

45.(심화)Jupyter Notebook

2019.06

상병김재형

IPython IP[y]: IPython Interactive Computing

IPython이란?

- 상호작용 인터프리터(interactive interpreter)
- 목적
 - 코드를 실행했을 때, 결과를 더 쉽게 볼 수 있도록 제작
 - 많은 데이터 처리 시,
결과를 바로 보아 원하는 답을 쉽게 찾게 설계

IPython IP[y]: IPython Interactive Computing

IPython 특징

- tab 키를 통해 자동 완성
- system shell(bash 등)에 접근해 system 명령어 사용
- 현재 실행한 명령어 목록 기록
- IPython 자체만의 명령어 제공

IPython과 Jupyter Notebook

역사

- IPython 프로젝트 내 IPython notebook 생성
- R, Ruby 등의 언어도 사용할 필요가 있음
- Project Jupyter로 파생하여 지속





Project Jupyter

Jupyter notebook과 Jupyterhub

- Jupyter Notebook
 - 대화형 기술 문서를 작성하기 위한 제품
 - 혼자서 사용
- Jupyterhub
 - 여러 사람이 사용할 수 있도록 만든 Jupyter Notebook
 - 서버기반 배포

Jupyter Notebook

Jupyter Notebook의 특징

- IPython의 특징을 그대로 가짐
- literate programming(문학적 프로그래밍)이 가능
 - 사람이 이해하기 쉬운 코드를 만듦
 - 마크다운, html, Javascript, LaTeX 언어로 설명 추가 가능

기본정보 확인

In [2]:

```
# 실행중인 운영체제 확인

import platform
platform.platform()
```

Out[2]:

```
'Linux-4.9.17-c9-x86_64-with-Ubuntu-14.04-trusty'
```

Jupyter Notebook

Jupyter Notebook의 특징

- Jupyter 서버를 만들어
인터넷에서 Jupyter Notebook으로 접속 가능

Jupyter Notebook-설치

pip

—pip install jupyter

```
C:\Users\WKJH>pip install jupyter
Collecting jupyter
  Downloading https://files.pythonhosted.org/packages/83/df/0f5dd132200728a86190
397e1ea87cd76244e42d39ec5e88efd25b2abd7e/jupyter-1.0.0-py2.py3-none-any.whl
Collecting jupyter-console (from jupyter)
  Downloading https://files.pythonhosted.org/packages/cb/ee/6374ae8c21b7d0847f9c
3722dcdfac986b8e54fa9ad9ea66e1eb6320d2b8/jupyter_console-6.0.0-py2.py3-none-any.
whl
Collecting nbconvert (from jupyter)
  Downloading https://files.pythonhosted.org/packages/35/e7/f46c9d65f149271e47fc
a6ab084ef5c6e4cb1870f4c5cce6690feac55231/nbconvert-5.5.0-py2.py3-none-any.whl (4
47kB)
  |████████████████████████████████████████| 450kB 819kB/s
Collecting ipywidgets (from jupyter)
  Downloading https://files.pythonhosted.org/packages/30/9a/a008c7b1183fac9e5206
6d80a379b3c64eab535bd9d86cdc29a0b766fd82/ipywidgets-7.4.2-py2.py3-none-any.whl (
111kB)
  |████████████████████████████████████████| 112kB 2.2MB/s
Collecting qtconsole (from jupyter)
```


Jupyter Notebook-설치

Anaconda

- Python의 데이터분석 관련 라이브러리 모음
- Numpy, Matplotlib, Bokeh 등

Jupyter Notebook-설치

Anaconda

—Anaconda 홈페이지

<https://www.anaconda.com/distribution/>

Anaconda 2019.03 for Windows Installer

Python 3.7 version

Download

64-Bit Graphical Installer (662 MB)

32-Bit Graphical Installer (546 MB)

Python 2.7 version

Download

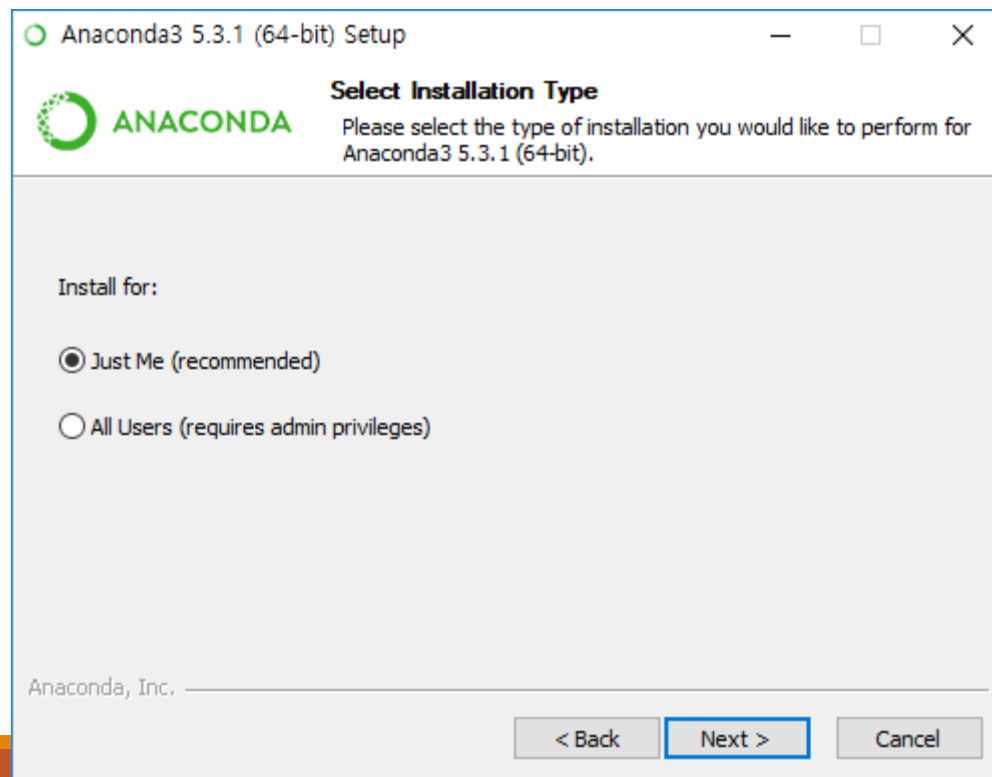
64-Bit Graphical Installer (587 MB)

32-Bit Graphical Installer (493 MB)

Jupyter Notebook-설치

Anaconda

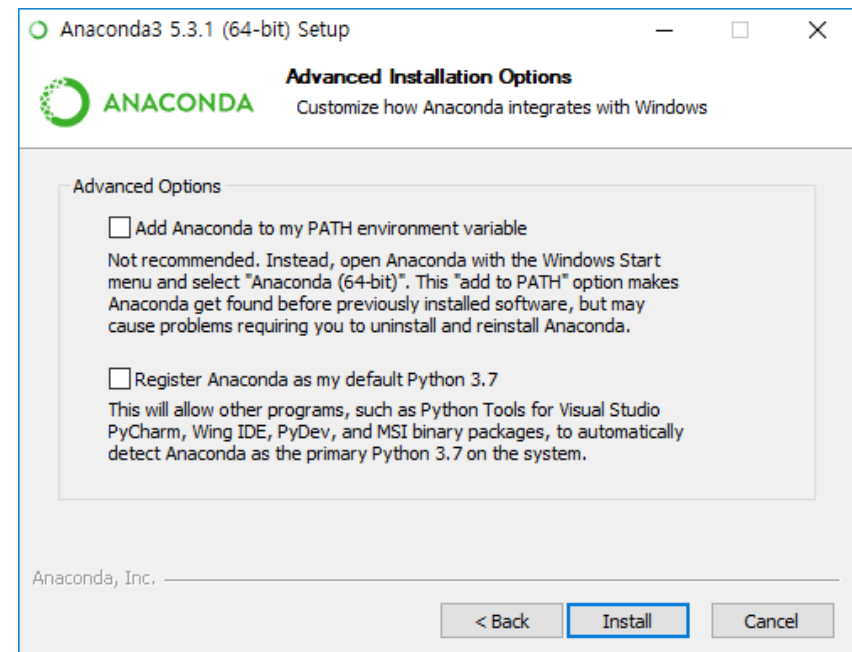
- 초기 설정은 변경 필요 없음



Jupyter Notebook-설치

Anaconda

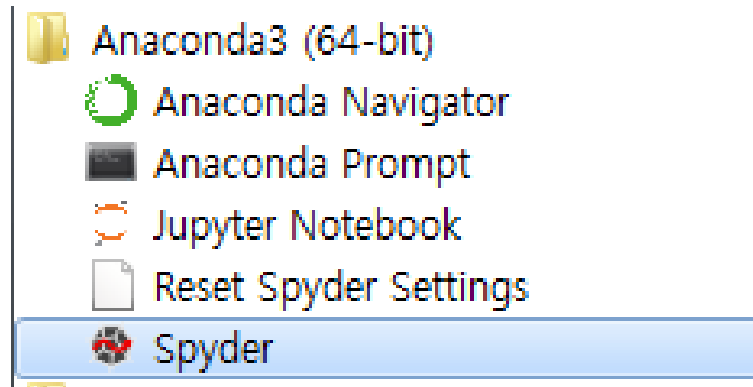
- 아나콘다를 환경 변수에 추가
 - 윈도우 환경변수
- 아나콘다를 파이썬 3.7 기본으로 등록
 - Python을 이용하는 어플리케이션이 아나콘다를 기본으로 사용



Jupyter Notebook-설치

Anaconda

- Anaconda Prompt를 사용한다.
- Jupyter Notebook를 누르면 Jupyter Notebook이 실행



Jupyter Notebook

실행

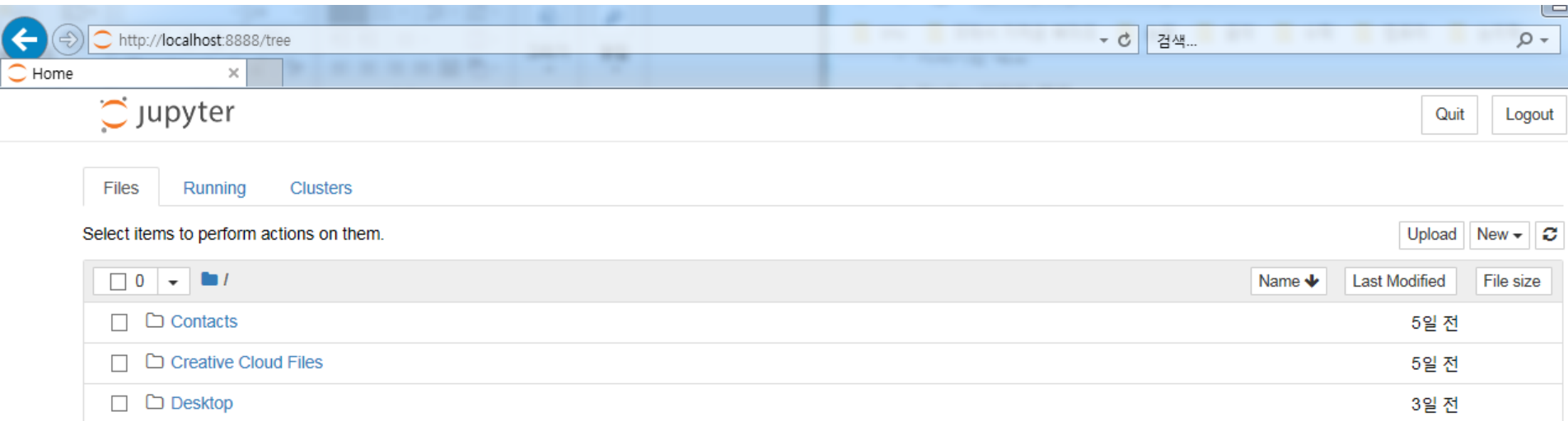
—cmd-jupyter notebook

```
C:\Users\WKJH>jupyter notebook
[I 07:58:32.061 NotebookApp] Writing notebook server cookie secret to C:\Users\WKJH\AppData\Roaming\jupyter\runtime\notebook_cookie_secret
[I 07:58:34.684 NotebookApp] Serving notebooks from local directory: C:\Users\WKJH
```

Jupyter Notebook

실행

—이후 기본 웹브라우저로 실행 (8888포트)



Jupyter Notebook

실행

- cmd창의 아래에 나타나는 주소 입력
=> 토큰도 입력

```
[C 07:58:34.871 NotebookApp]
```

```
To access the notebook, open this file in a browser:
```

```
file:///C:/Users/KJH/AppData/Roaming/jupyter/runtime/nbserver-37532-open  
.html
```

```
Or copy and paste one of these URLs:
```

```
http://localhost:8888/?token=1f11bd68efac7927d1b81eedb540196e1792c0d17e4  
4aech
```


Jupyter Notebook

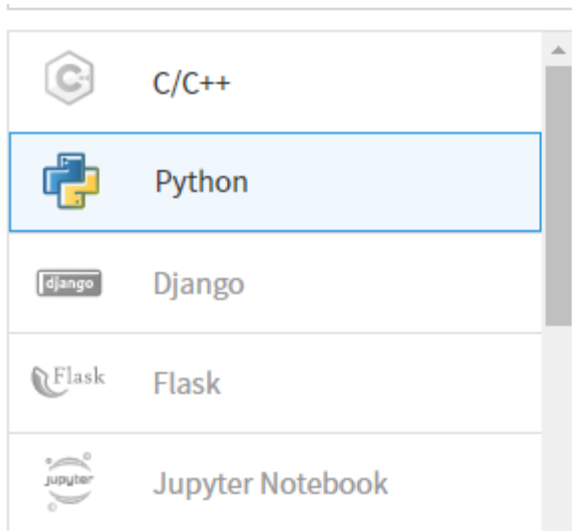
종료

- 웹 브라우저를 닫고, cmd창을 종료

Jupyter Notebook

구름 IDE

- python container에는 설치가 되어 있음
- 하위 Jupyter Notebook으로 만들면, 명령어가 존재하는 차이점이 존재



프로젝트 유형	
Python 프로젝트	
OS	✓ Ubuntu 14.04 LTS
Python3	✓ 3.6.5
Python	✓ 2.7.6
pip3	✓ 19.0.3
pip2	✓ 19.0.3

Jupyter Notebook

구름 IDE

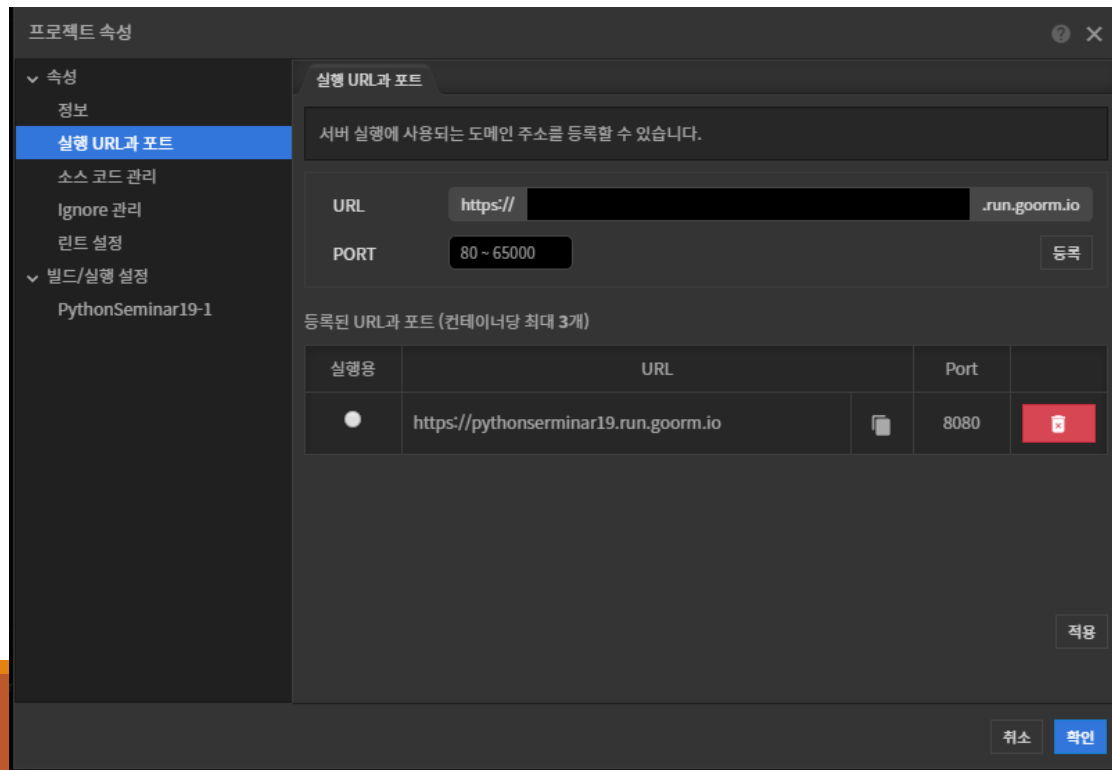
- 접속 가능하도록 실행 'URL과 포트'를 제공
- 프로젝트-속성 클릭



Jupyter Notebook

구름 IDE

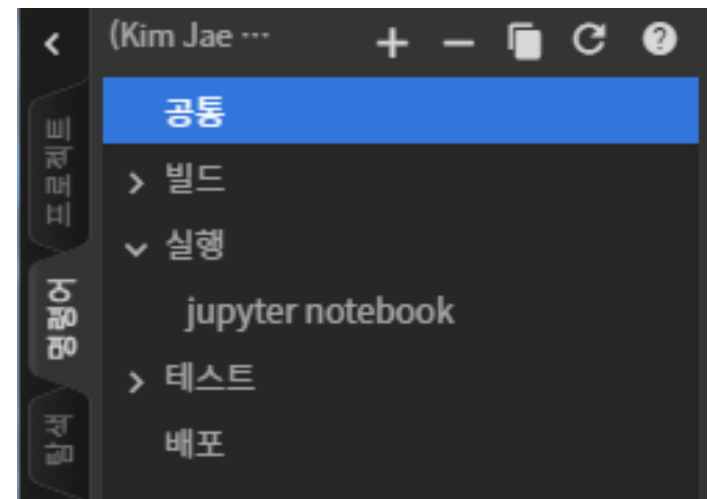
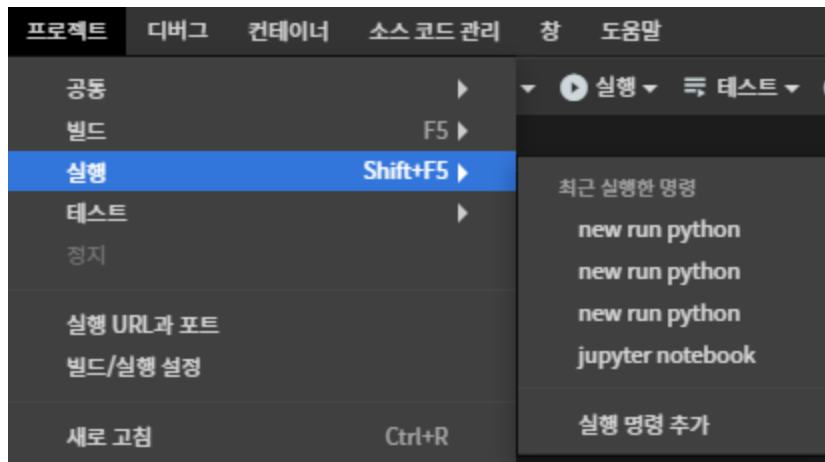
- 프로젝트 속성-실행 URL과 포트
- 설정이 되어 있으므로, 실행용 라디오버튼 클릭



Jupyter Notebook

구름 IDE

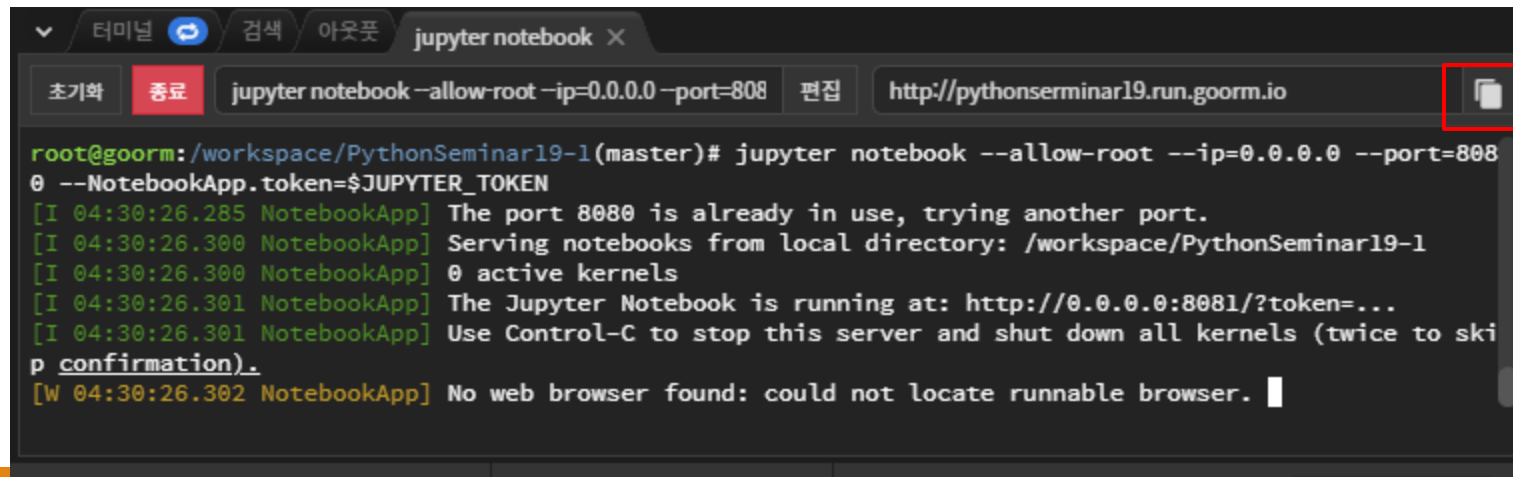
- 명령어 실행
- 프로젝트-실행-최근 실행한 명령-'jupyter notebook'
- 왼쪽 명령어탭-실행-jupyter notebook



Jupyter Notebook

구름IDE

- 실행시키면 아래쪽에 새로운 창으로 명령어 실행
- 설정은 완료되어 있음
- URL을 복사하여 웹브라우저에 붙여넣기

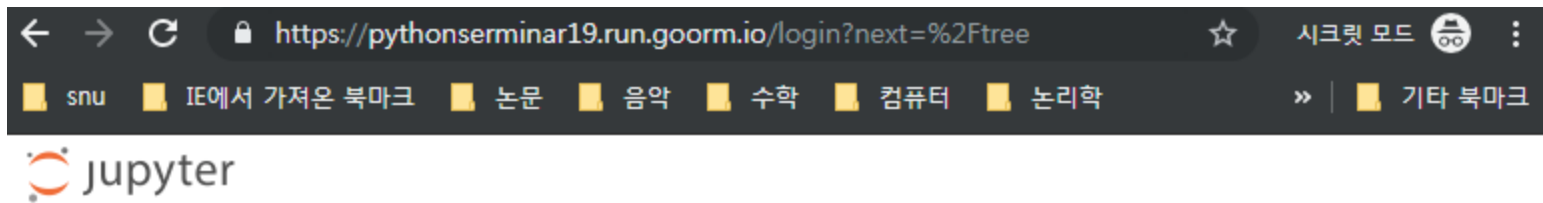


```
root@goorm:/workspace/PythonSeminar19-1(master)# jupyter notebook --allow-root --ip=0.0.0.0 --port=8080
--NotebookApp.token=$JUPYTER_TOKEN
[I 04:30:26.285 NotebookApp] The port 8080 is already in use, trying another port.
[I 04:30:26.300 NotebookApp] Serving notebooks from local directory: /workspace/PythonSeminar19-1
[I 04:30:26.300 NotebookApp] 0 active kernels
[I 04:30:26.301 NotebookApp] The Jupyter Notebook is running at: http://0.0.0.0:8081/?token=...
[I 04:30:26.301 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip
confirmation).
[W 04:30:26.302 NotebookApp] No web browser found: could not locate runnable browser.
```

Jupyter Notebook

구름IDE

—토큰(pythonseminar)입력



Password or token:

Log in

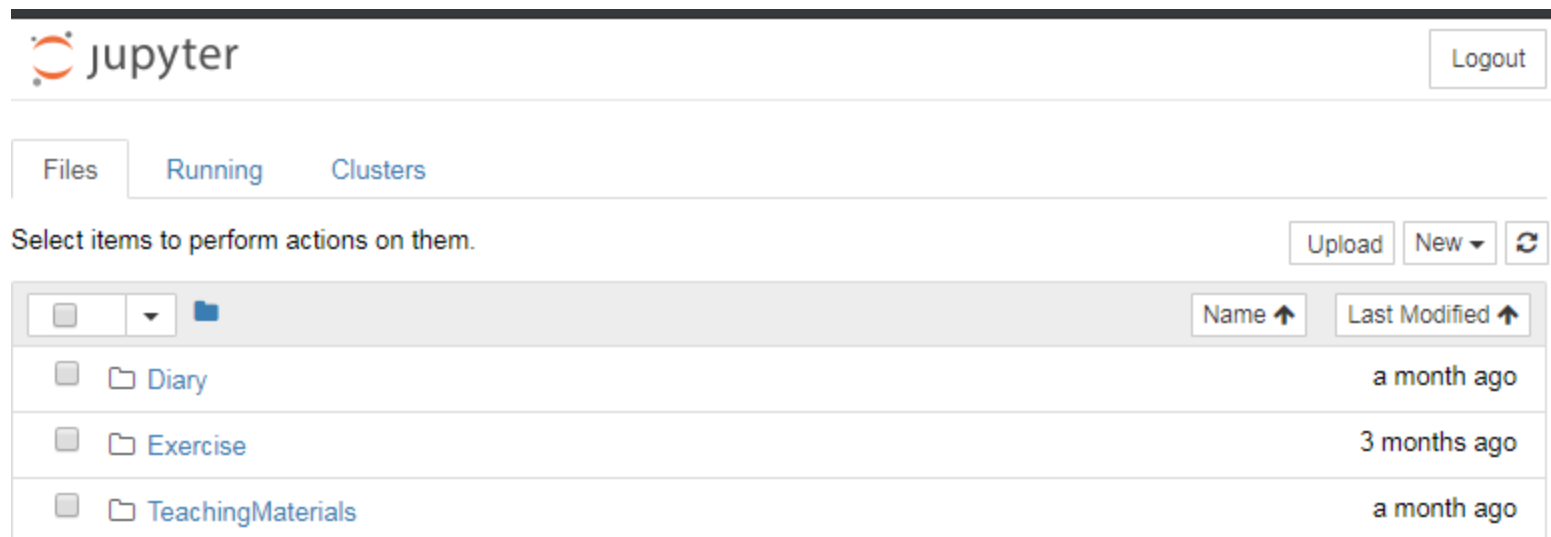
Token authentication is enabled

If no password has been configured, you need to open the notebook server with its login token in the URL, or paste it above. This requirement will be lifted if you [enable a password](#).

Jupyter Notebook

구름IDE

—접속이 완료



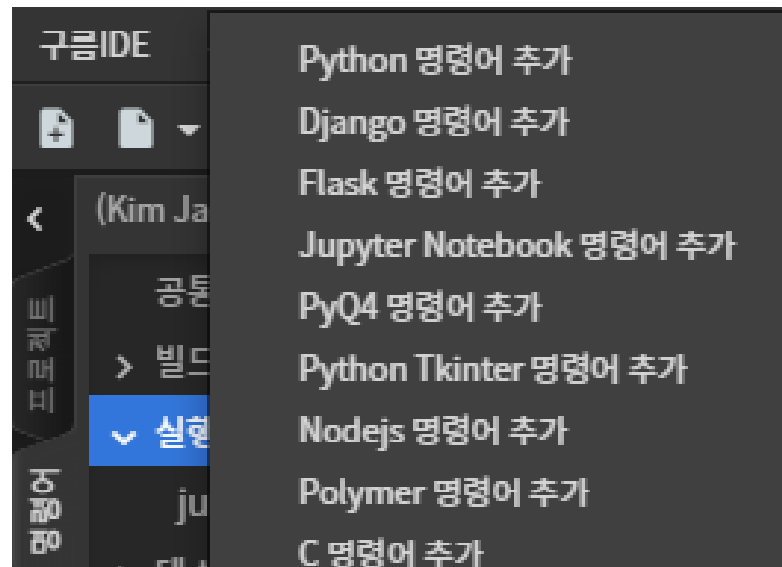
The screenshot shows the Jupyter Notebook web interface. At the top, there is a "jupyter" logo and a "Logout" button. Below this, there are three tabs: "Files", "Running", and "Clusters". The "Files" tab is active. Below the tabs, there is a message "Select items to perform actions on them." and three buttons: "Upload", "New", and a refresh icon. Below this, there is a table of files and folders. The table has two columns: "Name" and "Last Modified". The files listed are "Diary", "Exercise", and "TeachingMaterials".

Name	Last Modified
Diary	a month ago
Exercise	3 months ago
TeachingMaterials	a month ago

Jupyter Notebook

구름IDE

- 명령어 새로 만들기
 - 명령어 탭에서 오른쪽 버튼을 클릭하면 템플릿 존재



Jupyter Notebook

구름IDE

— 새로 만들기

— 클릭하면 새로운 명령어 이름/스크립트 제작 가능

— 아래 환경변수의 JUPYTER_TOKEN의 값을 바꾸면 토큰값 변경

The screenshot shows the Goorm IDE interface. At the top, there's a section titled '이름' (Name) with a button labeled 'new run jupyter notebook'. Below this is a toggle switch for '서버 프로세스' (Server Process), which is currently turned on. Underneath is a section titled '스크립트' (Script) with a code editor containing the following command:

```
1 jupyter notebook --allow-root --ip=0.0.0.0 --port=${current.using.port} --  
NotebookApp.token=$JUPYTER_TOKEN
```

At the bottom, there's a section titled '환경변수' (Environment Variables) with a table:

이름	값
JUPYTER_TOKEN	goorm

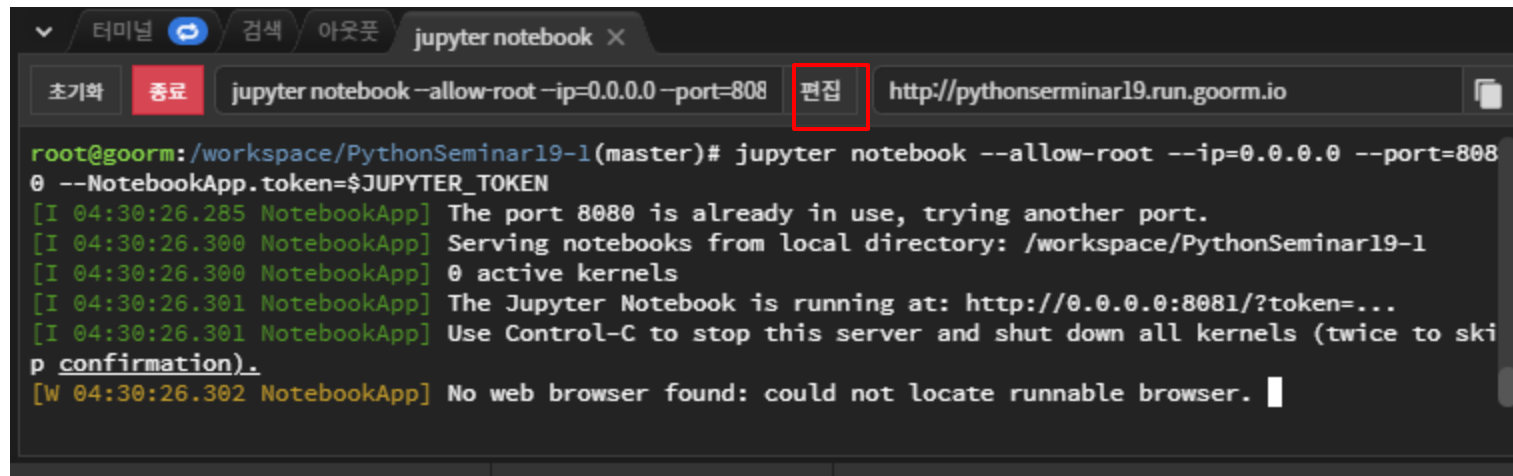
There is a red trash icon next to the 'goorm' value in the table.

Jupyter Notebook

구름IDE

— 편집

— 실행된 명령어의 편집을 클릭



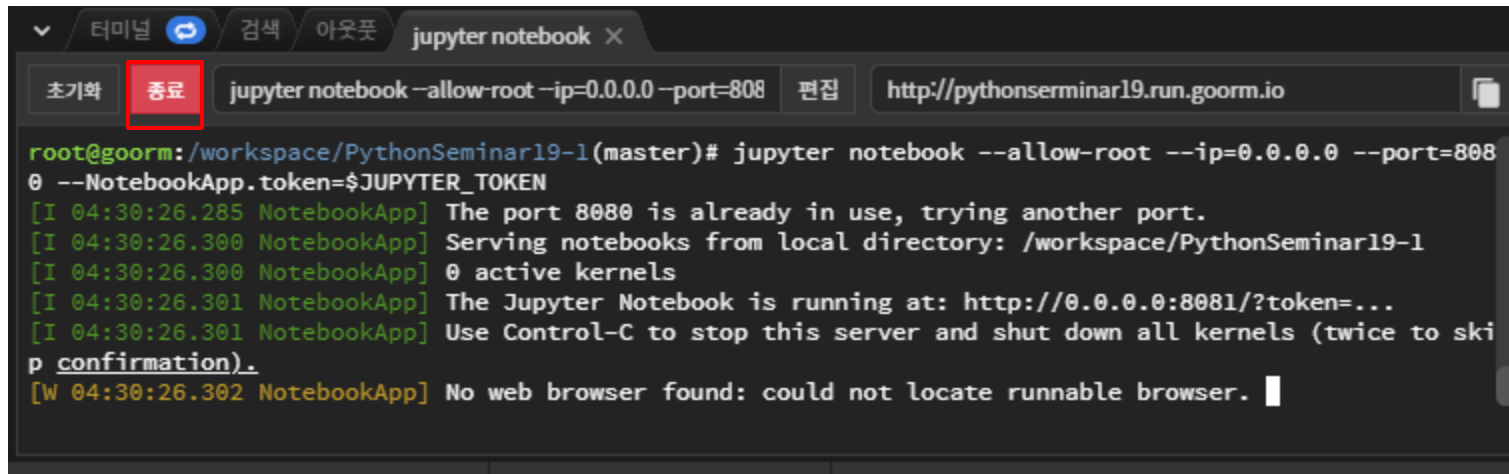
```
root@goorm:/workspace/PythonSeminar19-1(master)# jupyter notebook --allow-root --ip=0.0.0.0 --port=8080
0 --NotebookApp.token=$JUPYTER_TOKEN
[I 04:30:26.285 NotebookApp] The port 8080 is already in use, trying another port.
[I 04:30:26.300 NotebookApp] Serving notebooks from local directory: /workspace/PythonSeminar19-1
[I 04:30:26.300 NotebookApp] 0 active kernels
[I 04:30:26.301 NotebookApp] The Jupyter Notebook is running at: http://0.0.0.0:8081/?token=...
[I 04:30:26.301 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[W 04:30:26.302 NotebookApp] No web browser found: could not locate runnable browser.
```

Jupyter Notebook

구름IDE

—종료

—웹 브라우저를 닫고, 종료를 클릭

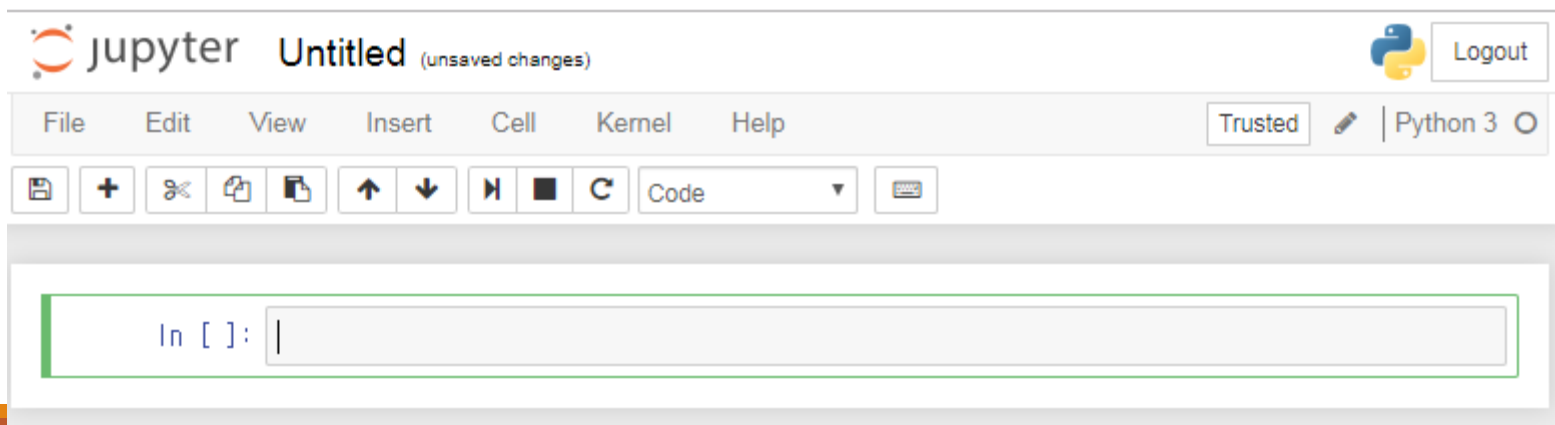
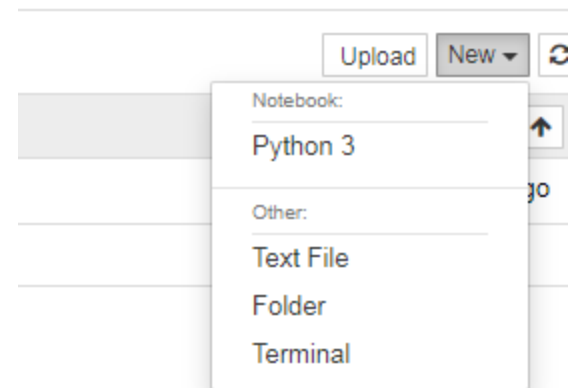


```
root@goorm:/workspace/PythonSeminar19-1(master)# jupyter notebook --allow-root --ip=0.0.0.0 --port=8080
0 --NotebookApp.token=$JUPYTER_TOKEN
[I 04:30:26.285 NotebookApp] The port 8080 is already in use, trying another port.
[I 04:30:26.300 NotebookApp] Serving notebooks from local directory: /workspace/PythonSeminar19-1
[I 04:30:26.300 NotebookApp] 0 active kernels
[I 04:30:26.301 NotebookApp] The Jupyter Notebook is running at: http://0.0.0.0:8081/?token=...
[I 04:30:26.301 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[W 04:30:26.302 NotebookApp] No web browser found: could not locate runnable browser.
```

Jupyter Notebook

사용법

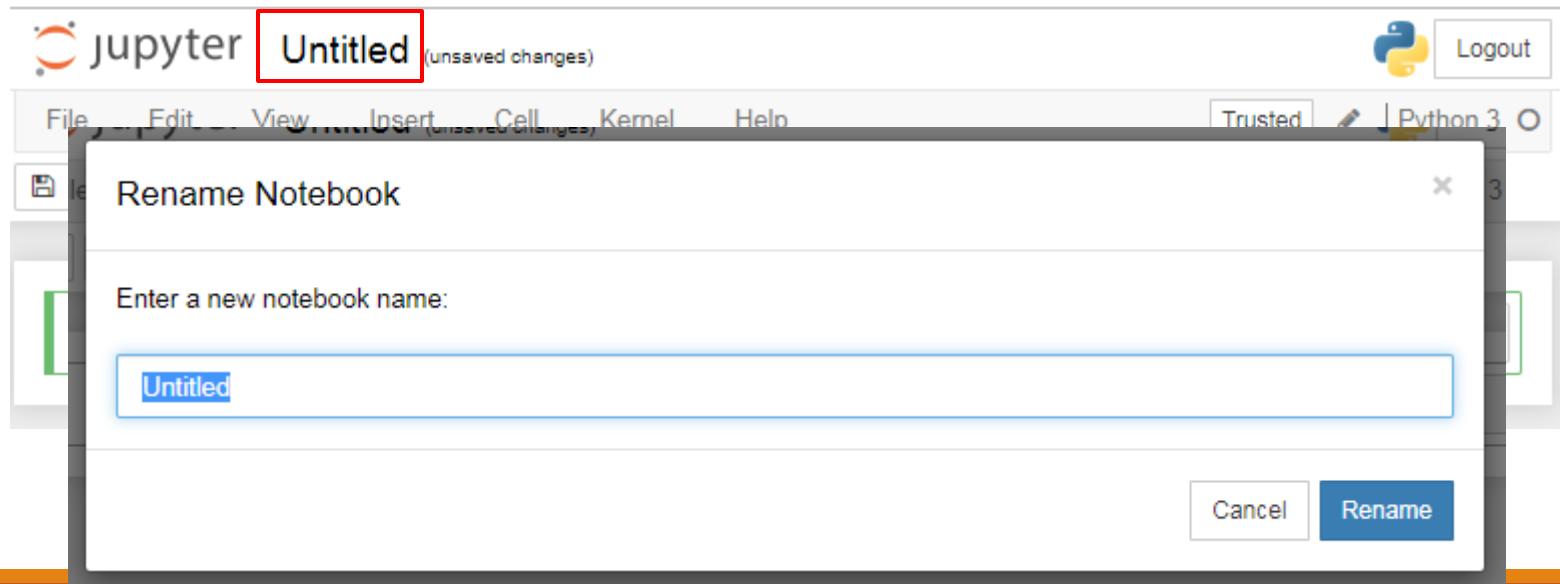
- 노트북 생성
 - 본인의 폴더로 이동
 - 오른쪽의 New-Python 3클릭
 - Untitled(생성된 파일)로 접속됨



Jupyter Notebook

사용법

- 노트북 이름 변경
 - 상단의 Untitled를 클릭
 - 제목 변경-Hello World



Jupyter Notebook

사용법

- hello world!
- print(“Hello World!”)를 입력
- shift+Enter 를 누르면 실행 후 아래 새로운 셀 생성

```
In [ ]: print("Hello World!")
```

```
In [4]: print("Hello World!")
```

Hello World!

```
In [ ]:
```

Jupyter Notebook

사용법

- hello world!
- 오류 생성시 바로 Error가 아래에 표시됨

```
In [5]: print("")  
File "<ipython-input-5-93263f49d639>", line 1  
    print()  
      ^  
SyntaxError: EOL while scanning string literal
```


Jupyter Notebook

사용법

- hello world!
- 수정 후, ctrl+enter(선택 셀만 재 실행)을 누르면 재실행

```
In [8]: print("test")
```

```
test
```

Jupyter Notebook

사용법

—Docstring

- 뒤에 ?를 붙이고 실행하면 하단에 Docstring이 나옴

```
In [9]: import math  
        math.sqrt?
```

Docstring:

sqrt(x)

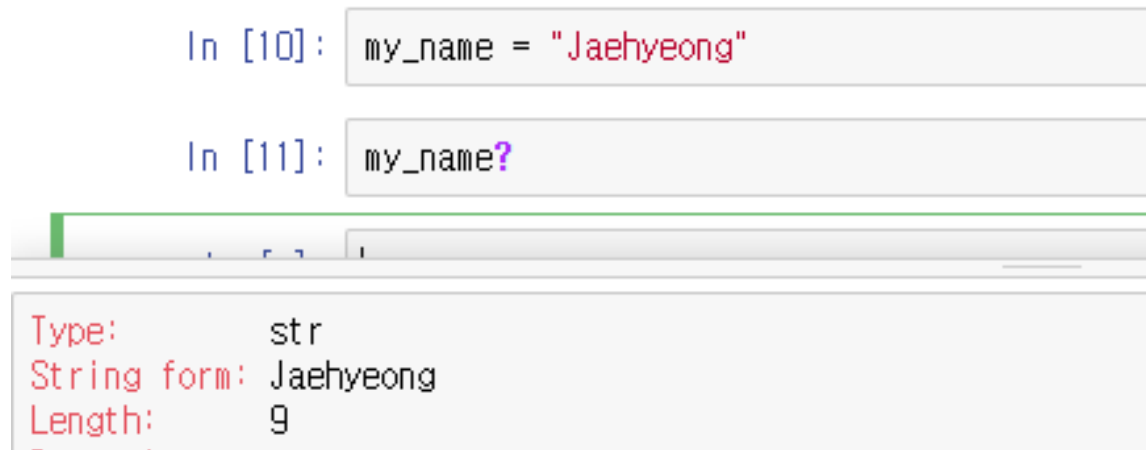
Return the square root of x.

Type: builtin_function_or_method

Jupyter Notebook

사용법

- 툴팁으로 확인하기
 - 뒤에 ?를 붙이고 shift+tap을 누른다



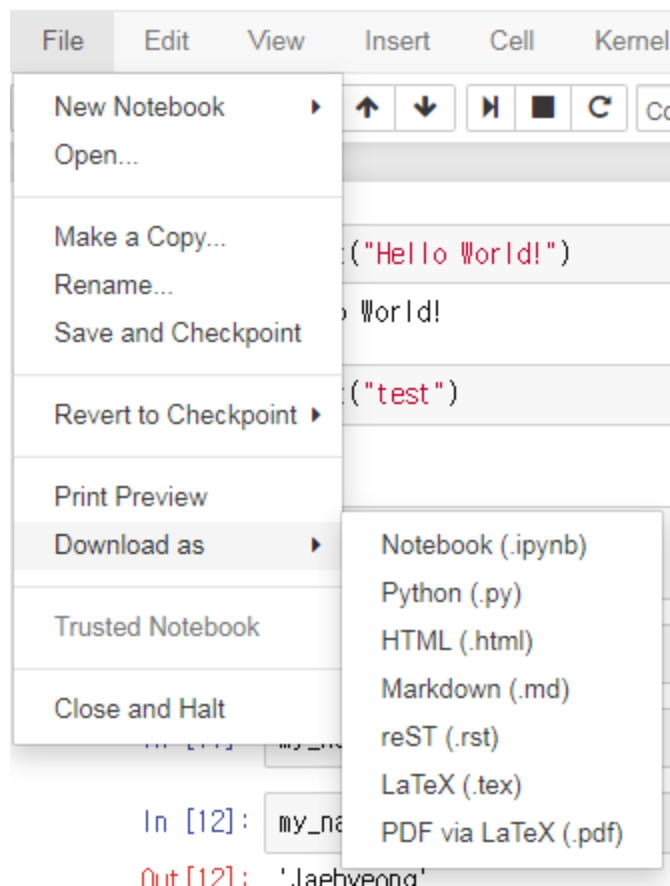
The screenshot shows a Jupyter Notebook interface. It contains two code cells. The first cell, labeled 'In [10]:', contains the code `my_name = "Jaehyeong"`. The second cell, labeled 'In [11]:', contains the code `my_name?`. Below the second cell, a tooltip is displayed, showing the type and value of the variable `my_name`. The tooltip text is as follows:

```
Type:      str
String form: Jaehyeong
Length:    9
```

Jupyter Notebook

사용법

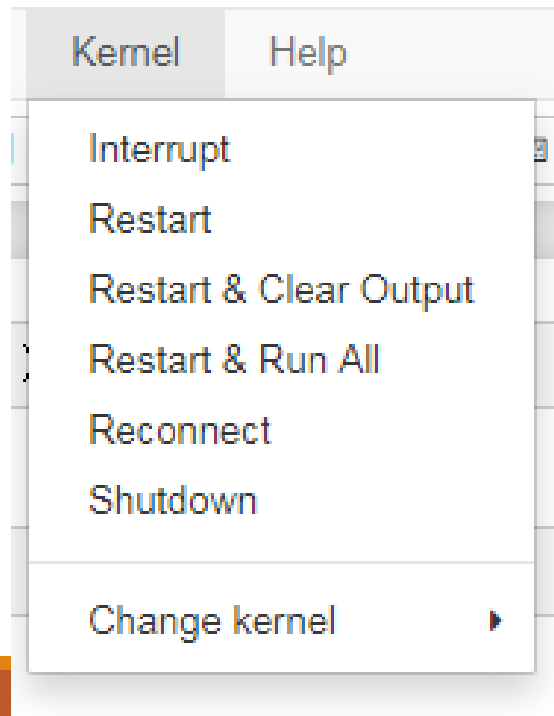
- 내보내기
 - 원하는 방식으로 파일을 내보낼 수 있다.



Jupyter Notebook

사용법

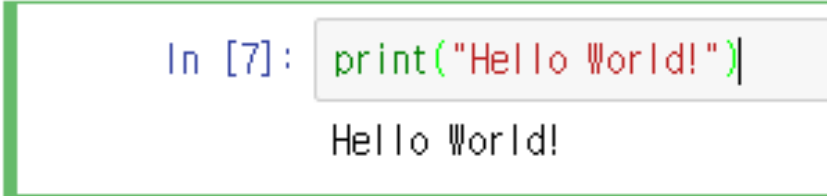
- 강제 종료
 - interrupt를 누르면 keyboard Interrupt를 한 것과 같다.




Jupyter Notebook

키보드 단축키

- 편집모드와 명령모드
 - 편집모드: 코드 셀 내용을 편집 (녹색)
 - 명령모드: 코드 셀에 대한 명령 (파란색)

A screenshot of a Jupyter Notebook cell in Edit mode. The cell is outlined with a green border. The prompt 'In [7]:' is in blue. The code 'print("Hello World!")' is in a light gray box with green text. Below the code, the output 'Hello World!' is displayed.

```
In [7]: print("Hello World!")  
Hello World!
```

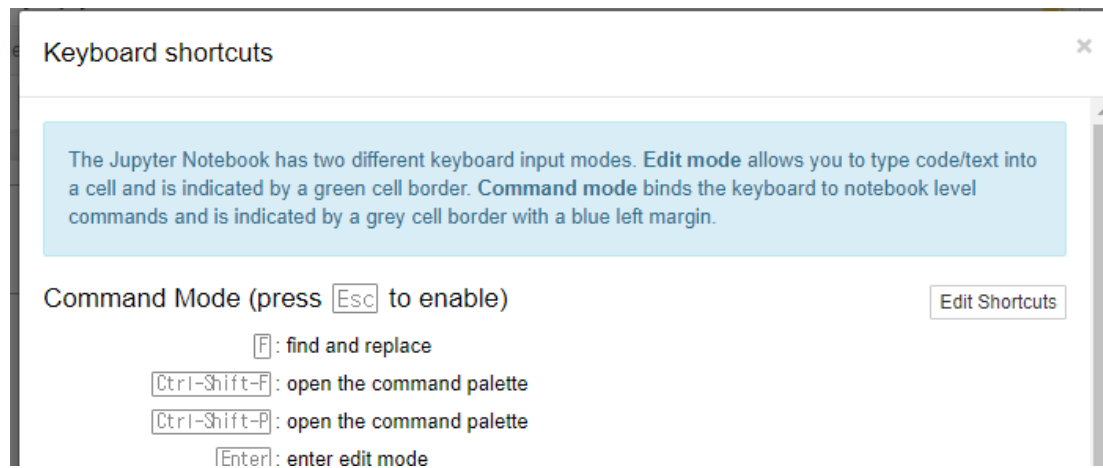
A screenshot of a Jupyter Notebook cell in Command mode. The cell is outlined with a blue border. The prompt 'In [7]:' is in blue. The code 'print("Hello World!")' is in a light gray box with green text. Below the code, the output 'Hello World!' is displayed.

```
In [7]: print("Hello World!")  
Hello World!
```

Jupyter Notebook

키보드 단축키

- 단축키 목록
- 명령모드에서 h를 누름



Jupyter Notebook

키보드 단축키

- 중요 단축키(편집모드)
 - Shift + Enter: 액티브 셀을 실행하고 다음 셀을 선택, 마지막일 경우 새로운 셀을 도입
 - Ctrl + Enter: 액티브 셀을 실행(명령모드도 가능)
 - Alt + Enter: 액티브 셀을 실행하고 아래에 새로운 셀 도입
 - Ctrl + Z: 되돌리기
 - Ctrl + Shift + Z: 되살리기
 - ESC: 명령모드로 전환

Jupyter Notebook

키보드 단축키

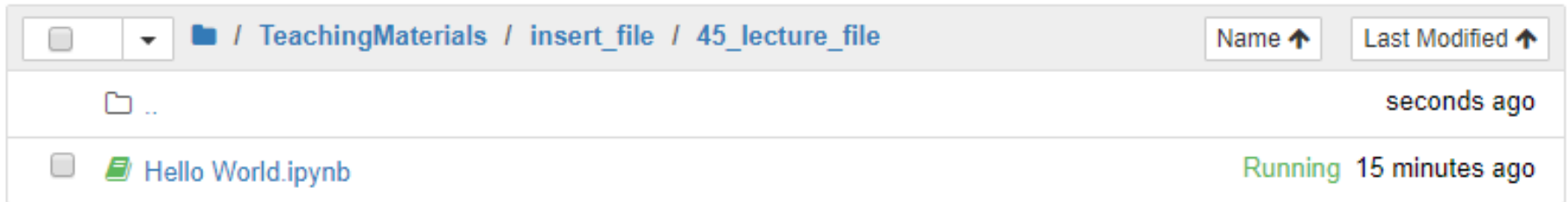
- 중요 단축키(명령모드)
 - Ctrl+S: Notebook 저장
 - A: 위에 새로운 셀을 도입
 - B: 아래에 새로운 셀을 도입
 - D, D: 선택 중인 셀 삭제
 - Z: 삭제한 셀을 복원
 - P; 명령 팔레트 열기(다양한 명령 목록 확인)
 - Enter: 선택 중인 셀을 편집모드로 변환

Jupyter Notebook

저장과 자동 저장

- 파일 생성

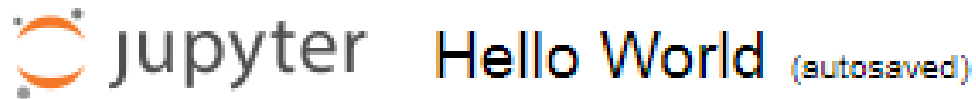
- 새로 작성한 시점에 Notebook 파일 만들어짐



Jupyter Notebook

저장과 자동 저장

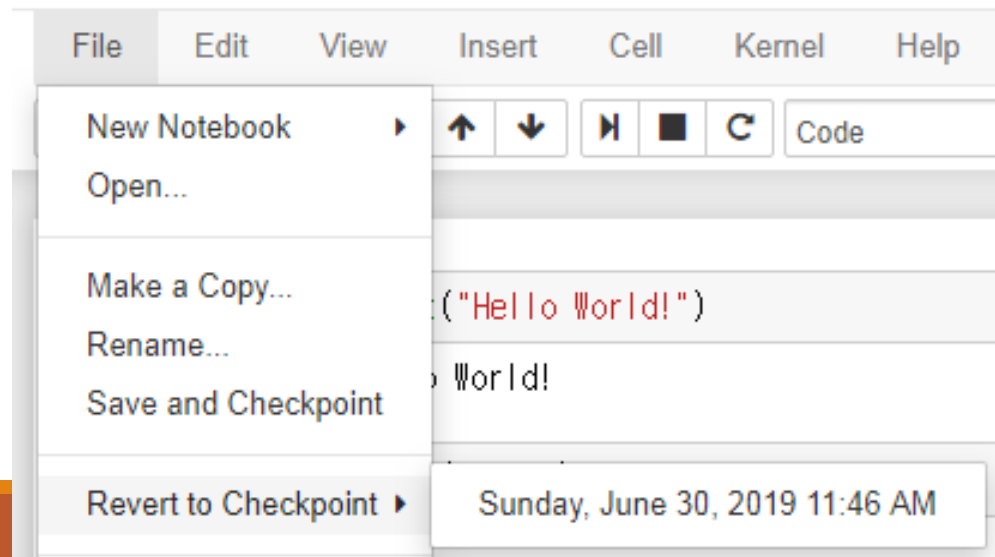
- 자동 저장(Auto Save)
 - 120초마다 Notebook의 상태를 자동 저장하는 기능
 - 저장 상황은 제목에 표시



Jupyter Notebook

저장과 자동 저장

- 체크포인트(check point)
 - 셀 실행유무, 코드의 입력 등을 저장하고, 되돌릴 수 있음
 - Auto Save로는 체크포인트가 생성되지 않음.
 - Revert to Checkpoint로 불러올 경우, Auto Save 저장분이 삭제



Jupyter Notebook

설명 입력

- Markdown에 대한 설명은 46.(Advanced) Markdown에서 설명한다.
- LaTeX는 수식을 입력하는 언어로
조판도 가능하나 매우 복잡하여 서술하지 않는다.

Jupyter Notebook

OS Shell 명령어 사용

- !를 붙이고 사용
- Ex) !cat /etc/hosts

In [2]: !cat /etc/hosts

```
127.0.0.1      localhost
::1           localhost ip6-localhost ip6-loopback
fe00::0       ip6-localnet
ff00::0       ip6-mcastprefix
```

Jupyter Notebook

매직명령어

- %가 앞에 붙는 명령어
- ex) %pwd

```
In [3]: %pwd
```

```
Out [3]: '/workspace/PythonSeminar19/TeachingMaterials/'
```

- !pwd와 같아 보이나, IPython에서 내부적으로 처리
- 변수로 받아서 사용가능

```
In [4]: directory = %pwd  
directory
```

```
Out [4]: '/workspace/PythonSeminar19/TeachingMaterials/'
```

Jupyter Notebook

매직명령어

- Cell/Line magic
 - %가 두 개 붙으면, 셀 전체가 대상이 됨(cell magics)
 - %가 한 개 붙으면, 그 라인만 대상이 됨(line magics)

Jupyter Notebook

매직명령어

- 자주 쓰는 명령
- %time 명령어
- 명령어 실행 시간 측정

```
In [5]: %time sum(range(100000))
```

```
CPU times: user 4 ms, sys: 0 ns, total: 4 ms  
Wall time: 2.57 ms
```

```
Out [5]: 4999950000
```

Jupyter Notebook

매직명령어

— 자주 쓰는 명령

- `%timeit -n` 반복수 `-r` 시행수 명령어
- 명령어 실행 시간 측정
- 반복수 만큼 반복하고, 시행수만큼 시행하여 평균, 표준편차 계산

```
%timeit -n 100 -r 5 sum(range(100000))
```

2.16 ms \pm 207 μ s per loop (mean \pm std. dev. of 5 runs, 100 loops each)

Jupyter Notebook

매직명령어

— 자주 쓰는 명령

- `%%timeit -n 반복수 -r 시행수` 명령어
- 명령어 실행 시간 측정
- Cell 전체에 대한 계산

```
%%timeit -n 100 -r 3  
  
for i in range(10000):  
    i * i
```

787 μ s \pm 277 μ s per loop (mean \pm std. dev. of 3 runs, 100 loops each)

Jupyter Notebook

매직명령어

- 자주 쓰는 명령
 - %history -l 개수
 - 셀 실행 이력을 개수만큼 보여줌

```
: %history -l 5

%pwd
directory = %pwd
directory
%time sum(range(100000))
%timeit -n 100 -r 5 sum(range(100000))
%%timeit -n 100 -r 3

for i in range(10000):
    i * i
```

Jupyter Notebook

매직명령어

— 자주 쓰는 명령

— %ls

— 현재 디렉터리의 파일 목록 취득

— 단, OS와 상관 없이 사용 가능(windows의 명령어는 dir)

```
: %ls
```

```
Hello World.ipynb
```

Jupyter Notebook

매직명령어

- %autosave
 - 자동 저장 기능 빈도 수정
 - %autosave 시간(초)
 - 0일 경우, 자동저장을 하지 않는다.