

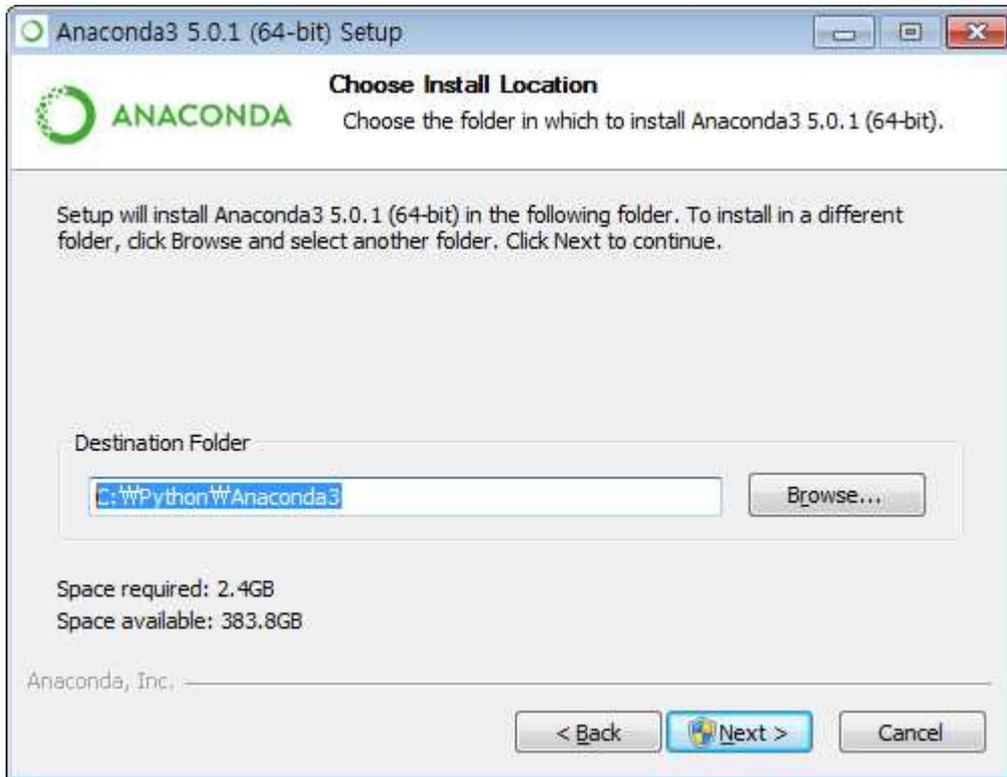
1. 아나콘다 설치와 주피터

- ① Anaconda(아나콘다) - Python 기반의 데이터 분석에 필요한 오픈소스를 모아놓은 개발 플랫폼이다.
 - 가상환경 관리자 제공 / - 패키지 관리자 제공
- ② 아나콘다에 포함된 수준 높은 패키지 관리자를 통해서 파이썬의 효율성을 극대화 시켜 활용할 수 있습니다.
- ③ 아나콘다의 또 다른 장점은 가상환경 관리자를 각 프로젝트별 개발 환경을 효율적으로 구성할 수 있다

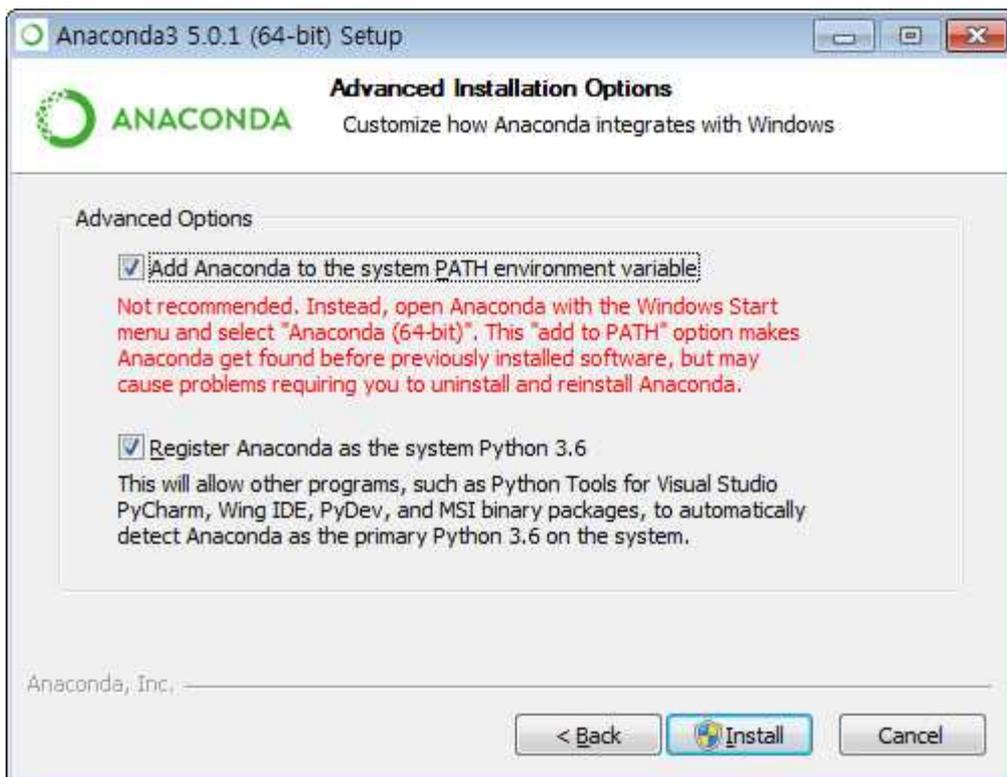
- 1) 크롬 브라우저 실행. (익스프로러는 윈도우 설치파일 다운 안됨)
- 2) <https://www.anaconda.com/download/> 링크로 이동 설치파일 다운로드
(64bit or 32bit 개발환경에 맞는 버전다운)
(Anaconda3-5.0.1-Windows-x86_64 , Anaconda3-5.0.1-Windows-x86)
- 3) 설치 파일 실행



All Users를 선택하는 걸 권장 드립니다.

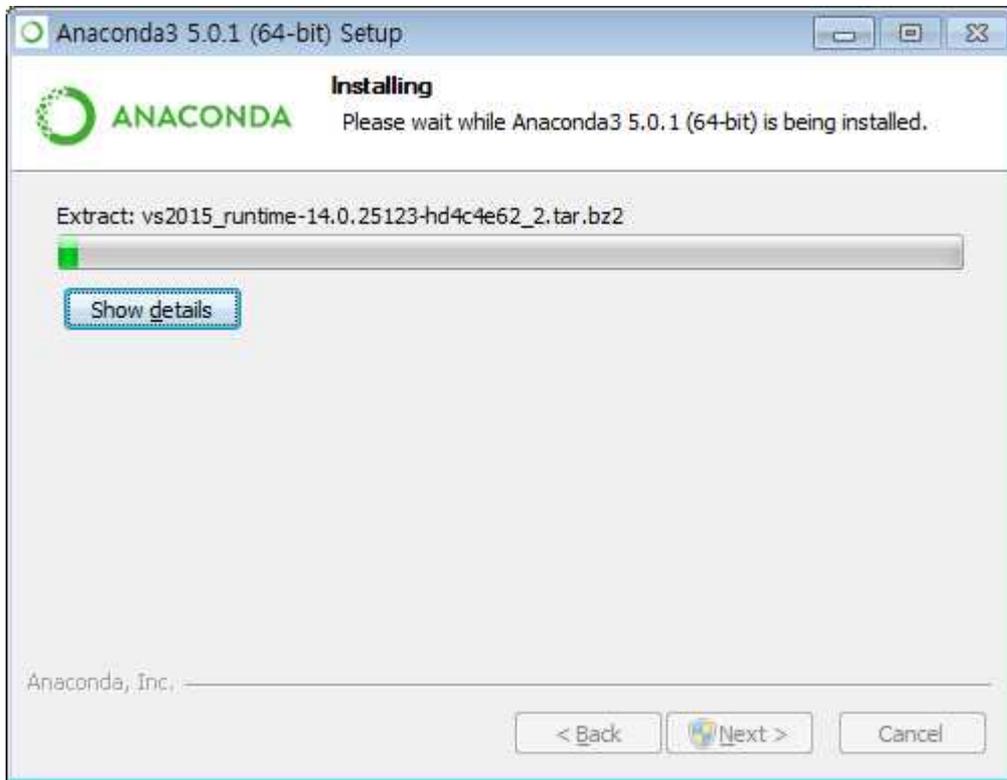


설치 폴더를 선택하는 건데 원하는 경로로 지정 하셔도 되지만 움직이지 않는 폴더로 지정해 놓으셔야 합니다.
 필자는 "C:\Python\Anaconda3" 설정함



Options는 2개 모두 체크 선택

아나콘다의 경로를 환경 변수의 PATH에 추가할지 설정하는 화면입니다.
 이미 파이썬이 설치되어 있다면 "Add Anaconda to my PATH environment variable"의 체크를 해제해
 기존 python.exe와 혼동되지 않도록 아나콘다의 경로는 PATH에 추가하지 않습니다.



설치가 완료되면 아래 아이콘을 클릭하거나 없다면, 검색에서 Anaconda Prompt 를 찾아 실행합니다.



프롬프트가 준비되면 python 명령을 실행해서 파이썬 셸(python shell)로 들어갑니다. 파이썬 셸이 준비되면 아래 명령을 통해 파이썬이 잘 작동하는지 확인해봅니다.

```
python
(C:\Users\jaewo\Anaconda3) C:\Users\jaewo>python
Python 3.6.1 [Anaconda 4.4.0 (64-bit)] (default, May 11 2017, 13:25:24) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("hello world")
hello world
>>>
```

[기본 명령어]

#아나콘다 버전 확인

```
conda --version
```

#아나콘다 정보 보기

```
conda info
```

#아나콘다 업데이트

```
conda update conda
```

#아나콘다 가상환경 생성

```
conda create --name(-n) 가상환경명 설치할 패키지
```

#예) 파이썬 3.5 버전 설치 & test 이름으로 가상환경 생성

```
conda create --name test python=3.5 <=> conda create --n test python=3.5
```

아나콘다 가상환경 목록보기

```
conda info --envs <=> conda info -e
```

아나콘다 개발환경 활성화 하기

```
activate <개발환경 이름>
```

아나콘다 개발환경 제거

```
conda remove --name <개발환경 이름> --all
```

#패키지 설치

#예)conda install 패키지명

```
conda install simplejson
```

#패키지 리스트 확인

```
conda list
```

#패키지 삭제

#예)conda remove --name test --all

```
conda remove --name 가상환경명 --all <=> conda remove -n 가상환경명 --all
```

#아나콘다 클린(인덱스 캐시, 잠긴 파일, 사용하지 않는 패키지, 소스 캐시 등을 삭제)

```
conda clean -all <=> conda clean -a
```

2. 주피터 사용

Jupyter NoteBook 이란 아나콘다 환경을 웹에서 사용할 수 있도록 연결해주는 웹서비스이다. Jupyter NoteBook 을 실행하면 아나콘다 내부에 설치되어 있는 패키지들을 웹에서 사용할 수 있다.

아나콘다 설치가 완료되었다면, 윈도우 기준으로 윈도우 파워셸(Windows PowerShell)을 켜주자. 명령 프롬프트를 실행(윈도우 키+R 을 누른 뒤 cmd 를 입력)한 뒤 다음 명령을 입력합니다.

```
$ jupyter notebook
```

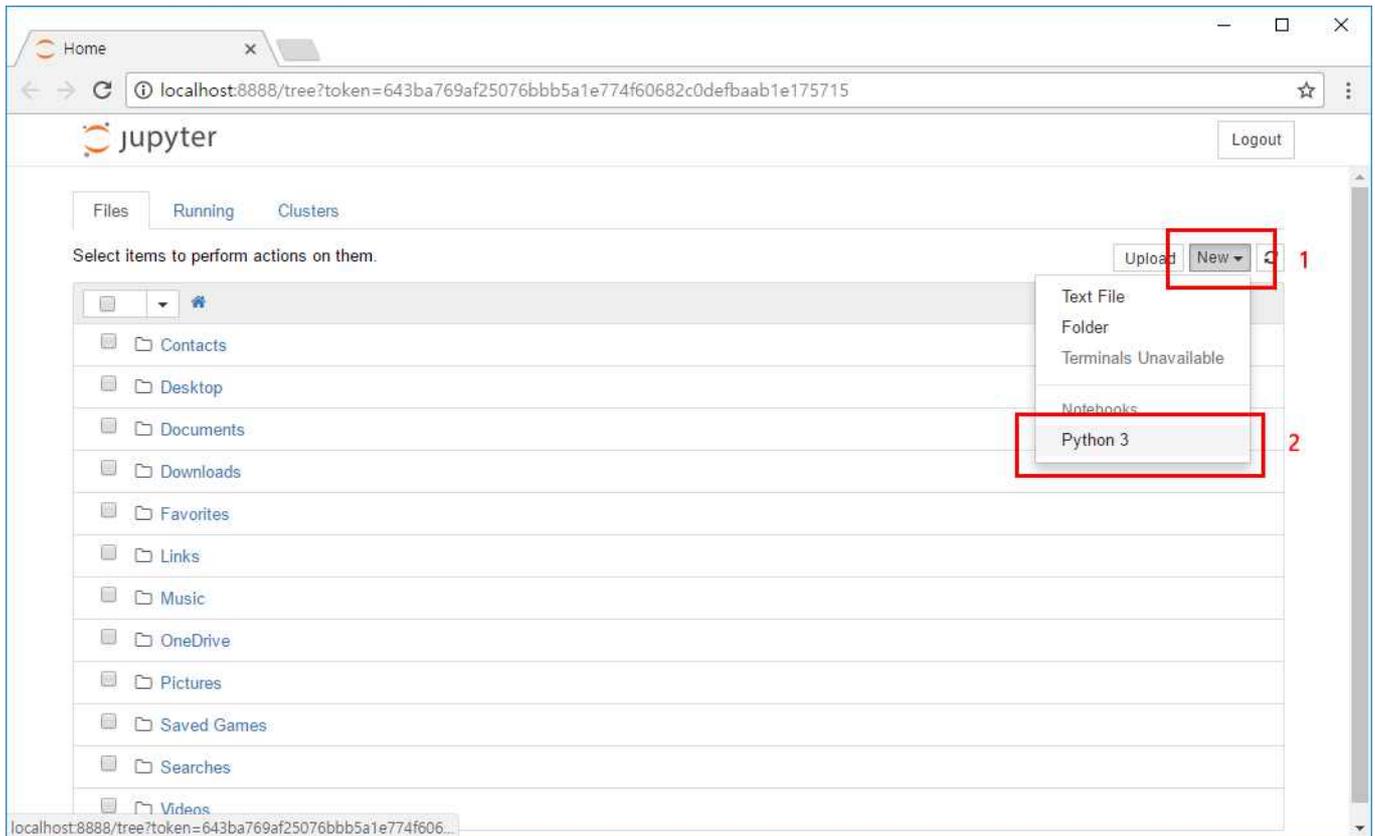
시작 > Anaconda3 (64-bit) > Jupyter Notebook 을 클릭합니다. 또는, 명령 프롬프트를 실행(윈도우 키+R 을 누른 뒤 cmd 를 입력)한 뒤 다음 명령을 입력합니다.

- **C:\Users\<사용자계정>\Anaconda3\python.exe -m notebook**

```
C:\Users\dojang>C:\Users\dojang\Anaconda3\python.exe -m notebook
```

명령을 실행하면 웹 브라우저에 주피터 노트북이 표시됩니다. 이 화면에서 파이썬 노트북을 만들어보겠습니다. 오른쪽 **New** 버튼을 클릭한 뒤 **Python 3** 을 클릭합니다.

▼ 그림 1. 주피터 노트북 초기 화면



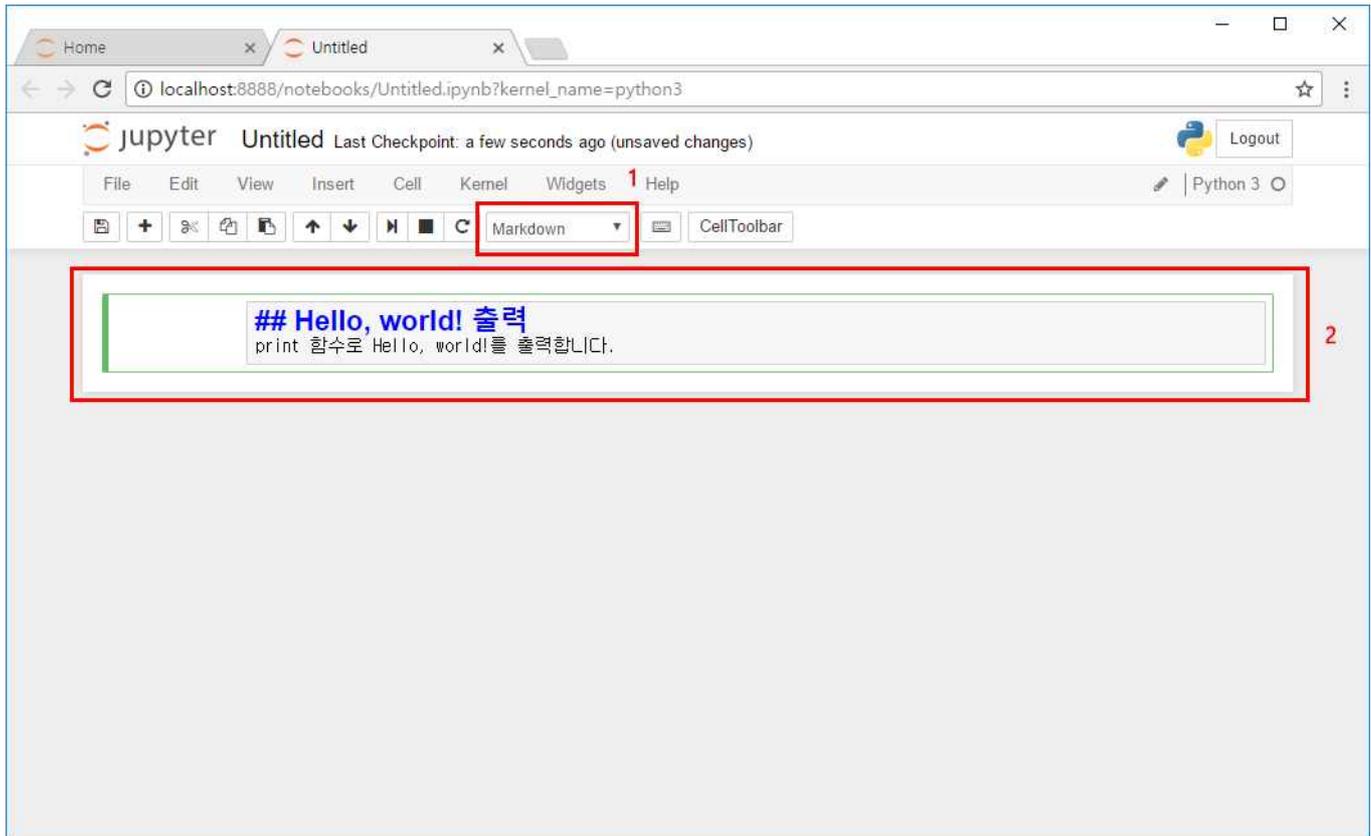
이제 새 노트북 화면이 나옵니다. 주피터 노트북은 노트북이라는 말 그대로 공책을 사용하듯이 코드를 작성하면서 설명도 함께 넣을 수 있습니다.

그럼 코드를 작성하기 전에 먼저 설명부터 넣어보겠습니다. 메뉴의 드롭다운 목록에서 **Markdown** 을 선택하고 아래 빈칸에 다음 내용을 입력합니다. #은 제목이라는 뜻이며 #이 하나씩 늘어날 수록 하위 제목이 됩니다.

Hello, world! 출력

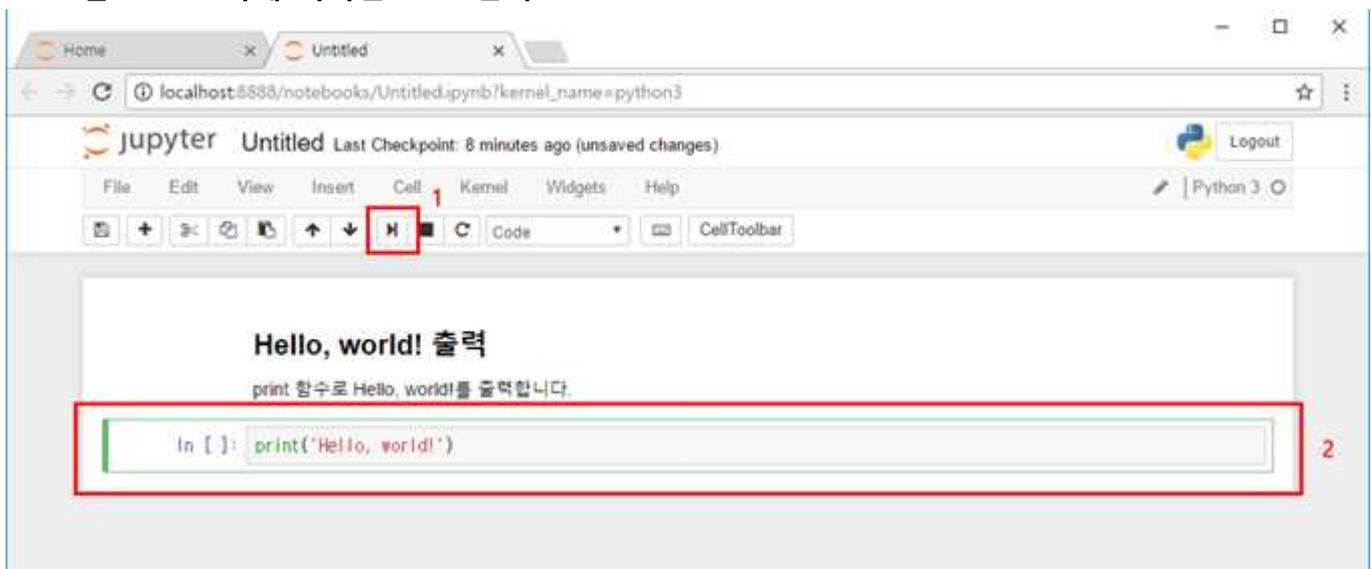
print 함수로 Hello, world!를 출력합니다.

▼ 그림 2. 노트북 생성 및 설명 추가



이제 설명을 적용한 뒤 파이썬 코드를 입력해보겠습니다. 메뉴에서 ▶| 버튼을 클릭하면 설명이 적용되고 아래에 셀(Cell)이 생깁니다. In []: 오른쪽에 print('Hello, world!')를 입력합니다.

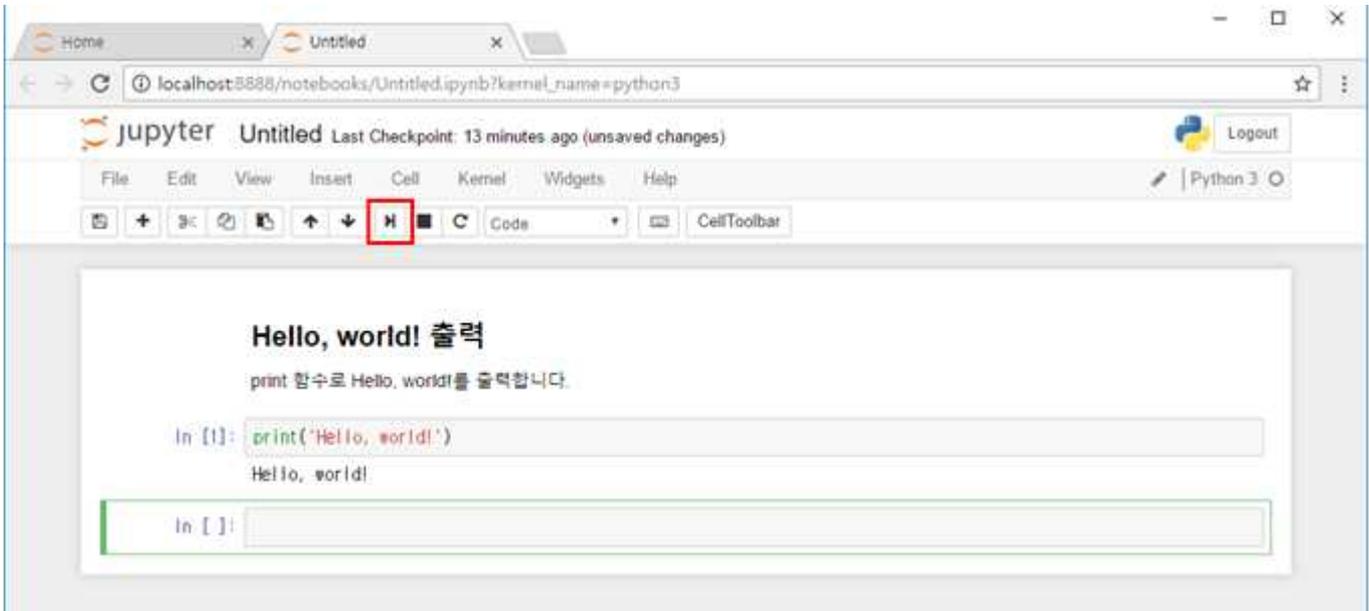
▼ 그림 3. 노트북에 파이썬 코드 입력



파이썬 코드를 입력했으면 실행을 해보겠습니다. 메뉴에서 다시 ▶| 버튼을 클릭하면 코드가 실행되고 결과가 출력됩니다. 이때 In []:이 In [1]:로 바뀌는데 1은 첫 번째로 실행된 코드라는 뜻입니다.

노트북에서는 코드 입력 셀이 추가될 수록 숫자가 높아집니다.

▼ 그림 4. 노트북의 파이썬 코드 실행



이런 방식으로 셀을 추가하면서 코드와 설명을 작성합니다. 주피터 노트북은 코드, 코드의 실행 결과, 코드에 대한 설명을 한 번에 작성할 수 있어서 체계적인 기록이 가능합니다.

특히 한 노트북 안에 들어있는 코드 셀은 모두 실행 상태가 연결됩니다. 즉, 코드 셀에서 변수를 만들면 다른 코드 셀에서 해당 변수를 사용할 수 있습니다. 이때 코드 셀의 실행 순서는 In []:에 표시된 숫자 순서를 따르며 메뉴의 ↑ ↓ 버튼을 이용하여 코드 셀의 위치를 바꿀 수 있습니다(코드 셀의 위치를 바꾸더라도 실행 순서는 In []:의 숫자를 따름).

참고로 노트북의 제목을 바꾸려면 맨 위 Jupyter 로고 옆의 Untitled 를 클릭하면 됩니다. 그리고 메뉴의 **File > Save and Checkpoint** 를 클릭하면 노트북이 파일로 저장됩니다. 이번 실습에서는 노트북의 제목을 바꾸지 않았으므로 C:\Users\<사용자계정> 폴더에 Untitled.ipynb 파일로 저장됩니다(노트북이름.ipynb 형식).

3. 단축키

Command Mode	Enter : 에디트 모드(Edit Mode)로 진입	Y : 코드(Code) 셀로 전환
	R : Raw NB 셀로 전환	M : 마크다운(Markdown) 셀로 전환
	1~6 : 마크다운 H1~H6	a : 위에 셀 삽입
	b : 아래 셀 삽입	

Edit Mode	Tab : 코드 자동완성, 들여쓰기	Ctrl+] : 들여쓰기(intent)
	Ctrl+[: 내어쓰기(detent)	Ctrl+a : 전체 선택
	Ctrl+z : 실행 취소(undo)	Esc : 커맨드 모드(Command mode)로 진입
	Shift+Enter : 셀 실행하고 다음 셀 선택	Ctrl+Enter : 셀 실행
	Alt+Enter : 셀 실행하고, 아래 셀 삽입	

[참고 | 노트북 파일 저장 경로 바꾸기]

주피터 노트북은 기본적으로 C:\Users\<사용자계정> 폴더에 노트북 파일을 저장합니다. 노트북 파일 저장 경로를 바꾸려면 --notebook-dir 옵션에 폴더를 지정해주면 됩니다.

```
C:\Users\dojang>C:\Users\dojang\Anaconda3\python.exe -m notebook --notebook-dir C:\project
```

시작 > Anaconda3 (64-bit) > Jupyter Notebook 의 노트북 파일 저장 경로를 바꾸는 방법은 다음과 같습니다.

Jupyter Notebook 메뉴에서 마우스 오른쪽 버튼을 클릭하고 **자세히 > 파일 위치 열기** 를 클릭

탐색기 창에서 **Jupyter Notebook** 바로가기를 선택한 뒤 마우스 오른쪽 버튼을 클릭하고 **속성(R)** 클릭

속성 창의 **대상(T)**에서 %USERPROFILE%을 지우고 C:\project 입력

참고 | 아나콘다를 설치하지 않고 주피터 노트북을 사용하려면?

아나콘다를 설치하지 않고 주피터 노트북을 사용하려면 pip 로 notebook 패키지를 설치한 뒤 jupyter notebook 으로 실행하면 됩니다. 단, Windows 에서는 패키지 설치 중에 에러가 나는 경우가 많습니다. 이때는 '부록 48.10 Windows 에서 패키지 설치 에러 해결하기'를 참조하세요.

Windows

```
C:\Users\dojang>pip install notebook
```

```
C:\Users\dojang>jupyter notebook
```

리눅스나 macOS 에서는 notebook 패키지를 설치한 뒤 jupyter notebook 으로 실행합니다.

리눅스, macOS

```
$ sudo pip install notebook
```

```
$ jupyter notebook
```

노트북 파일 저장 경로는 **jupyter notebook --notebook-dir 디렉터리** 형식으로 지정하면 됩니다.

참고 | IPython Notebook 과 Jupyter Notebook

IPython 은 Interactive Python 이라는 파이썬 인터프리터인데 기존 파이썬 인터프리터에 각종 편의기능을 추가한 버전입니다(In [1]:처럼 나오는 것이 IPython 의 프롬프트). 여기에 노트북 기능을 붙여서 IPython Notebook 이 나왔는데 이후 버전이 올라가고 파이썬 이외의 다른 프로그래밍 언어(Ruby, R, JavaScript 등)도 지원하면서 이름을 Jupyter Notebook 으로 바꾸게 됩니다. 그래서 최신 버전은 Jupyter Notebook 이며 IPython Notebook 은 따로 찾아서 설치하지 않아도 됩니다.