가기**

주피터 노트북을 만든 프로젝트 주피터의 이름이 주피터가 지원하는 세 개의 핵심 언어인 Julia, Python 그리고 R에서 유래했다고 한다.[[1]](#footnote-1) 이러한 유래를 알게 되면서 Julia라는 언어를 알게 되었고 어떤 언어인지 호기심이 생겼다. 하지만 매번 설치 오류와 환경 설정에서 알지 못하는 오류들로 인해 시작을 하지 못하였다. 그러던 어느날, 거짓말처럼 그동안 겪었던 오류들이 발생하지 않고 시원하게 원하는 패키지들을 설치하면서 줄리아를 본격적으로 실행해 볼 수 있게 되었다. 이제 나도 Jupyter의 일원이 될 수 있을 것만 같았다.(가자 목성으로!)

설레이는 기대감을 안고 줄리아에 관한 책을 찾아보니 현재 우리나라에는 두 권 정도 있는 것으로 나왔다. 하나는 2015년도에 발행된 에이콘 출판사의 ‘Julia 프로그래밍’이고 다른 하나는 2020년에 발행된 한빛미디어의 ‘Think Julia 줄리아를 생각하다’ 이었다. 둘 다 프로그래밍 언어를 소개하는 서적으로 무언가 줄리아의 응용(R, Python과 비교하면서 생각해 볼 수 있는 데이터 분석과 같은 분야)분야에 관한 자료는 어디 없을까 하고 찾다보니 유튜브에 휴다 나사르 박사님의 오픈 강의가 있었다. 박사님의 강의와 자료들 보면서 공부를 시작해 볼 수 있었다.[[2]](#footnote-2)

Julia for Data Science - Video 1: Data, by Dr. Huda Nassar (for JuliaAcademy.com)

Julia for Data Science - Video 2: Linear Algebra, by Dr. Huda Nassar (for JuliaAcademy.com)

Julia for Data Science - Video 3: Statistics, by Dr. Huda Nassar (for JuliaAcademy.com)

Julia for Data Science - Video 4: Dimensionality Reduction, by Dr. Huda Nassar (JuliaAcademy.com)

Julia for Data Science - Video 5: Clustering, by Dr. Huda Nassar (for JuliaAcademy.com)

Julia for Data Science - Video 6: Classification, by Dr. Huda Nassar (for JuliaAcademy.com)

Julia for Data Science - Video 7: Regression, by Dr. Huda Nassar (for JuliaAcademy.com)

Julia for Data Science - Video 8: Graphs, by Dr. Huda Nassar (for JuliaAcademy.com)

Julia for Data Science - Video 9: Optimization, by Dr. Huda Nassar (for JuliaAcademy.com)

Julia for Data Science - Video 10: Neural Nets, by Dr. Huda Nassar (for JuliaAcademy.com)

Julia for Data Science - Video 11: Other Languages, by Dr. Huda Nassar (for JuliaAcademy.com)

Julia for Data Science - Video 12: Plots, by Dr. Huda Nassar (for JuliaAcademy.com)

강의 주소: https://juliaacademy.com/p/julia-for-data-science (무료 강의로 오픈되어 있다.)

1. 준비
   1. 줄리아 설치

줄리아 설치파일을 <https://julialang.org/> 에서 다운받아 실행하면 간편하게 설치가 완료된다. 다만 주의할 점은 이후에 줄리아를 사용할 환경을 고려하여 사용자 권한으로 설치할 지, 설치 폴더는 어디에 할 지 정할 필요는 있다. 설치가 완료되어 실행하면 까만 창에 Julia 로고가 나오는 새로운 창이 뜰 것이다. 줄리아의 커맨드 라인인 REPL(Read-Evaluation-Print-Loop)이다. 마치 R의 Gui 콘솔 또는 파이썬의 IDLE과 비슷하다.

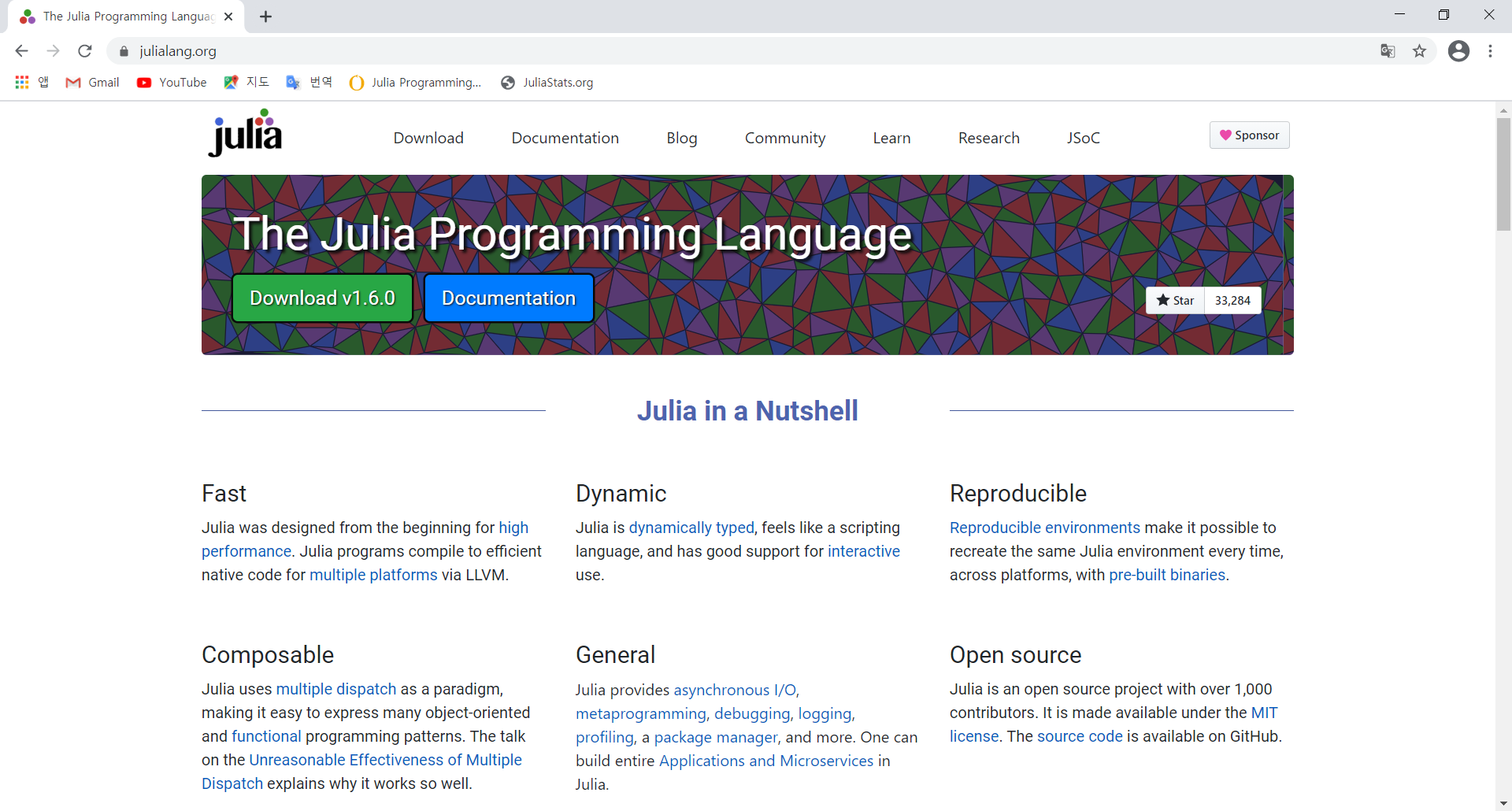


그림 줄리아 프로젝트 홈페이지 모습

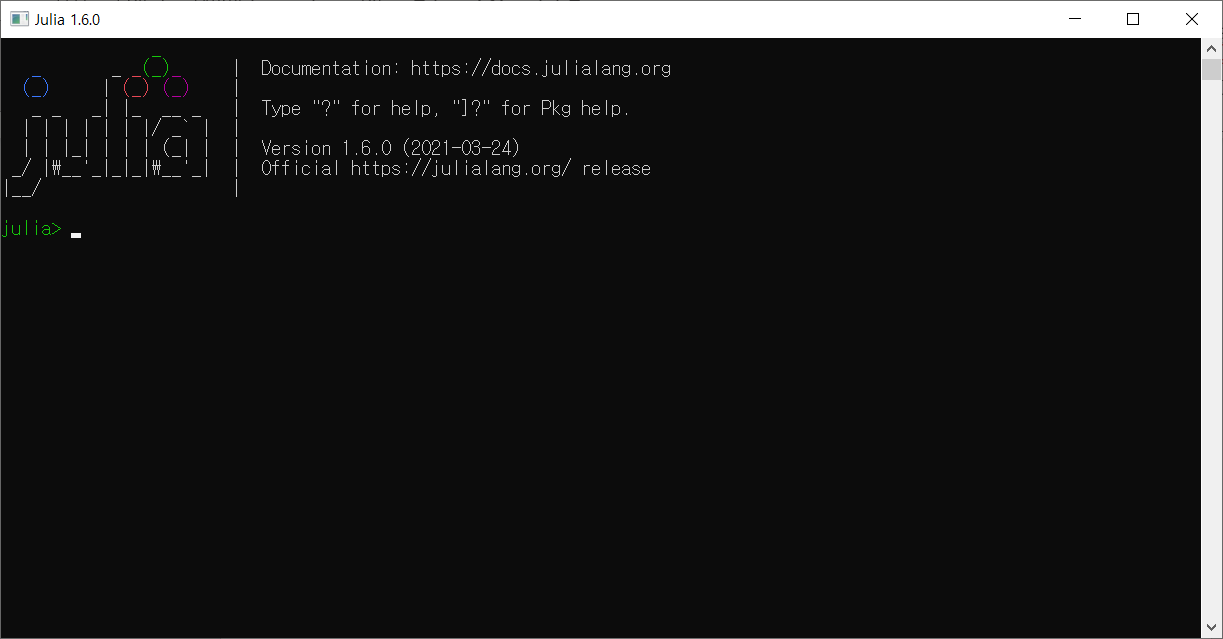


그림 줄리아 REPL 모습

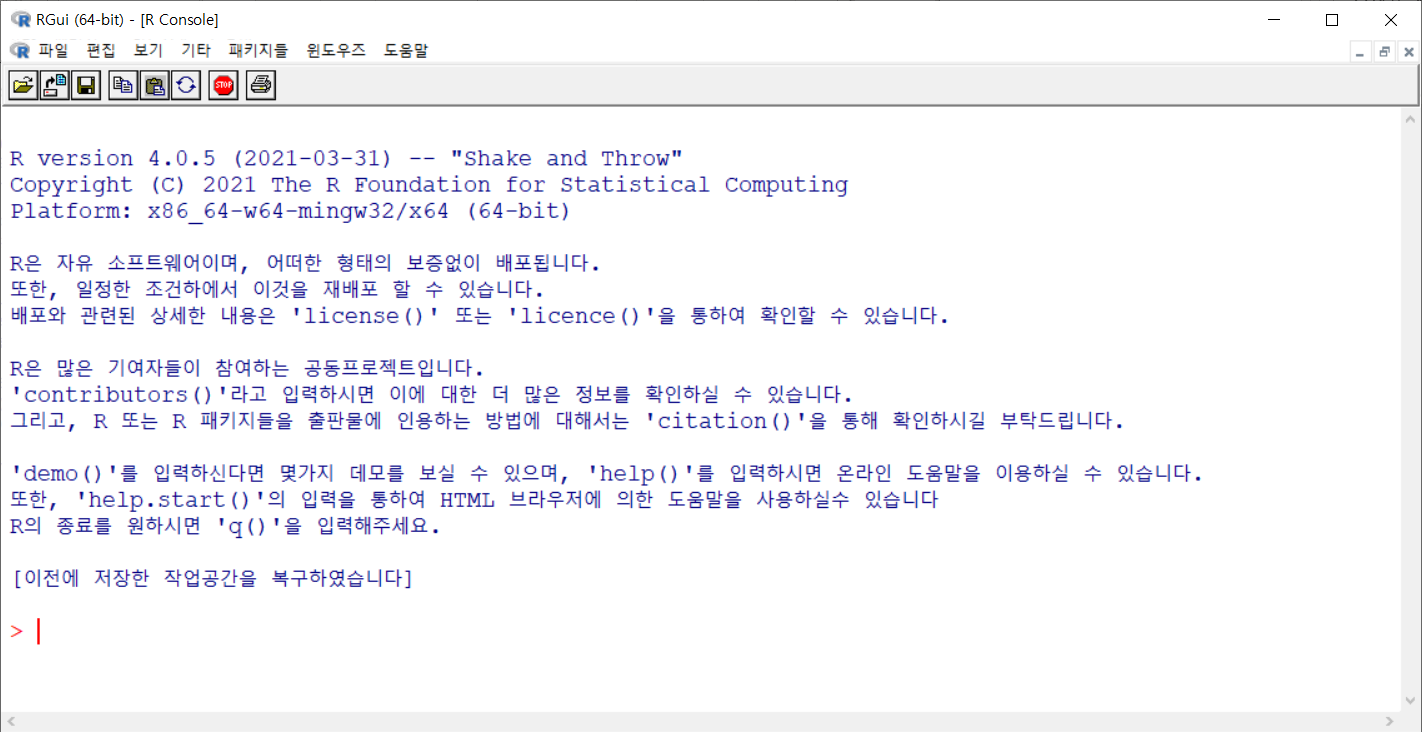


그림 R GUI 콘솔 모습

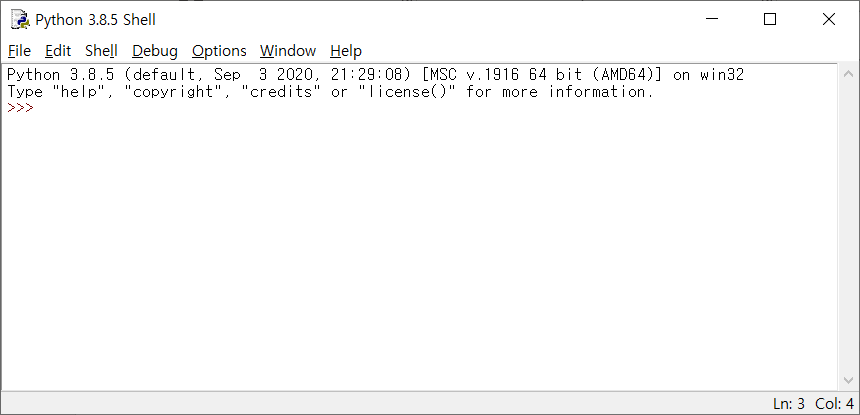
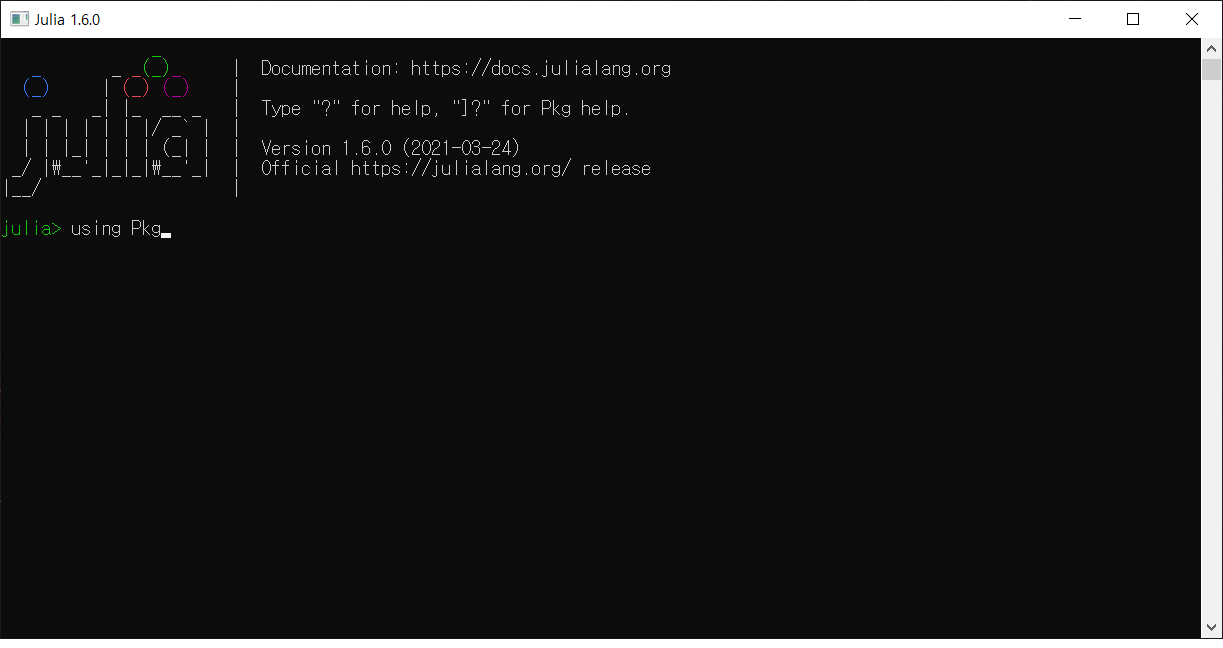


그림 파이썬 Idle shell 모습

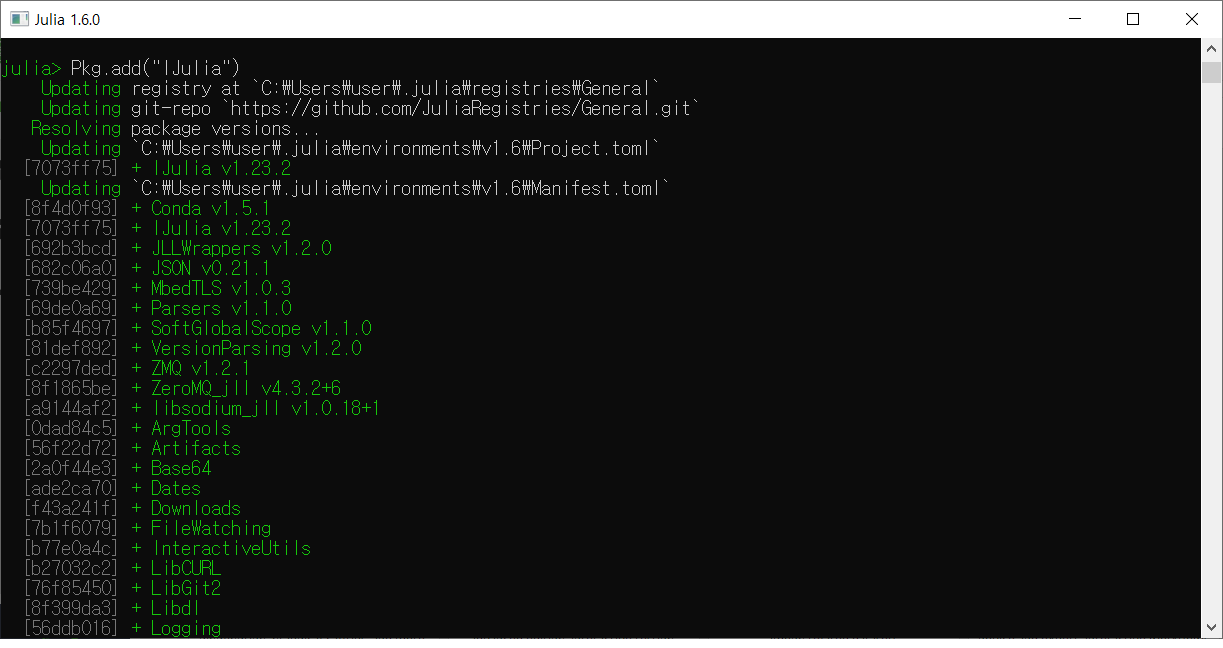
* 1. 주피터 노트북 환경 설정(IJulia)

줄리아 설치 후 구동할 수 있는 REPL도 좋은 툴이지만 조금 더 편하고 익숙한 툴을 사용해 보고자 한다. 바로 주피터 노트북이다. 줄리아를 주피터 노트북에 임포트하려면 간단한 명령어로 실행이 가능하다.[[3]](#footnote-3)

우선 REPL을 켜고 줄리아의 내장 패키지 매니저인 Pkg 패키지를 불러온다. 줄리아에서 패키지를 불러올 때는 using이라는 명령어를 사용한다.

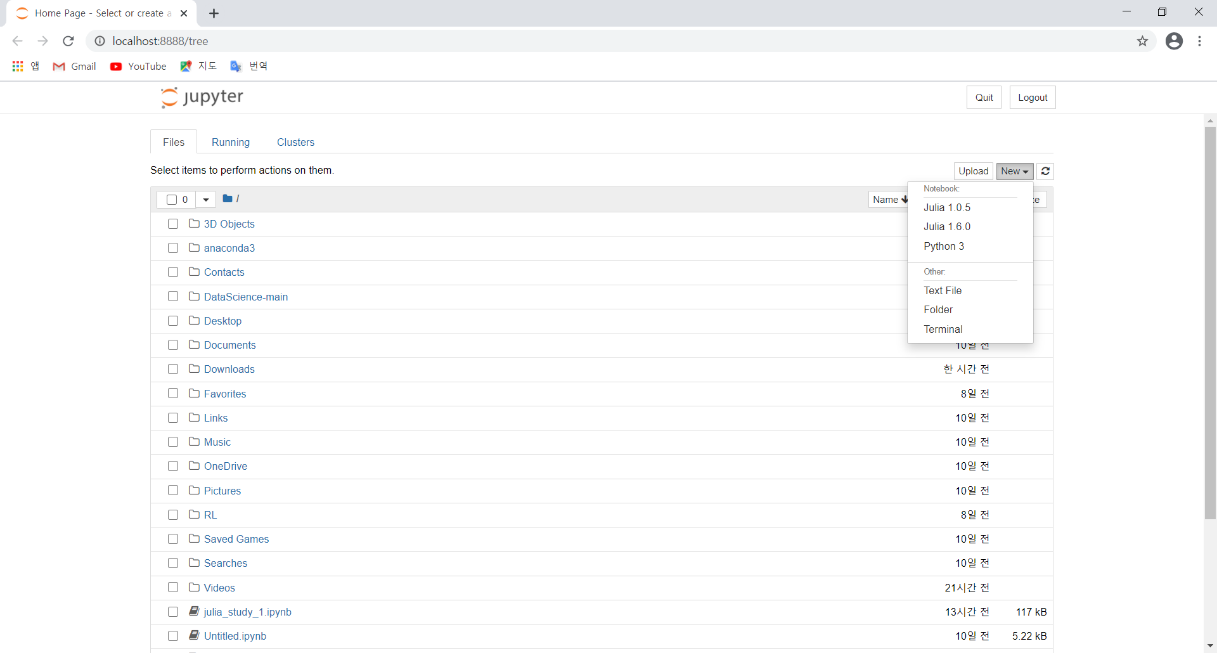


Pkg 패키지를 불러온 뒤, Pkg의 add 함수를 이용하여 “IJulia”라는 패키지를 다운받는다.



이는 마치 R에서 install.packages(“패키지 이름”) 명령어나 python에서 pip install 패키지 명령어를 사용하는 것과 비슷한 것 같다.

그 후 주피터 노트북을 켜면 오른쪽 상단에 new를 클릭하면 notebook이 사용 가능한 커널로 julia와 python3가 나오는 것을 볼 수 있다. 자 이제 jupyter notebook을 켜면 아래와 같은 화면이 나타나는 것을 볼 수 있다.



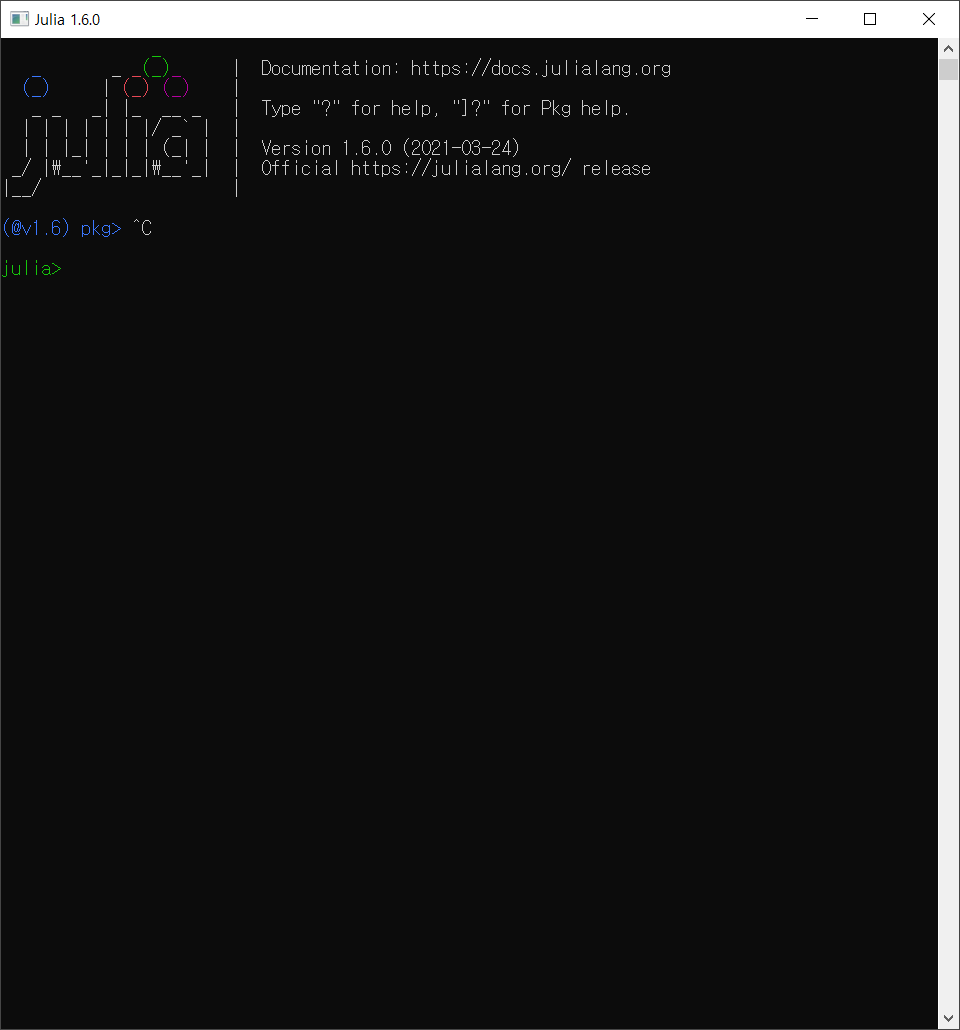
만약 여러 개 버전의 줄리아를 설치하여 사용하고 싶다면 새로운 버전의 줄리아의REPL에서 Pkg.add(“IJulia”) 명령어 실행 이후 Pkg.build(“IJulia”) 명령어를 한 번 더 실행해 주면 깔끔하게 주피터 노트북에 올라오는 것을 확인할 수 있다.

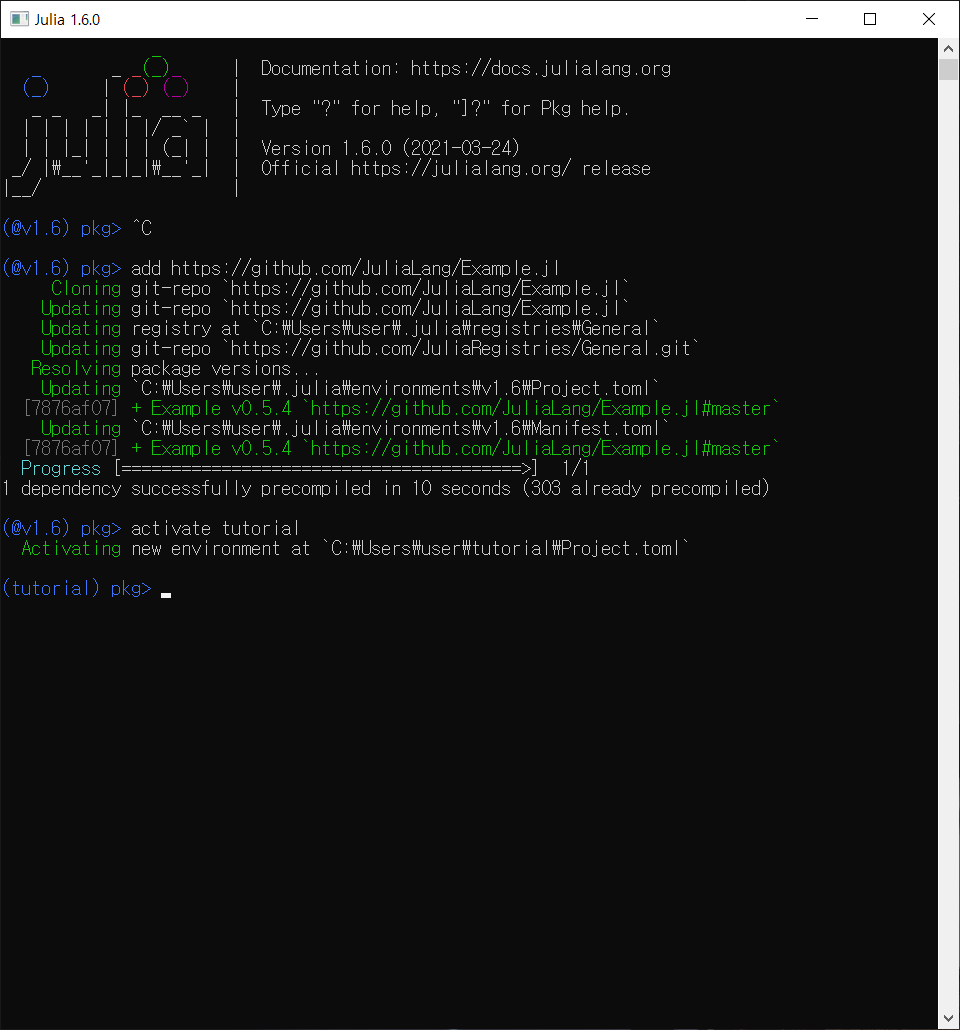
jupyter 3단 비교표

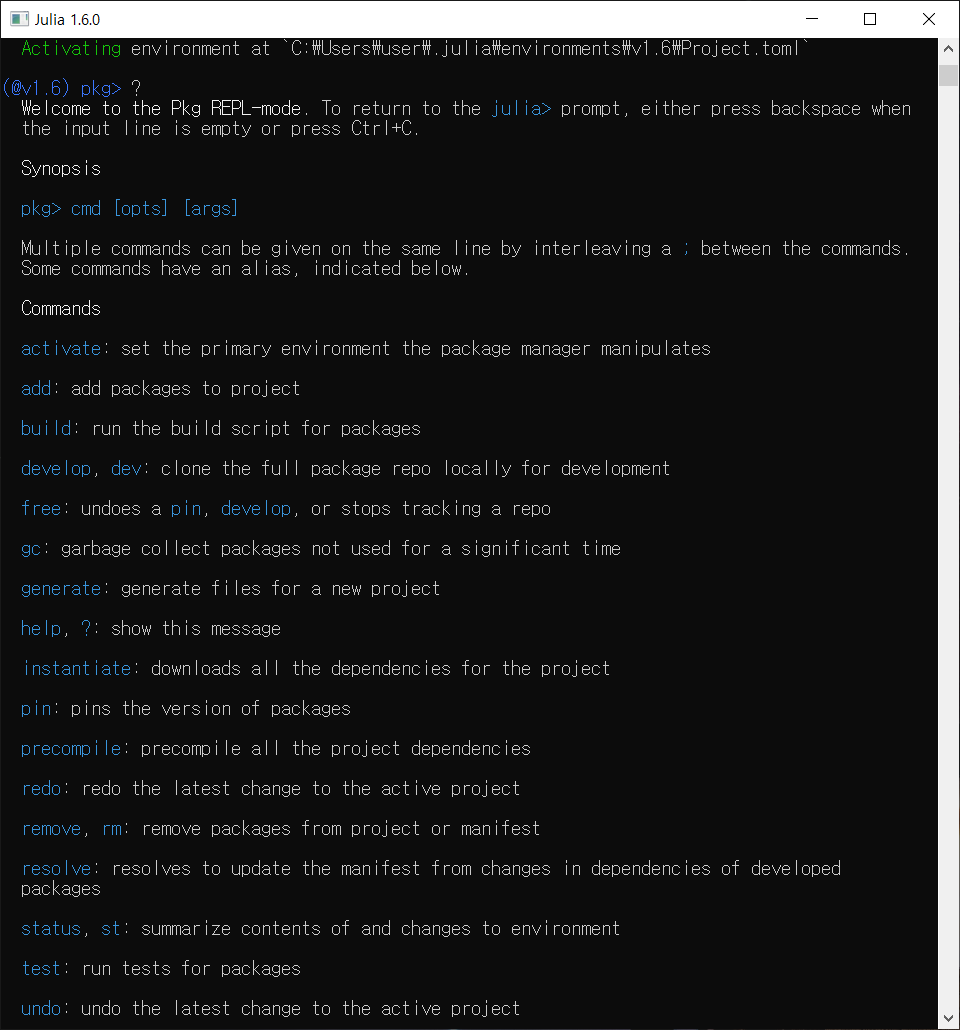
|  |  |  |  |
| --- | --- | --- | --- |
|  | **julia** | **python** | **R** |
| 패키지 불러오기 | using 패키지명 | import 패키지명 | library(패키지명) |
| using Pkg | import os | library(tidyverse) |
| 패키지 설치 | Pkg.add(“패키지명”) | pip install 패키지명 | install.package(패키지명) |

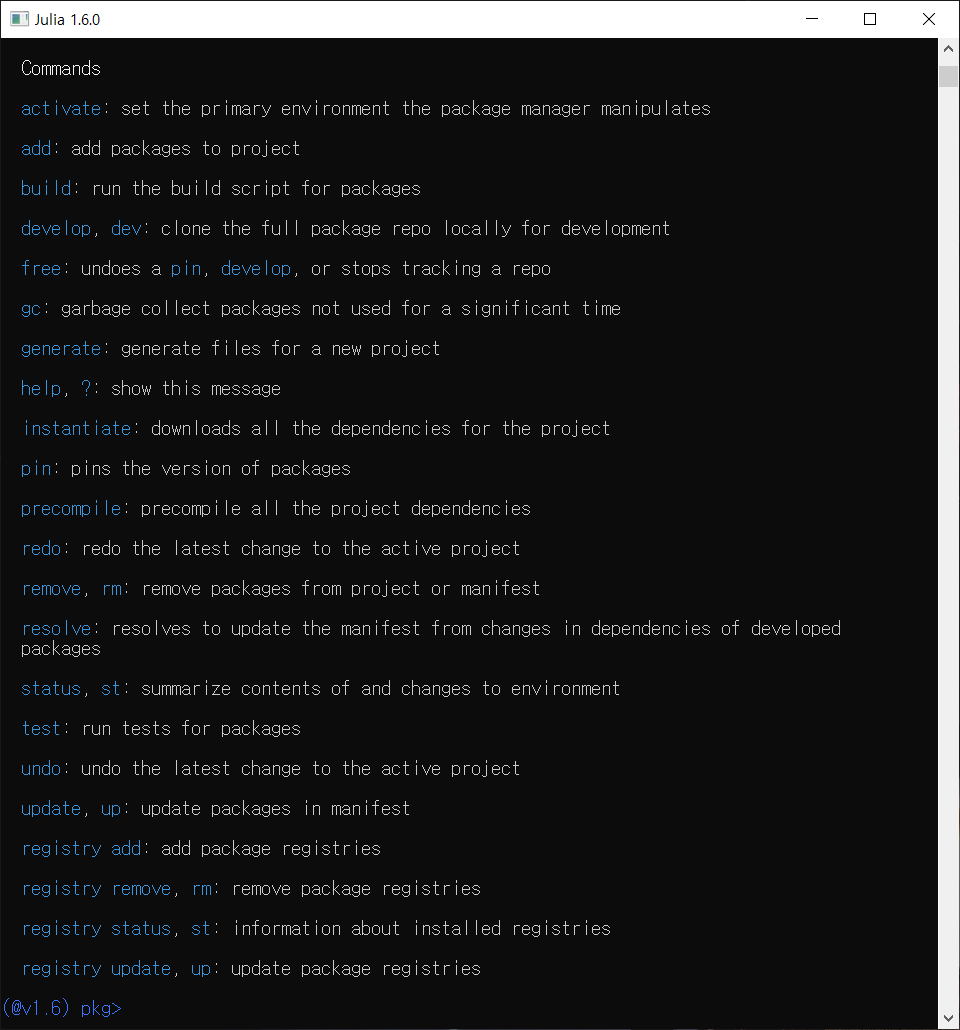
* 1. 패키지

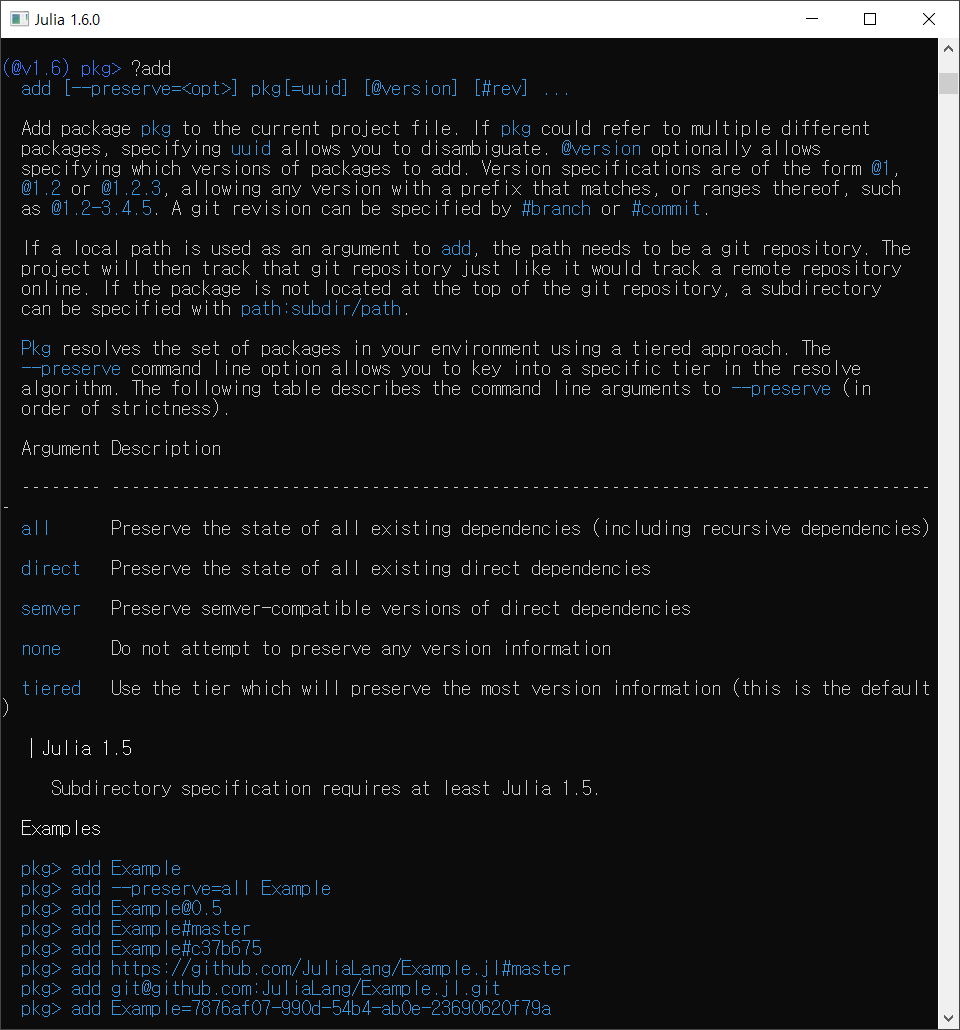
REPL에서 Pkg 모드 사용 (https://docs.julialang.org/en/v1/stdlib/Pkg/)







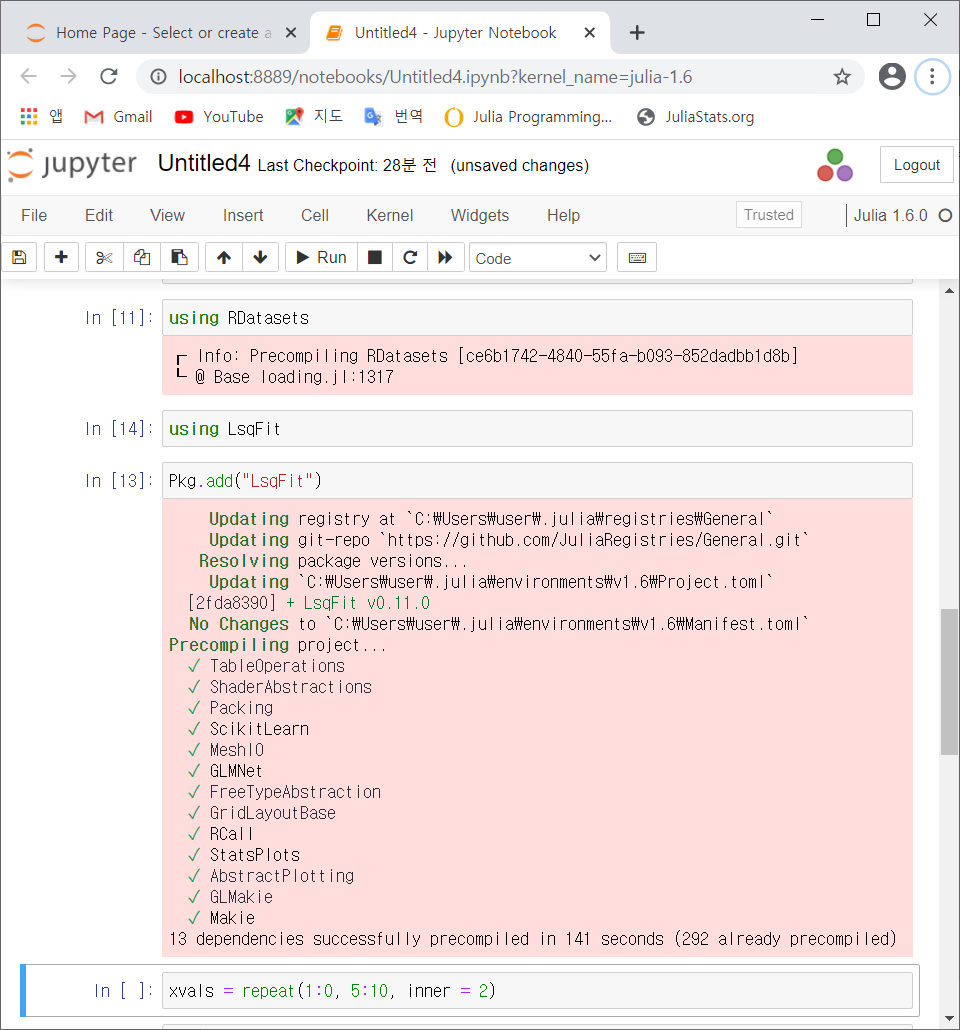




using과 import 차이점

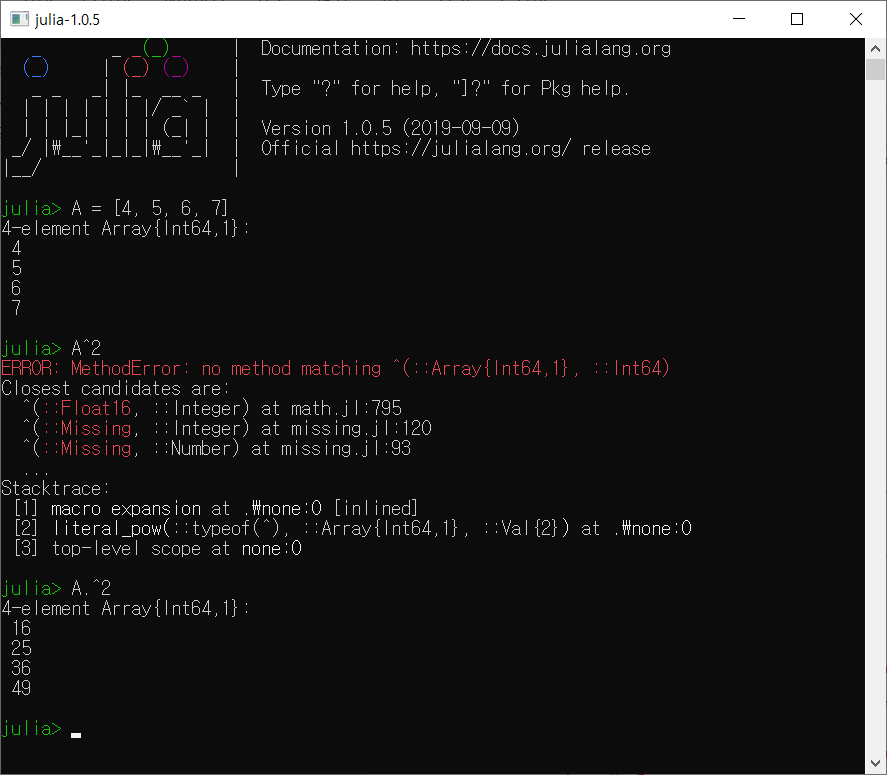
https://docs.julialang.org/en/v1/manual/modules/#modules

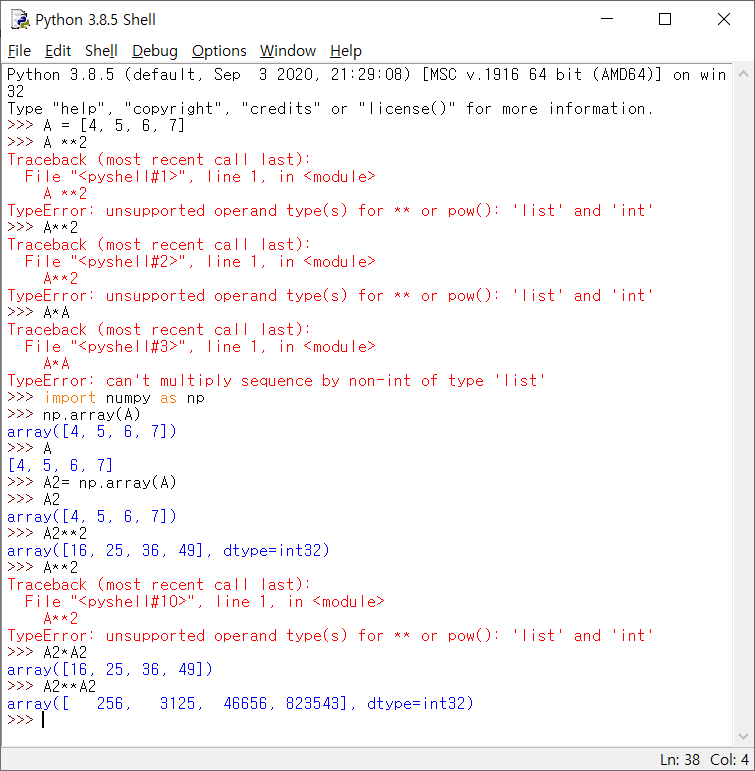
https://docs.julialang.org/en/v1/manual/code-loading/



Package에서 precompiling의 의미는 무엇일까?

* 1. 데이터 불러오기







* 1. 줄리아 연산자 확인하기

https://docs.julialang.org/en/v1/manual/mathematical-operations/

1. 선형 대수(행렬 연산)
   1. 데이터 타입(스칼라, 벡터, 행렬, 어레이)
   2. 행렬 연산
   3. LU 분해
   4. QR 분해
   5. 촐스키 분해
   6. 희소 선형 대수(Sparse Linear Algebra)
   7. 특이값 분해(Singular Value Decomposition)
   8. 사진 예제
2. 기초 통계
   1. 기술 통계
   2. 통계 그래프(Statistics Plots)
   3. 커널 밀도 추정
   4. 확률 분포
   5. 가설 검정
   6. 상관계수
   7. AUC와 혼돈 행렬(Confusion Matrix)
3. 차원 축소
   1. PCA
   2. t-SNE
   3. UMAP
4. 클러스터링
   1. K-means clustering
   2. K-medoids clustering
   3. Hierarchical Clustering
   4. DBscan
5. 분류(Clustering)
   1. LASSO
   2. RIDGE
   3. Elastic Net
   4. Decision Trees
   5. Random Forests
   6. Nearest Neighbor method
   7. Support Vector Machines
6. 회귀(Regression)
   1. 선형 회귀(Linear Regression)
   2. 로지스틱 회귀(Logistic Regression)
   3. 비선형 회귀(Non-Linear Regression)
7. 그래프(Graphs)
   1. vlplot
   2. Shortest path problem
   3. Minimum Spanning Tree (MST)
   4. PageRank
   5. Clustering Coefficients
8. 수리적 최적화(Numerical Optimization)
   1. Portfolio investment problem
   2. Image Recovery
   3. Diet optimization problem
   4. Travel optimization
9. Neural Net (인공신경망)
10. 타 언어 불러오기
11. 시각화(Visualization)
    1. Symmetric violin plots and annotations
    2. Bar charts, histograms, and insets
    3. Plots with error bars
    4. Plots with double axes
    5. High-dimensional data in a 2D plot

1. <https://ko.wikipedia.org/wiki/%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_%EC%A3%BC%ED%94%BC%ED%84%B0> [↑](#footnote-ref-1)
2. <https://github.com/JuliaAcademy/DataScience> (강의자료는 github에서 다운 받을 수 있다.) [↑](#footnote-ref-2)
3. <https://datatofish.com/add-julia-to-jupyter/> (How to Add Julia to Jupyter Notebook) [↑](#footnote-ref-3)