

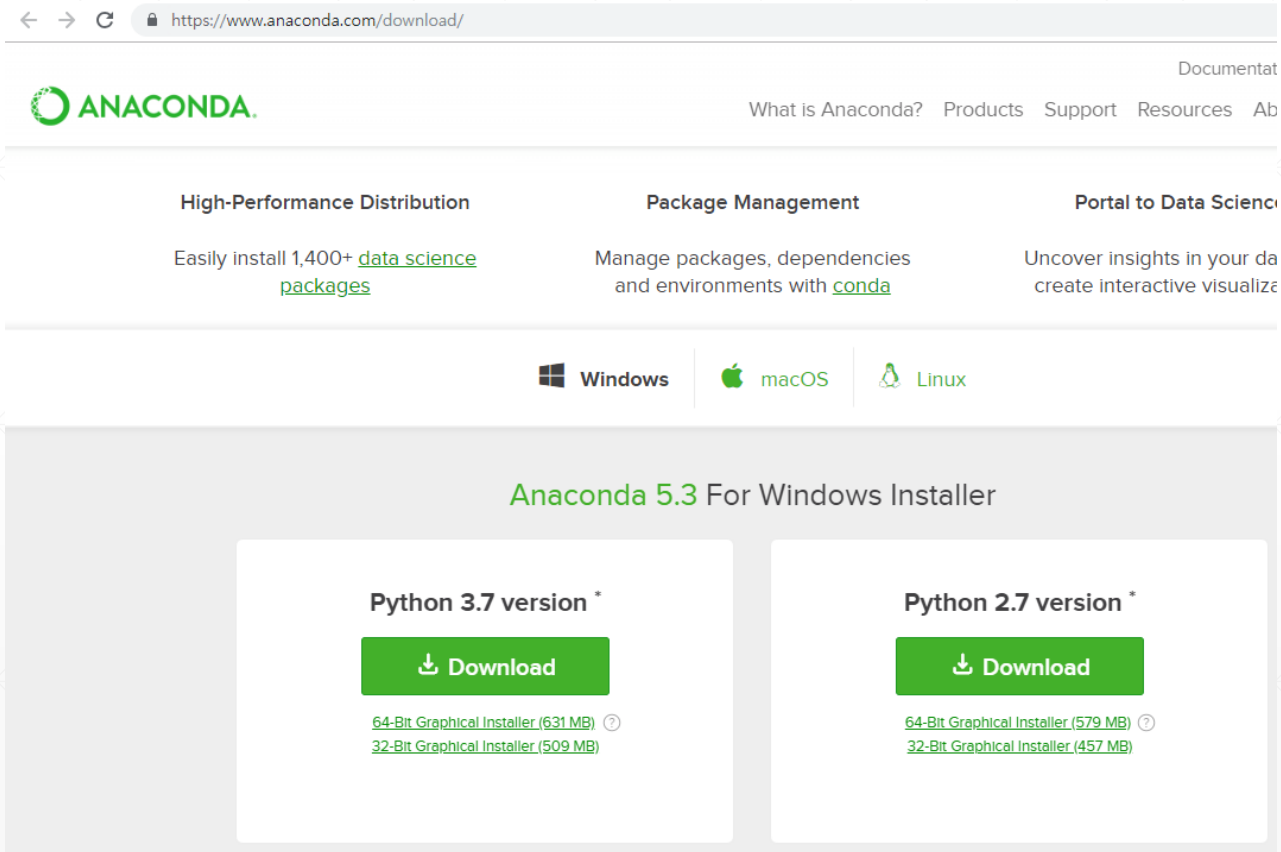
TensorFlow 기반의 딥러닝 구현

Python 기반의 개발환경과 개발도구 설치 및 설정

Python 개발 환경 구축

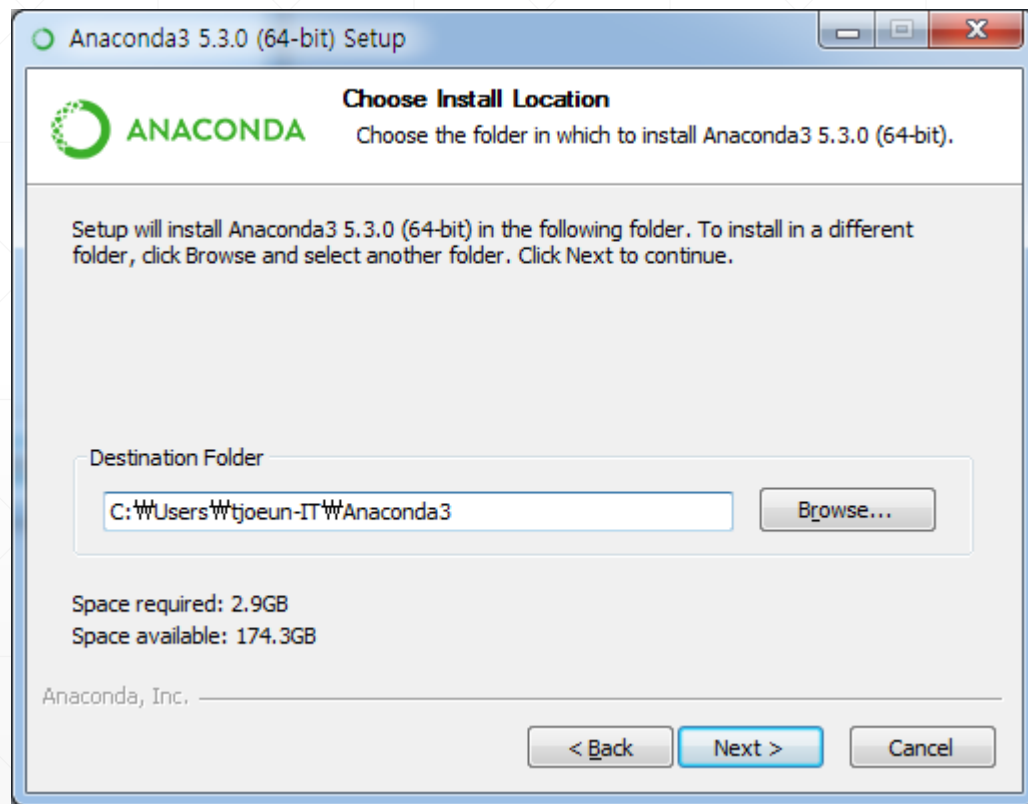
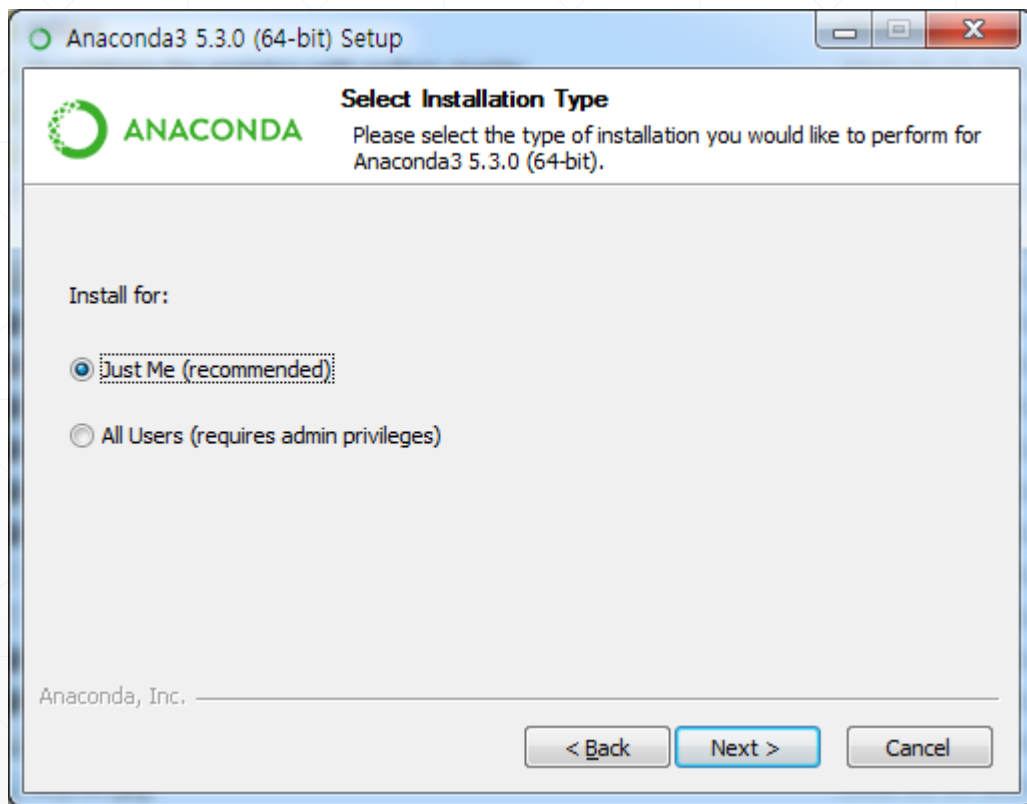
Anaconda 설치 및 가상환경 구축

Anaconda 다운로드

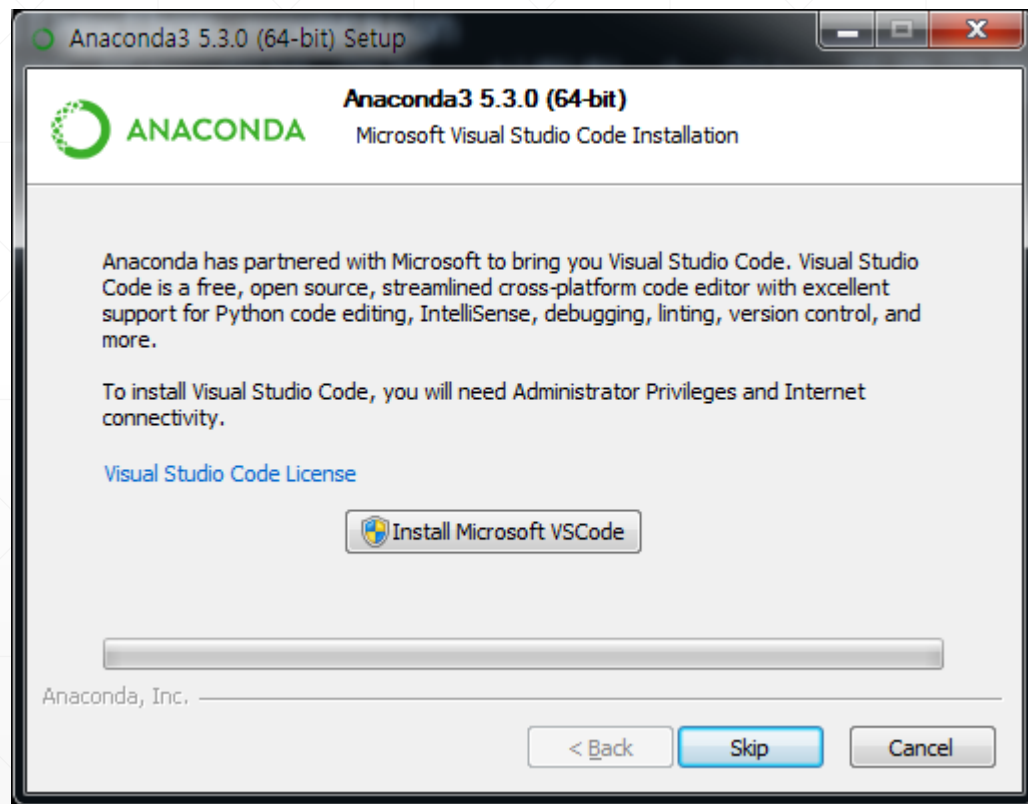
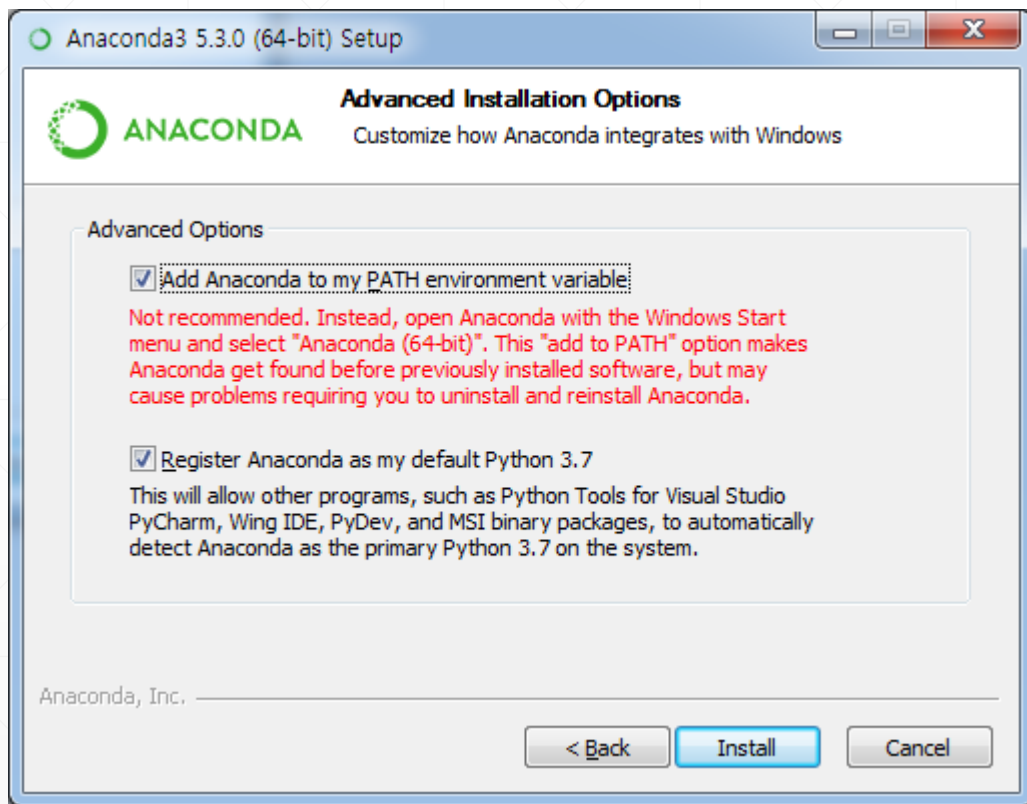


- <https://www.anaconda.com/download/>
- 시스템에 설치된 Python과 별도로 운영되는 환경
- 서로 다른 버전의 Python을 운영하고자 하는 경우 유용함
- 방대한 라이브러리를 기본 제공하며, 라이브러리의 추가 및 삭제가 용이함

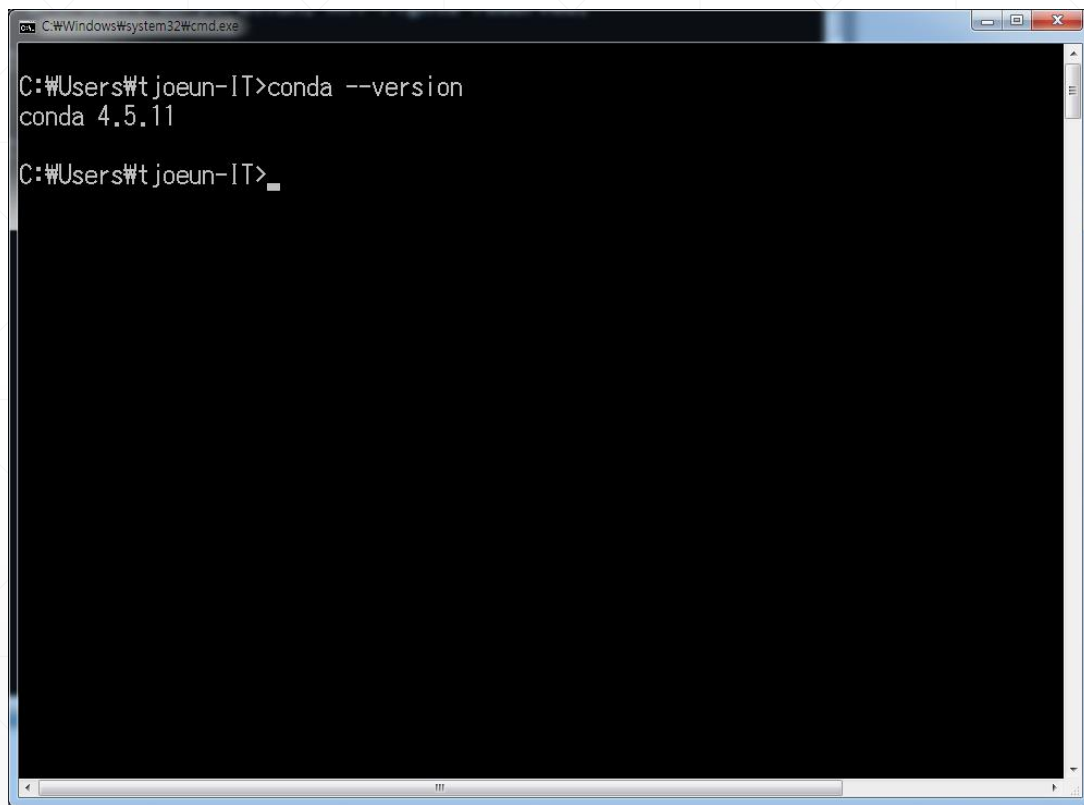
Anaconda 설치 과정 (1/2)



Anaconda 설치 과정 (2/2)

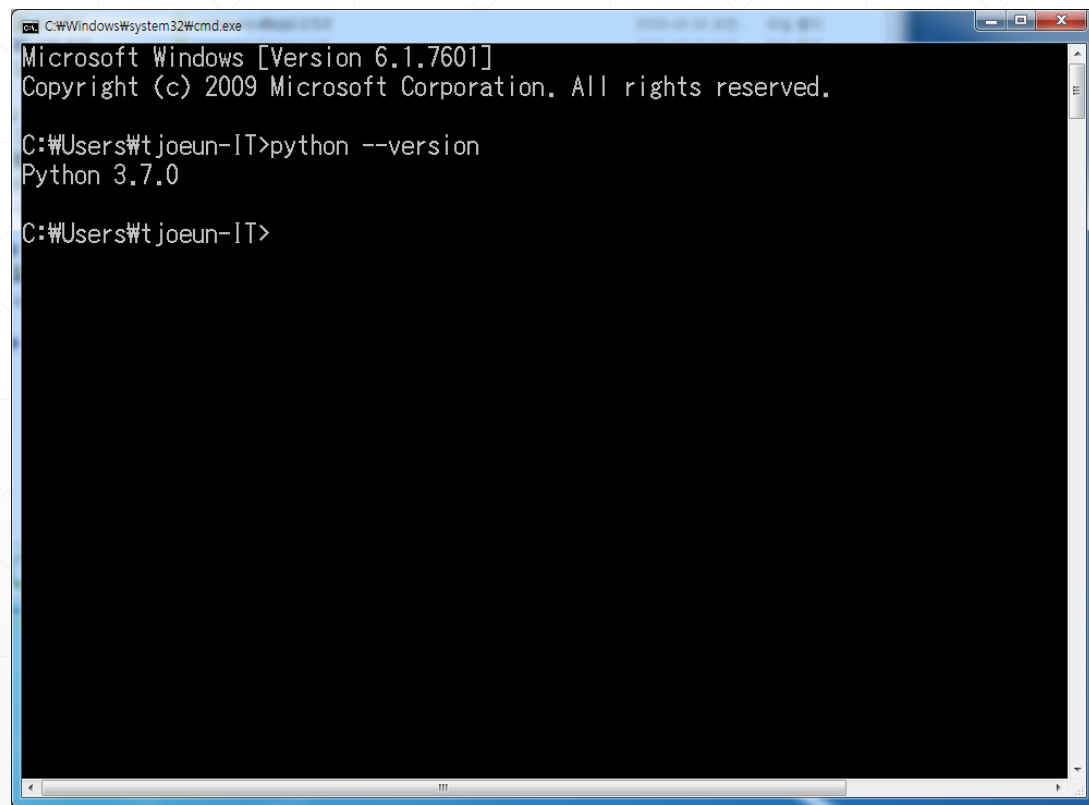


Anaconda 환경 테스트 (1/3)



A Windows command prompt window titled 'C:\Windows\system32\cmd.exe'. The prompt is 'C:\Users\jtjoeun-IT>'. The user has entered the command 'conda --version', and the output is 'conda 4.5.11'. The prompt is now 'C:\Users\jtjoeun-IT>' with a cursor.

```
C:\Windows\system32\cmd.exe
C:\Users\jtjoeun-IT>conda --version
conda 4.5.11
C:\Users\jtjoeun-IT>
```



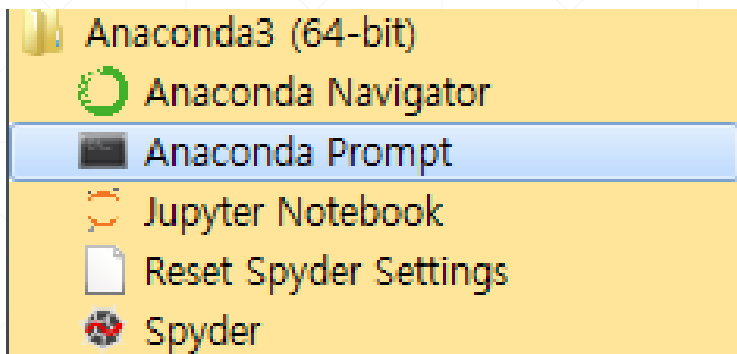
A Windows command prompt window titled 'C:\Windows\system32\cmd.exe'. The prompt is 'C:\Users\jtjoeun-IT>'. The user has entered the command 'python --version', and the output is 'Python 3.7.0'. The prompt is now 'C:\Users\jtjoeun-IT>' with a cursor.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

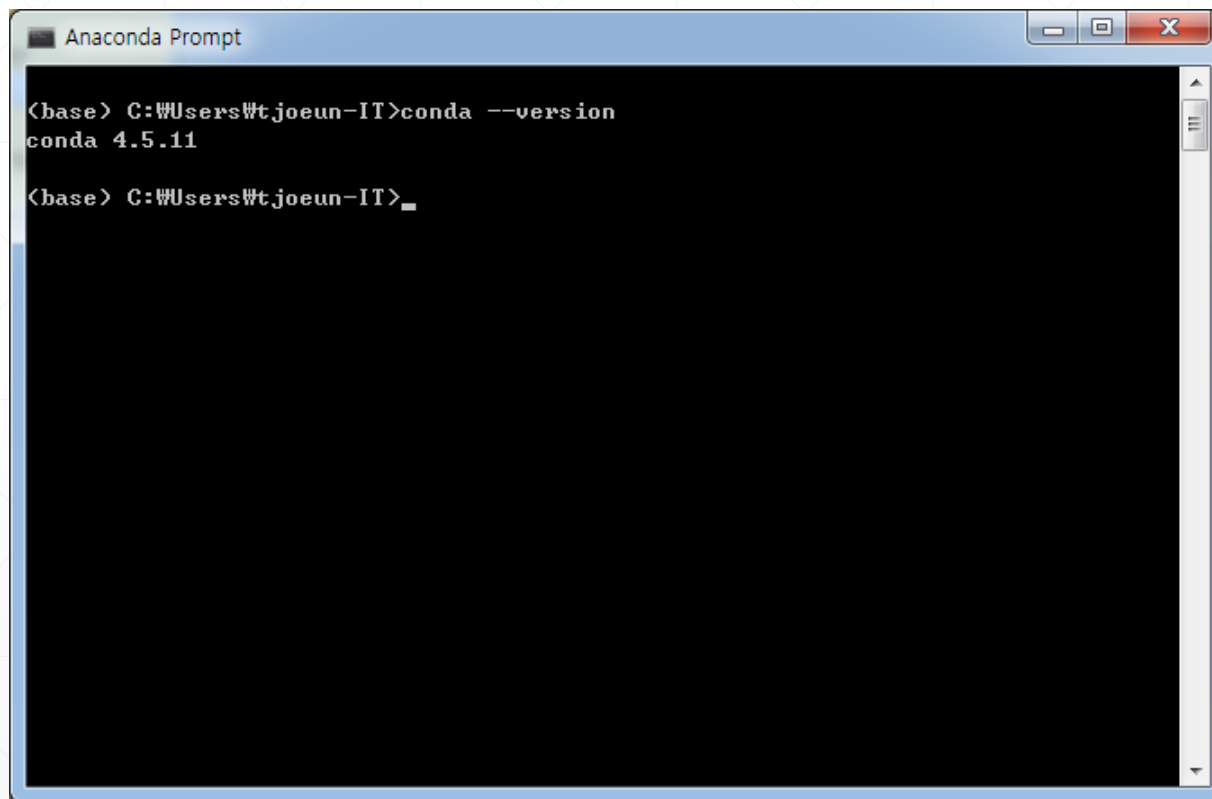
C:\Users\jtjoeun-IT>python --version
Python 3.7.0

C:\Users\jtjoeun-IT>
```

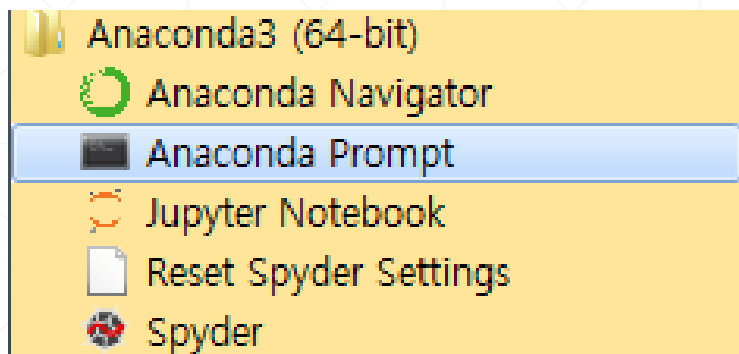
Anaconda 환경 테스트 (2/3)



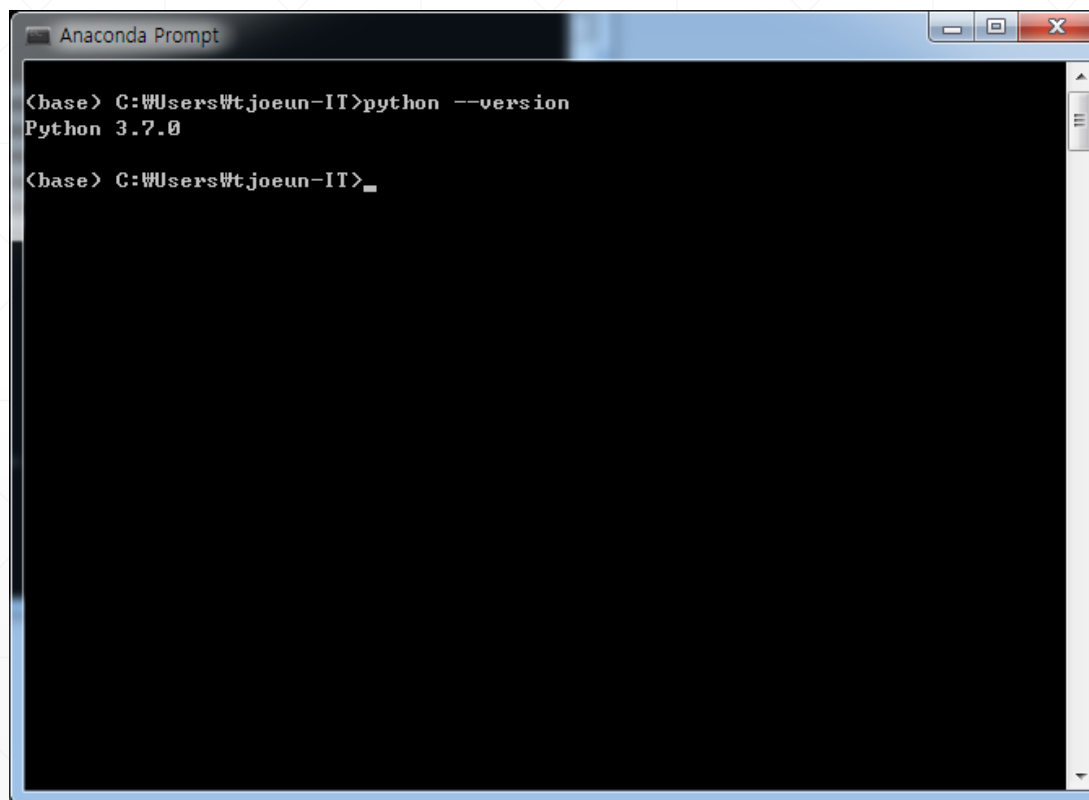
- 시작 -> 모든 프로그램
- Anaconda3 메뉴를 확장



Anaconda 환경 테스트 (3/3)



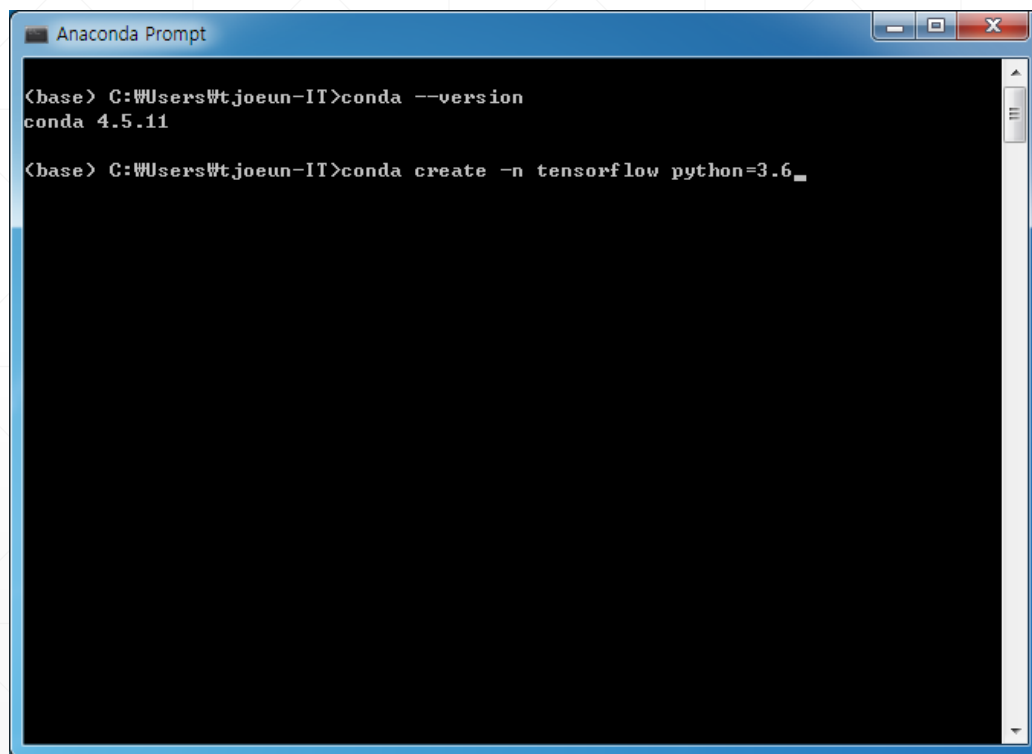
- 시작 -> 모든 프로그램
- Anaconda3 메뉴를 확장



Anaconda 가상환경 구축 (1/6)

- Anaconda 가상환경
 - 시스템에 설치된 Python과 별도로 운영되는 환경
 - 현재 머신에 설치되어 있는 Python과 별도로 동작됨
 - 서로 다른 버전의 Python을 운영하고자 하는 경우 유용함
 - 설치 및 삭제가 용이함
 - `conda create -n 가상환경이름 python=버전 [anaconda]`
 - anaconda 옵션을 사용하면 기본 환경이 복제됨(시간이 오래 걸릴 수 있음)
-

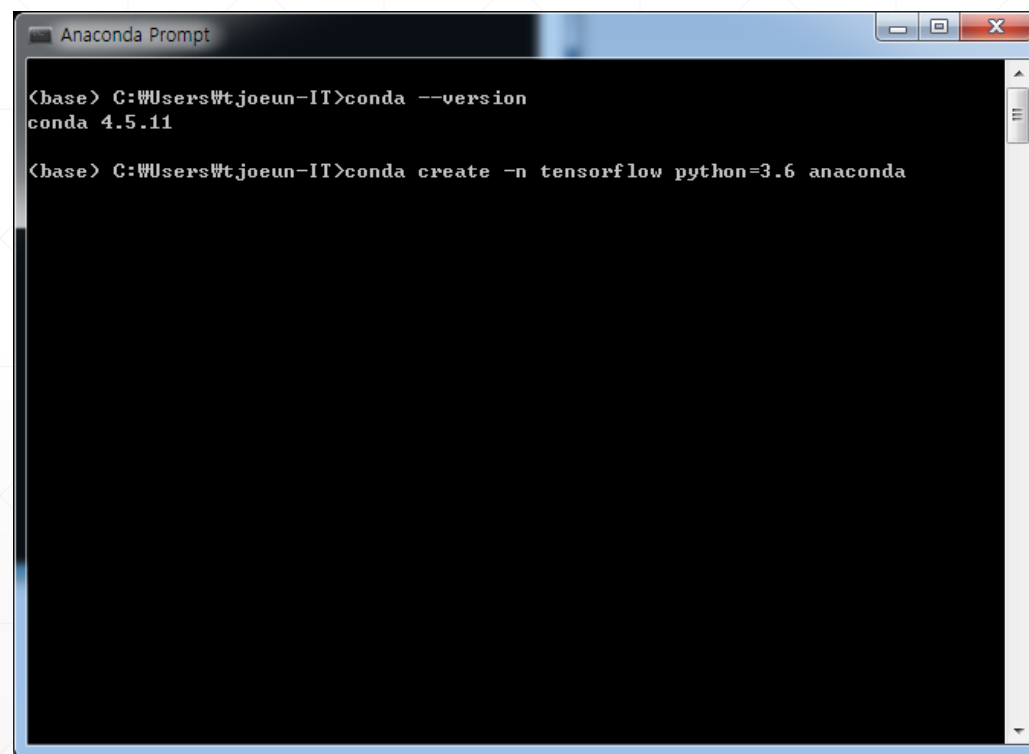
Anaconda 가상환경 구축 (2/6)



```
Anaconda Prompt

(base) C:\Users\wtjoeun-IT>conda --version
conda 4.5.11

(base) C:\Users\wtjoeun-IT>conda create -n tensorflow python=3.6_
```



```
Anaconda Prompt

(base) C:\Users\wtjoeun-IT>conda --version
conda 4.5.11

(base) C:\Users\wtjoeun-IT>conda create -n tensorflow python=3.6 anaconda
```

Anaconda 가상환경 구축 (3/6)

```
Anaconda Prompt - conda create -n tensorflow python=3.6

- python=3.6

The following packages will be downloaded:

package | build | size
-----|-----|-----
certifi-2018.10.15 | py36_0 | 138 KB
python-3.6.6 | hea74fb7_0 | 21.6 MB
wheel-0.32.2 | py36_0 | 52 KB
pip-10.0.1 | py36_0 | 1.8 MB
setuptools-40.4.3 | py36_0 | 576 KB
winertstore-0.2 | py36h7fe50ca_0 | 13 KB
-----|-----|-----
Total: | 24.1 MB

The following NEW packages will be INSTALLED:

certifi: 2018.10.15-py36_0
pip: 10.0.1-py36_0
python: 3.6.6-hea74fb7_0
setuptools: 40.4.3-py36_0
vc: 14.1-h0510ff6_4
vs2015_runtime: 14.15.26706-h3a45250_0
wheel: 0.32.2-py36_0
winertstore: 0.2-py36h7fe50ca_0

Proceed <[y]/n>? y
```

```
Anaconda Prompt

vs2015_runtime: 14.15.26706-h3a45250_0
wheel: 0.32.2-py36_0
winertstore: 0.2-py36h7fe50ca_0

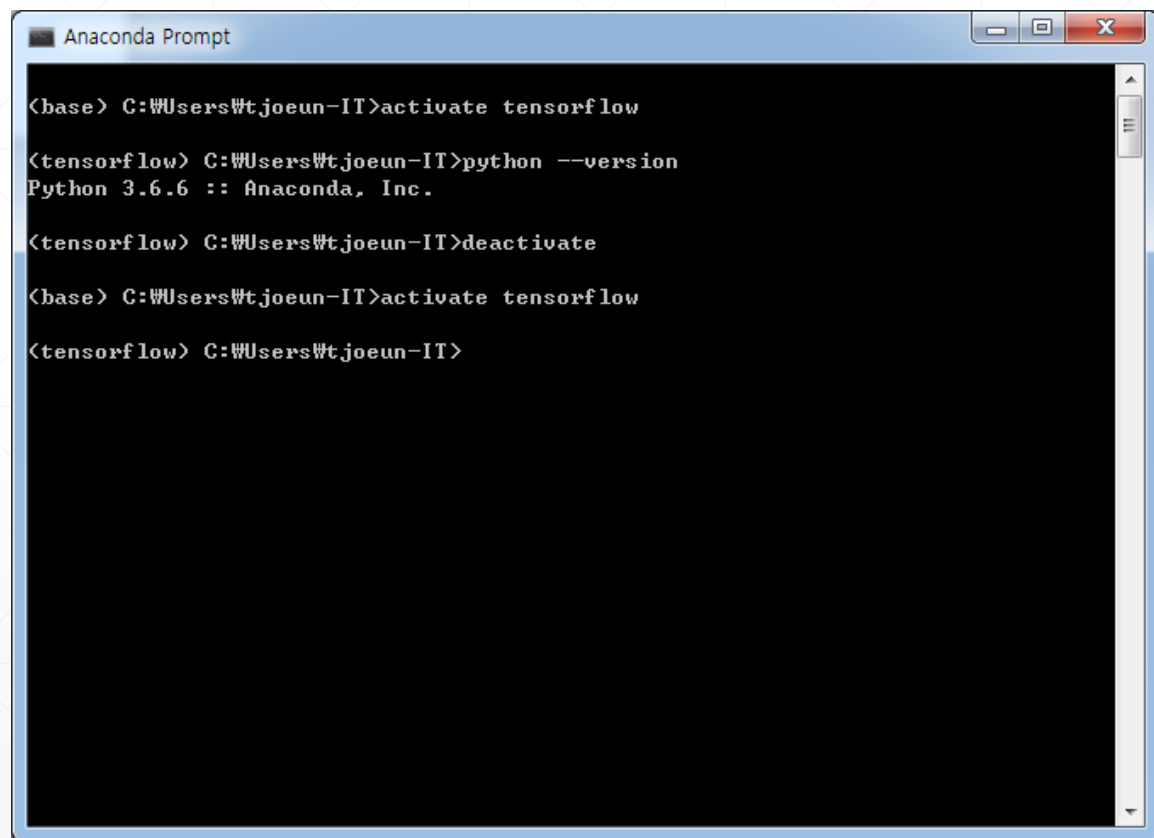
Proceed <[y]/n>? y

Downloading and Extracting Packages
certifi-2018.10.15 | 138 KB | ##### | 100%
python-3.6.6 | 21.6 MB | ##### | 100%
wheel-0.32.2 | 52 KB | ##### | 100%
pip-10.0.1 | 1.8 MB | ##### | 100%
setuptools-40.4.3 | 576 KB | ##### | 100%
winertstore-0.2 | 13 KB | ##### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done

#
# To activate this environment, use
#
# $ conda activate tensorflow
#
# To deactivate an active environment, use
#
# $ conda deactivate

(base) C:\Users\Wtjoeun-IT>
```

Anaconda 가상환경 구축 (4/6)



```
Anaconda Prompt

(base) C:\Users\Wtjoeun-IT>activate tensorflow

(tensorflow) C:\Users\Wtjoeun-IT>python --version
Python 3.6.6 :: Anaconda, Inc.

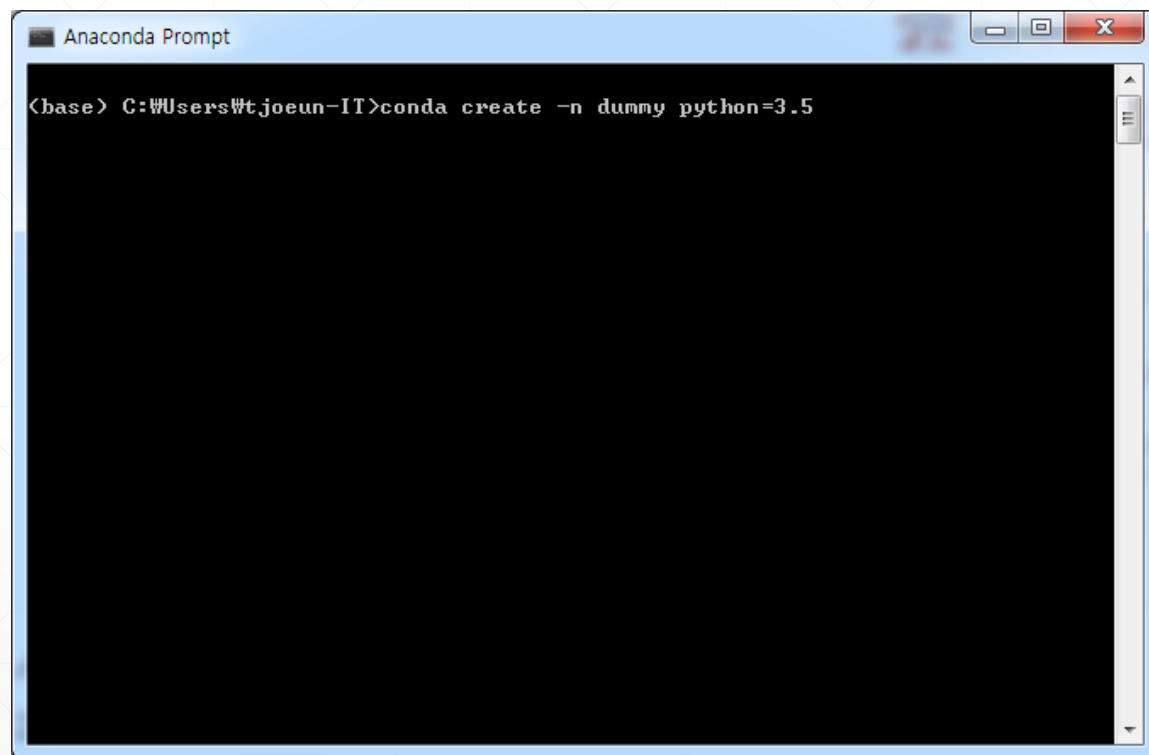
(tensorflow) C:\Users\Wtjoeun-IT>deactivate

(base) C:\Users\Wtjoeun-IT>activate tensorflow

(tensorflow) C:\Users\Wtjoeun-IT>
```

- 가상환경 활성화 방법
- conda activate 가상환경이름
- source activate 가상환경이름
- 가상환경 비활성화 방법
- conda deactivate
- source deactivate

Anaconda 가상환경 구축 (5/6)



The screenshot shows a Windows command prompt window titled "Anaconda Prompt". The prompt is at the root of the C: drive. The command entered is `conda create -n dummy python=3.5`. The prompt is currently at the end of the command line, waiting for execution.

```

C:\> conda create -n dummy python=3.5

```

- 설치된 가상환경 확인 방법
- `conda info --envs`
- 설치된 가상환경 삭제 방법
- `conda remove --name 가상환경이름 [--all]`
- `--all` 을 사용하는 경우 완전 삭제

Anaconda 가상환경 구축 (6/6)

```
Anaconda Prompt

(base) C:\Users\wtjoeun-IT>conda info --envs
# conda environments:
#
base                * C:\Users\wtjoeun-IT\Anaconda3
dummy               C:\Users\wtjoeun-IT\Anaconda3\envs\dummy
tensorflow          C:\Users\wtjoeun-IT\Anaconda3\envs\tensorflow

(base) C:\Users\wtjoeun-IT>activate dummy

(dummy) C:\Users\wtjoeun-IT>deactivate

(base) C:\Users\wtjoeun-IT>conda remove --name dummy --all_
```

```
Anaconda Prompt

(base) C:\Users\wtjoeun-IT>conda remove --name dummy --all

Remove all packages in environment C:\Users\wtjoeun-IT\Anaconda3\envs\dummy:

## Package Plan ##

  environment location: C:\Users\wtjoeun-IT\Anaconda3\envs\dummy

The following packages will be REMOVED:

certifi:          2018.8.24-py35_1
pip:              10.0.1-py35_0
python:           3.5.6-he025d50_0
setuptools:       40.2.0-py35_0
vc:               14.1-h0510ff6_4
vs2015_runtime:   14.15.26706-h3a45250_0
wheel:            0.31.1-py35_0
wincertstore:     0.2-py35hfebbdb8_0

Proceed [y]/n)? y

(base) C:\Users\wtjoeun-IT>conda info --envs
# conda environments:
#
base                * C:\Users\wtjoeun-IT\Anaconda3
tensorflow          C:\Users\wtjoeun-IT\Anaconda3\envs\tensorflow

(base) C:\Users\wtjoeun-IT>
```

TensorFlow 설치 및 확인

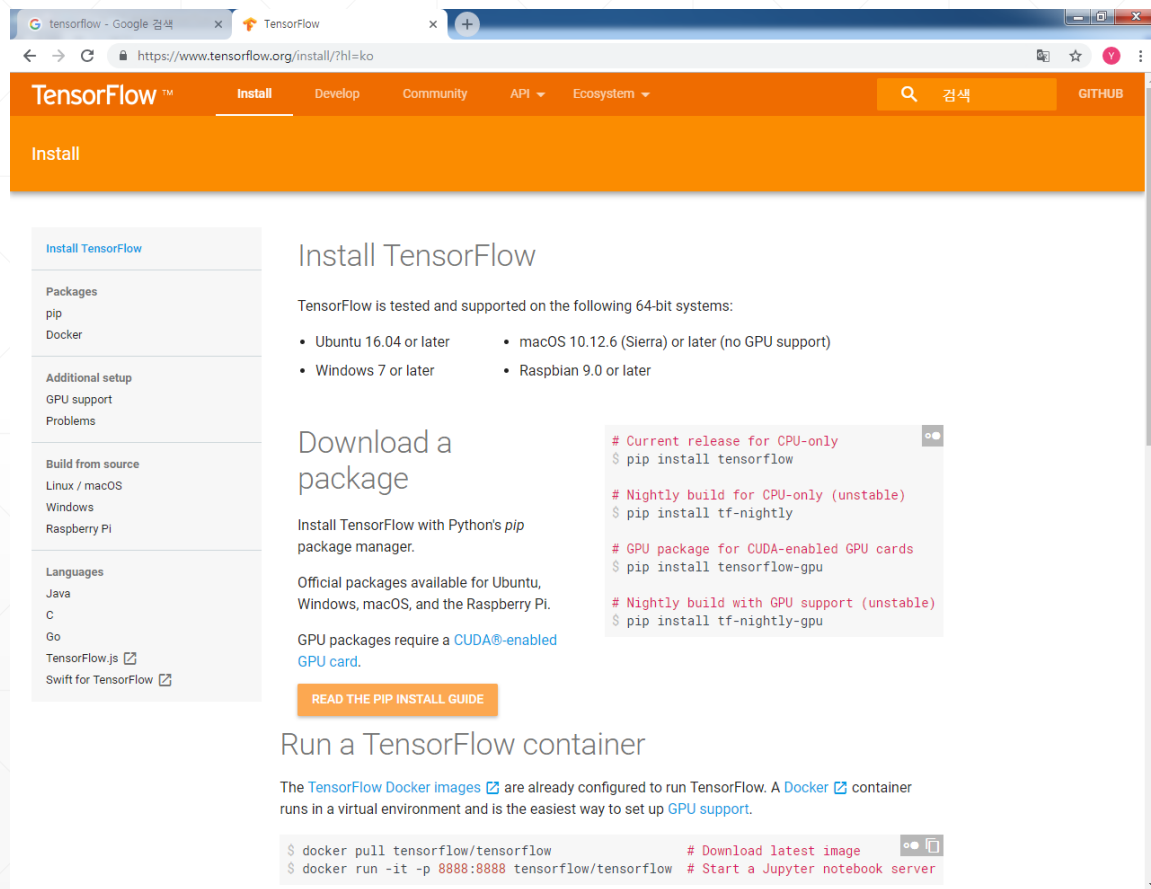
Anaconda 내부에 가상환경을 생성하여 TensorFlow 라이브러리 설치

TensorFlow 설치 (1/4)



- <https://www.tensorflow.org/>
- 딥러닝 기반의 인공지능을 구현하기 위한 라이브러리
- Python 및 Java 언어를 지원하며 추가적인 언어의 지원이 이루어지고 있음
- Python 을 사용하는 경우 Python 의 pip 명령을 통하여 설치할 수 있음
- TensorFlow 홈페이지에서 설치 방법을 확인할 수 있음

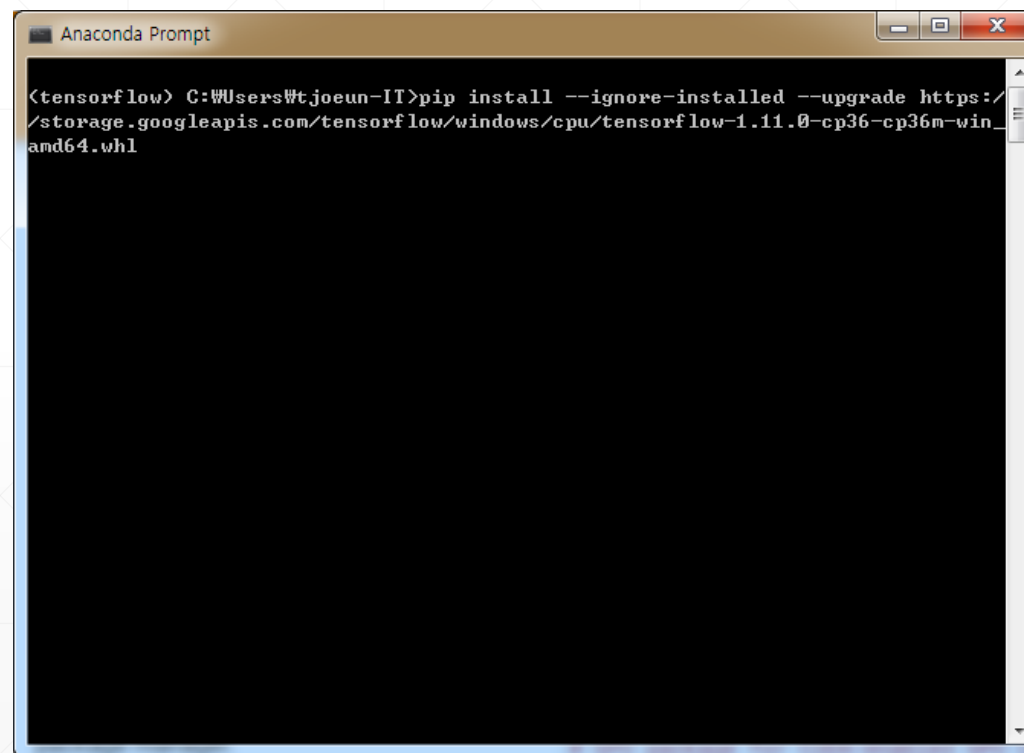
TensorFlow 설치 (2/4)



- TensorFlow의 설치 방법은 수시로 변경될 수 있기 때문에 새로운 버전이 출시되면 반드시 확인이 필요함

TensorFlow 설치 (3/4)

- Anaconda 가상환경 활성화
- TensorFlow 홈페이지에 소개된 설치 방법을 참고하여 설치를 진행



```
Anaconda Prompt

(tensorflow) C:\Users\wtjoeun-IT>pip install --ignore-installed --upgrade https://storage.googleapis.com/tensorflow/windows/cpu/tensorflow-1.11.0-cp36-cp36m-win-amd64.whl
```

TensorFlow 설치 (4/4)

```
Anaconda Prompt
Collecting h5py <from keras-applications>=1.0.5->tensorflow==1.11.0>
  Using cached https://files.pythonhosted.org/packages/12/6c/00c38c5ce9322f1cc42
1d93217c44739646a106c61859622eccc297a5c05/h5py-2.8.0-cp36-cp36m-win_amd64.whl
Collecting werkzeug>=0.11.10 <from tensorboard<1.12.0,>=1.11.0->tensorflow==1.11
.0>
  Using cached https://files.pythonhosted.org/packages/20/c4/12e3e56473e52375aa2
9c4764e70d1b8f3efa6682bef8d0aae04fe335243/Werkzeug-0.14.1-py2.py3-none-any.whl
Collecting markdown>=2.6.8 <from tensorboard<1.12.0,>=1.11.0->tensorflow==1.11.0
>
  Using cached https://files.pythonhosted.org/packages/7a/6b/5600647404ba15545ec
37d2f7f58844d690baf2f81f3a60b862e48f29287/Markdown-3.0.1-py2.py3-none-any.whl
Building wheels for collected packages: absl-py
  Running setup.py bdist_wheel for absl-py ... done
  Stored in directory: C:\Users\Wtjoeun-IT\AppData\Local\Pip\Cache\wheels\W91W3fWf
cW04598e99d64928bec676e07245b132e985df067cf5d469c460
Successfully built absl-py
Installing collected packages: numpy, six, h5py, keras-applications, setuptools,
protobuf, termcolor, astor, gast, wheel, grpcio, absl-py, werkzeug, markdown, t
ensorboard, keras-preprocessing, tensorflow
Successfully installed absl-py-0.6.0 astor-0.7.1 gast-0.2.0 grpcio-1.16.0 h5py-2
.8.0 keras-applications-1.0.6 keras-preprocessing-1.0.5 markdown-3.0.1 numpy-1.1
5.3 protobuf-3.6.1 setuptools-40.4.3 six-1.11.0 tensorboard-1.11.0 tensorflow-1.
11.0 termcolor-1.1.0 werkzeug-0.14.1 wheel-0.32.2
You are using pip version 10.0.1, however version 18.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' comm
and.

<tensorflow> C:\Users\Wtjoeun-IT>
```

```
Anaconda Prompt

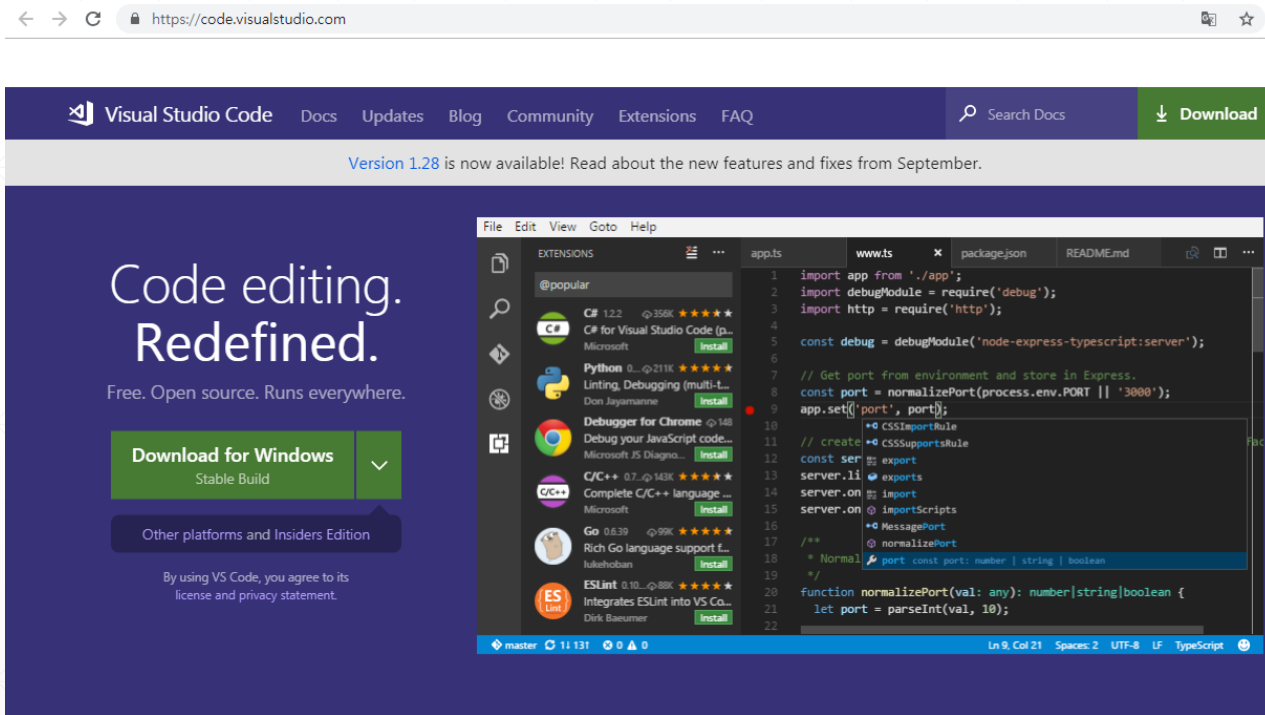
<tensorflow> C:\Users\Wtjoeun-IT>python
Python 3.6.6 |Anaconda, Inc.| <default, Jun 28 2018, 11:27:44> [MSC v.1900 64 bi
t (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow as tf
>>> exit()

<tensorflow> C:\Users\Wtjoeun-IT>
```

Python 개발도구

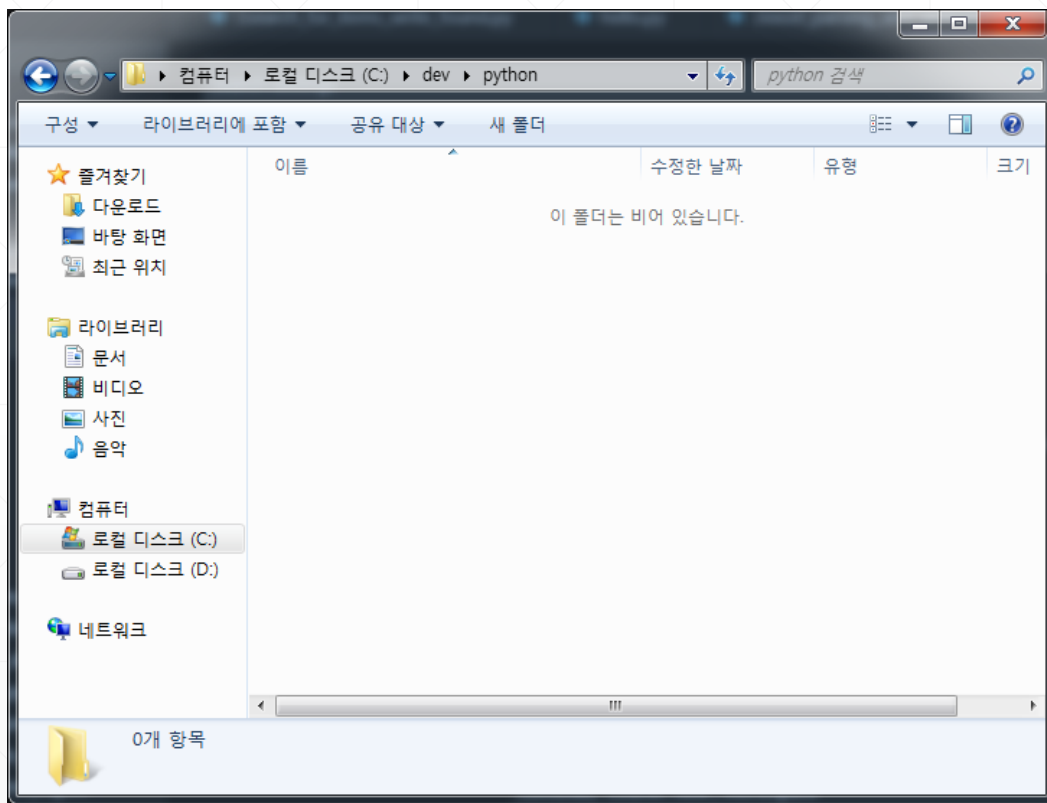
Visual Studio Code 설치와 사용법

Visual Studio Code 설치 및 사용법 (1/22)



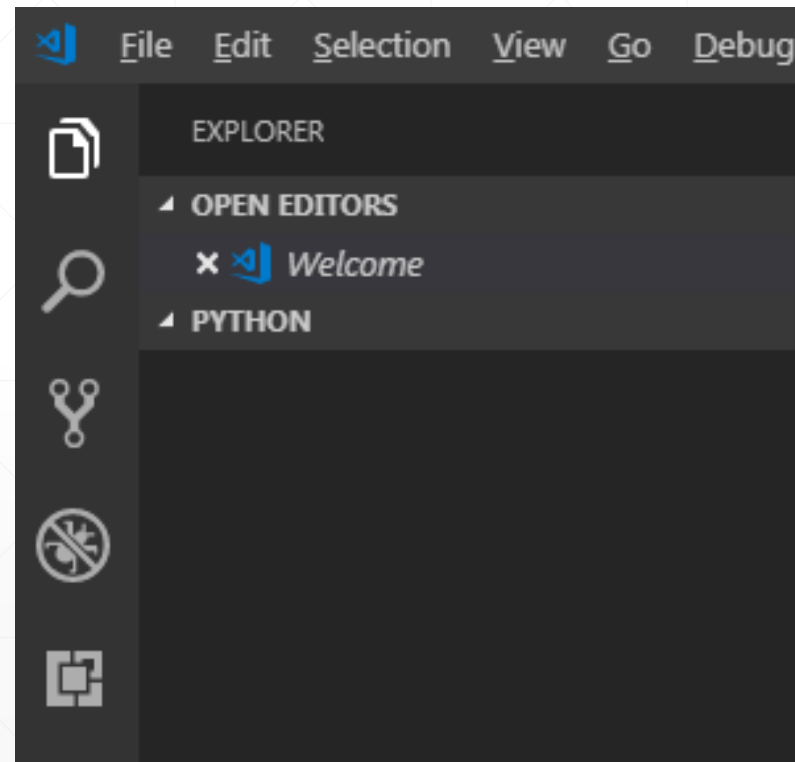
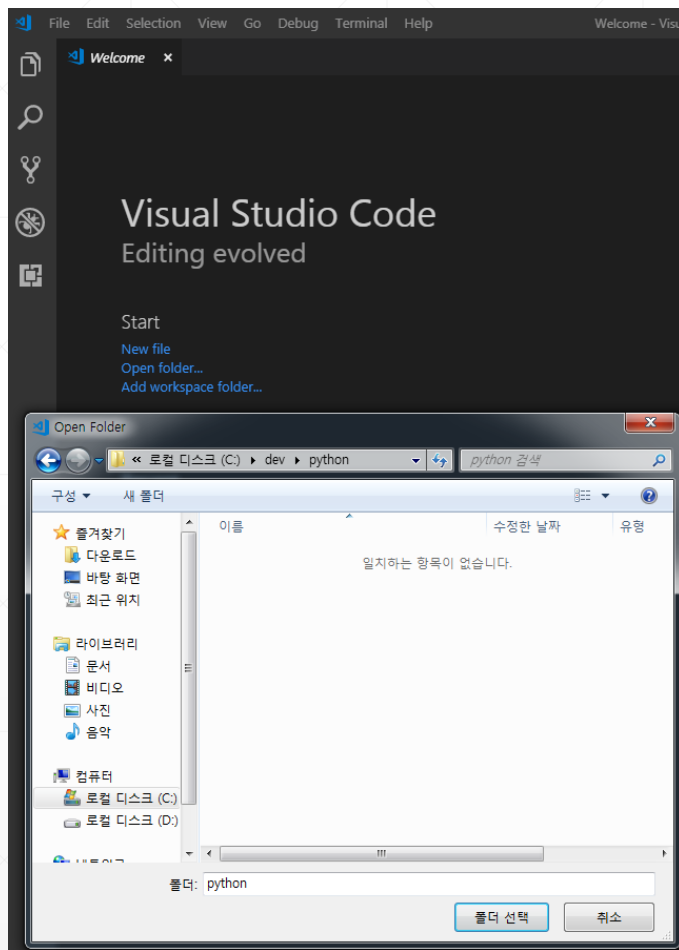
- <https://code.visualstudio.com/>
- Microsoft에서 제공하는 프로그램 개발 도구
- Python 뿐만 아니라 C C++, C# 등과 같은 언어의 개발을 지원
- 크로스 플랫폼을 지향하여, 윈도우, 리눅스, 맥OS 등에서 동일한 환경으로 개발이 가능
- 설치 과정은 기본으로 설치

Visual Studio Code 설치 및 사용법 (2/22)

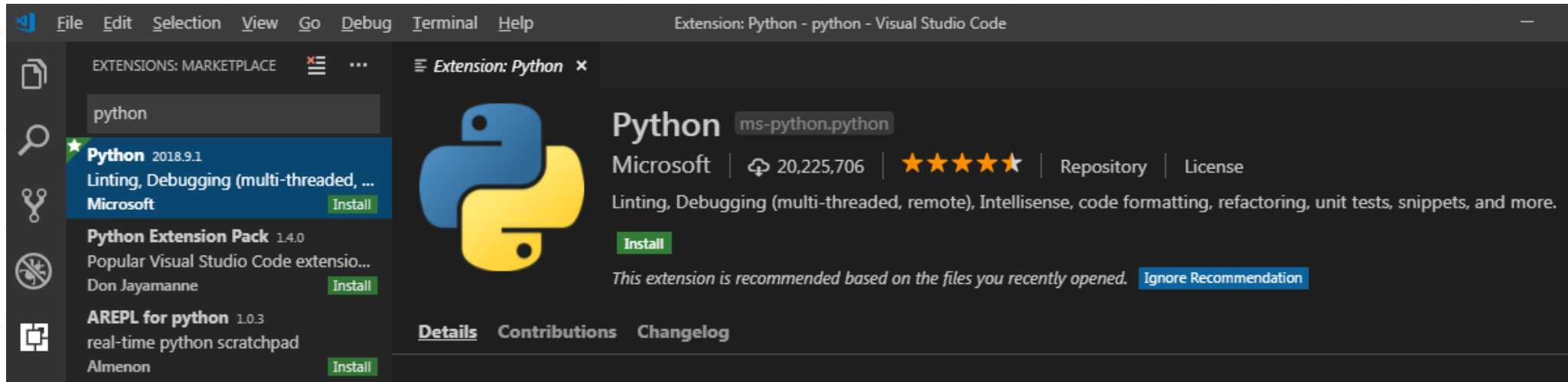


- Visual Studio Code(vscode)는 폴더를 기준으로 개발할 수 있음
- 기본 경로로 C:\dev\python을 사용

Visual Studio Code 설치 및 사용법 (3/22)

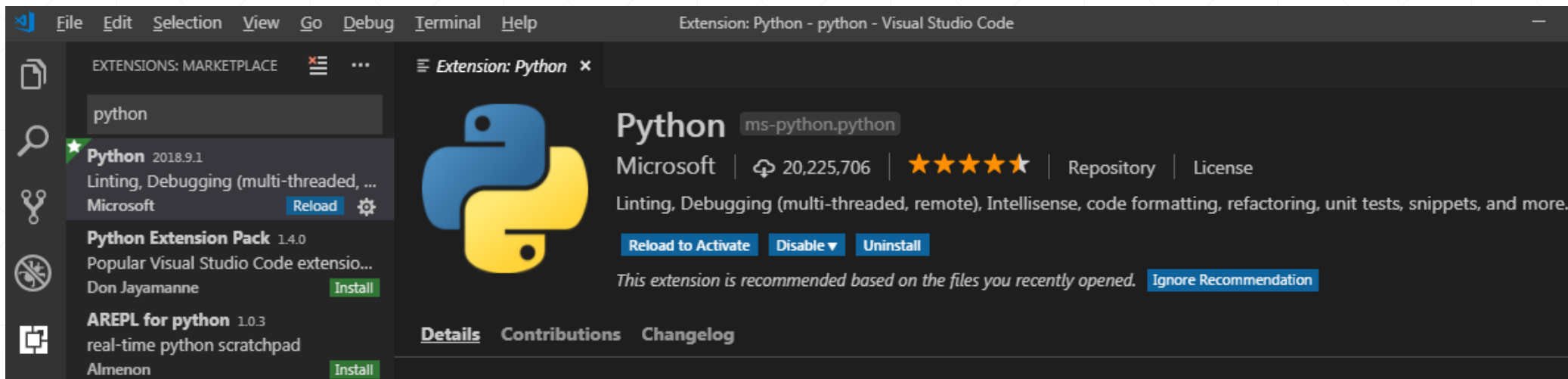


Visual Studio Code 설치 및 사용법 (4/22)



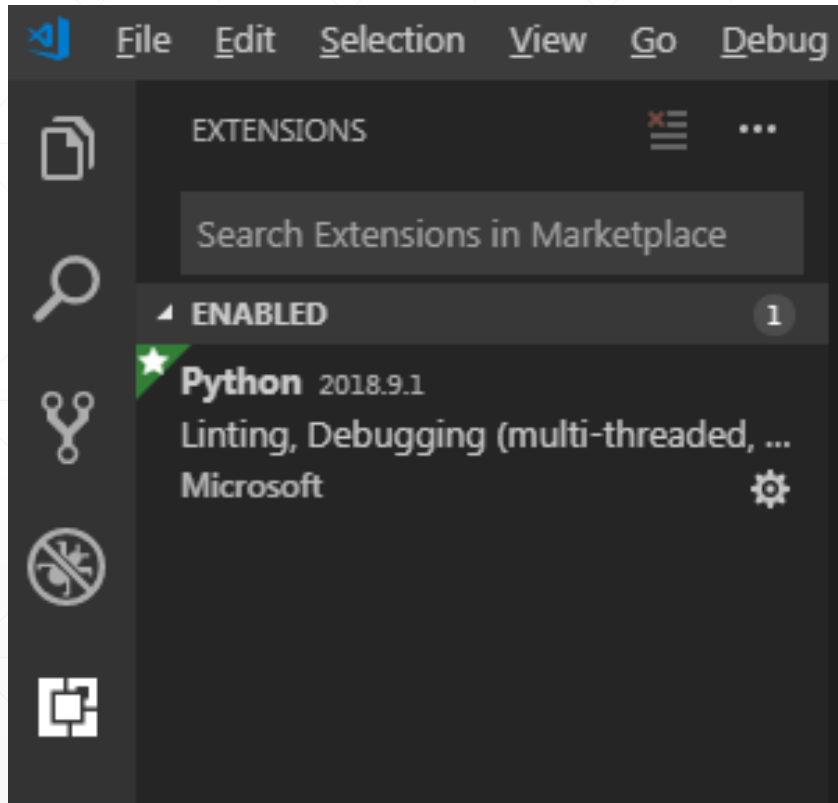
- Python을 개발하기 위한 Python 라이브러리(Extension)를 설치
- 개발 도구 왼쪽 가장 하단의 아이콘에서 확인이 가능함
- Ctrl + Shift + X 단축키로 접근 가능
- 검색 창에 python을 입력 후, 검색 결과에서 Python을 install 함

Visual Studio Code 설치 및 사용법 (5/22)



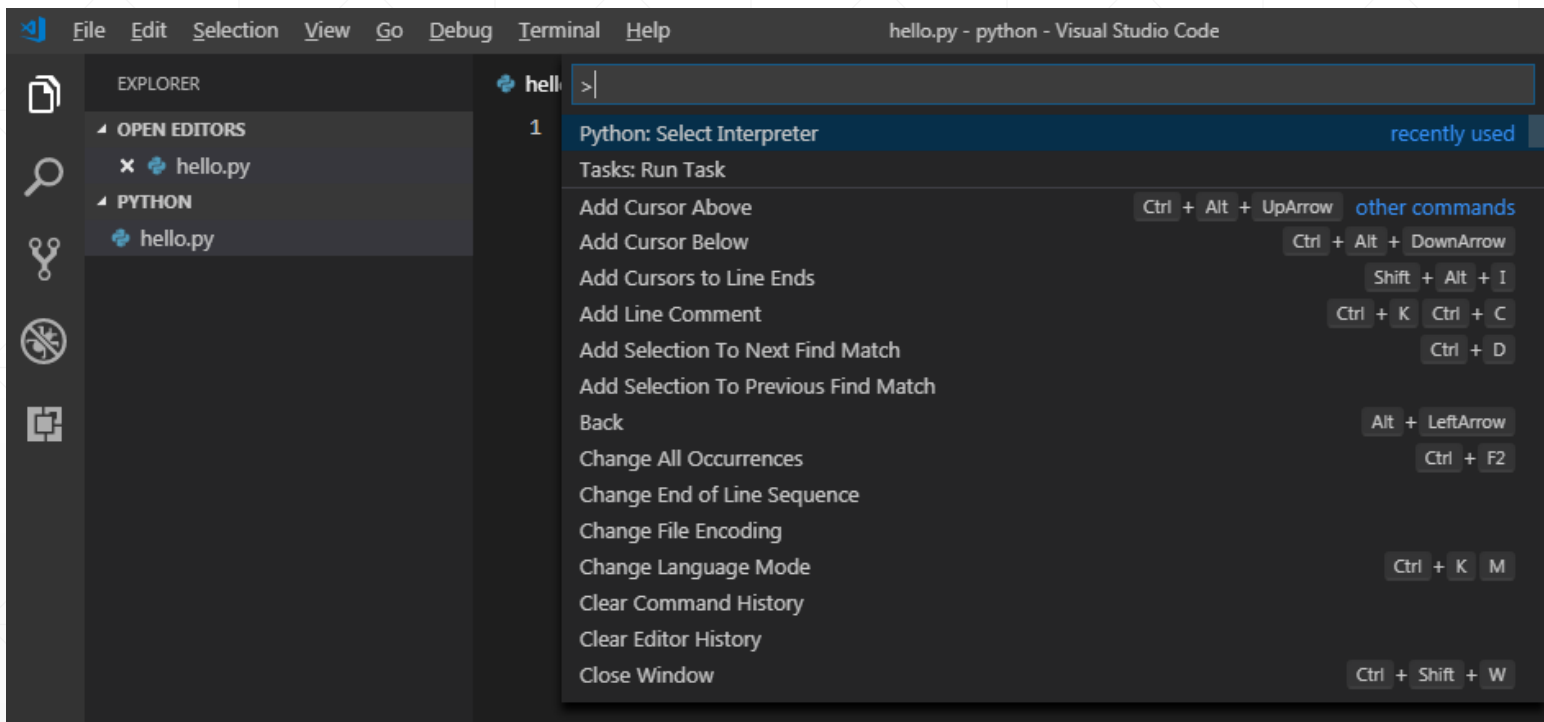
- 검색 창에 python을 입력 후, 검색 결과에서 Python을 install 함
- Install 과정이 종료된 후, Reload to Activate 버튼을 클릭하여 Visual Studio Code 가 다시 시작되는 것을 확인

Visual Studio Code 설치 및 사용법 (6/22)



- Visual Studio Code에 설치된 라이브러리(Extension) 확인
- 개발 도구 왼쪽 가장 하단의 아이콘에서 확인이 가능함
- Ctrl + Shift + X 단축키로 접근 가능

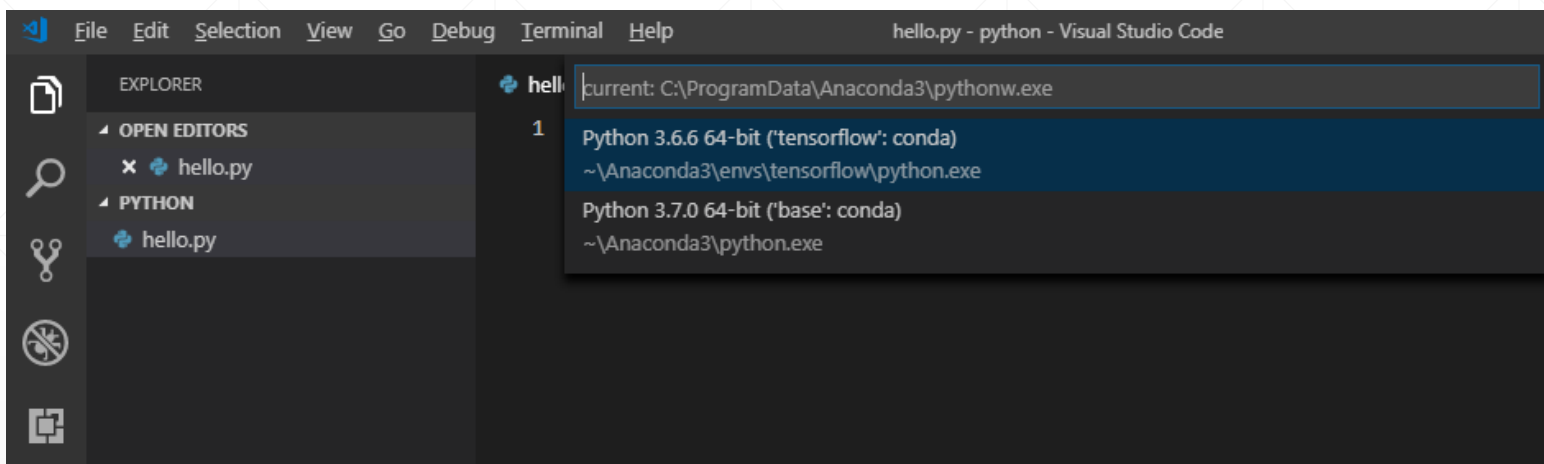
Visual Studio Code 설치 및 사용법 (7/22)



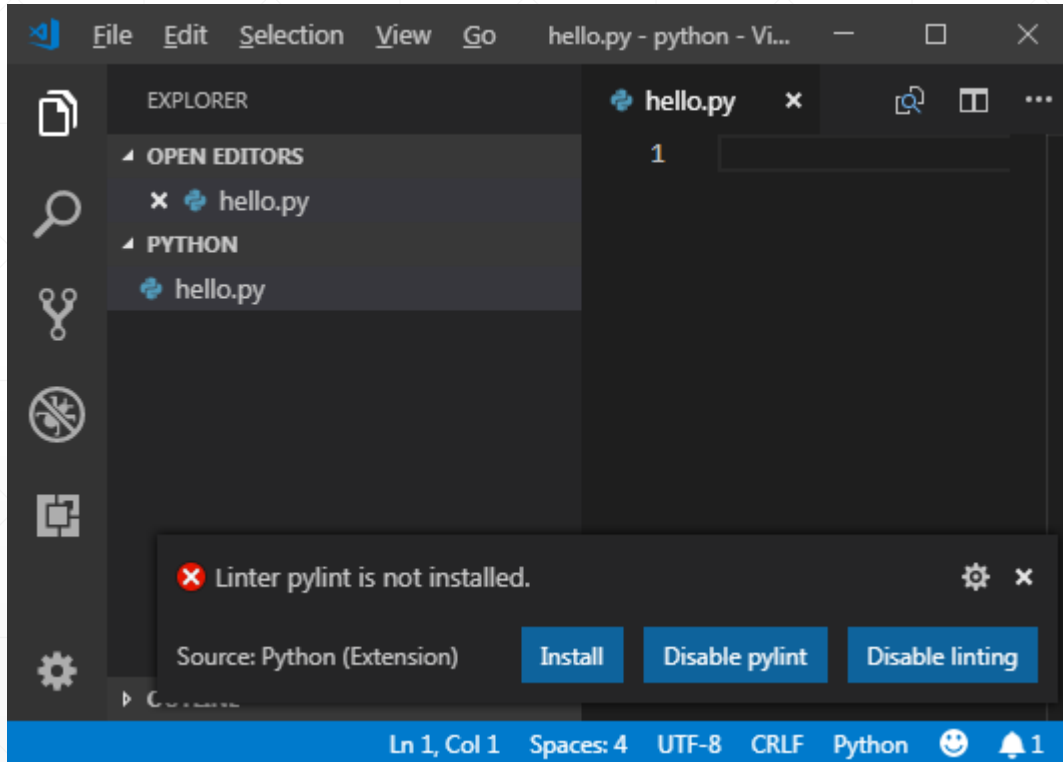
- Python 실행 환경 선택
- Ctrl + Shift + P 입력 후, Python: Select Interpreter 입력

Visual Studio Code 설치 및 사용법 (8/22)

- Python 실행 환경 선택
- 실행에 사용한 Python 버전을 선택
- Anaconda에 설치한 가상 환경도 선택이 가능함

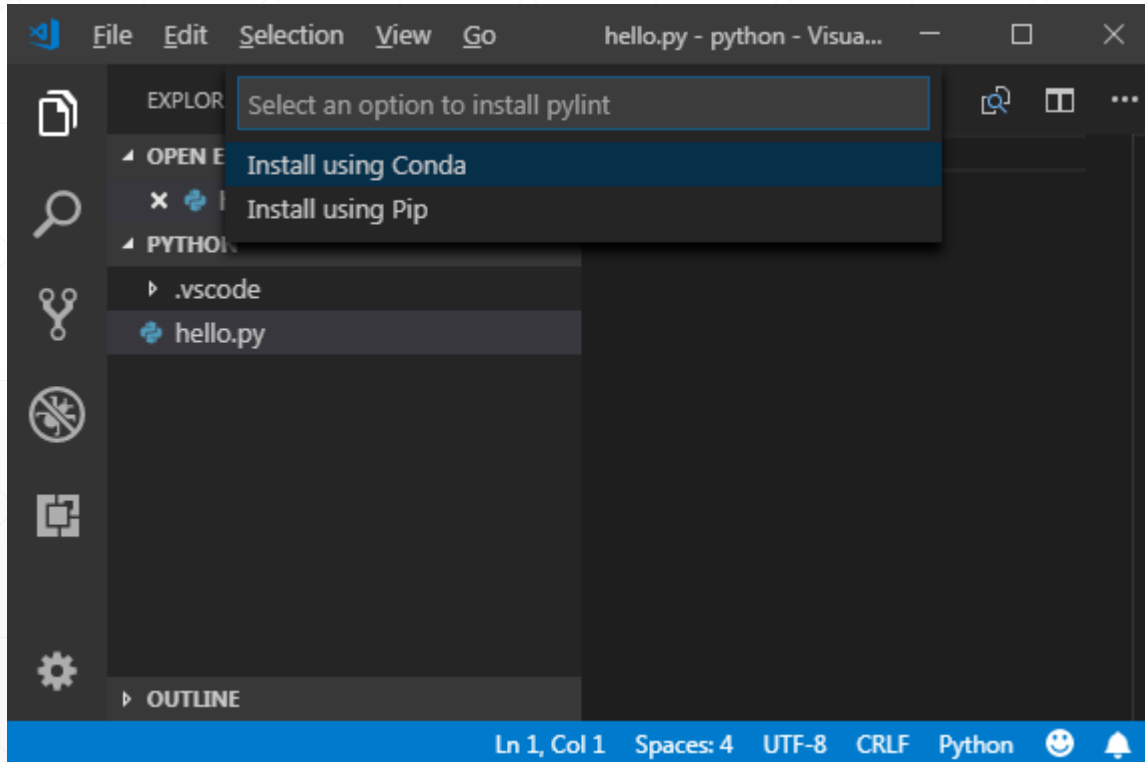


Visual Studio Code 설치 및 사용법 (9/22)



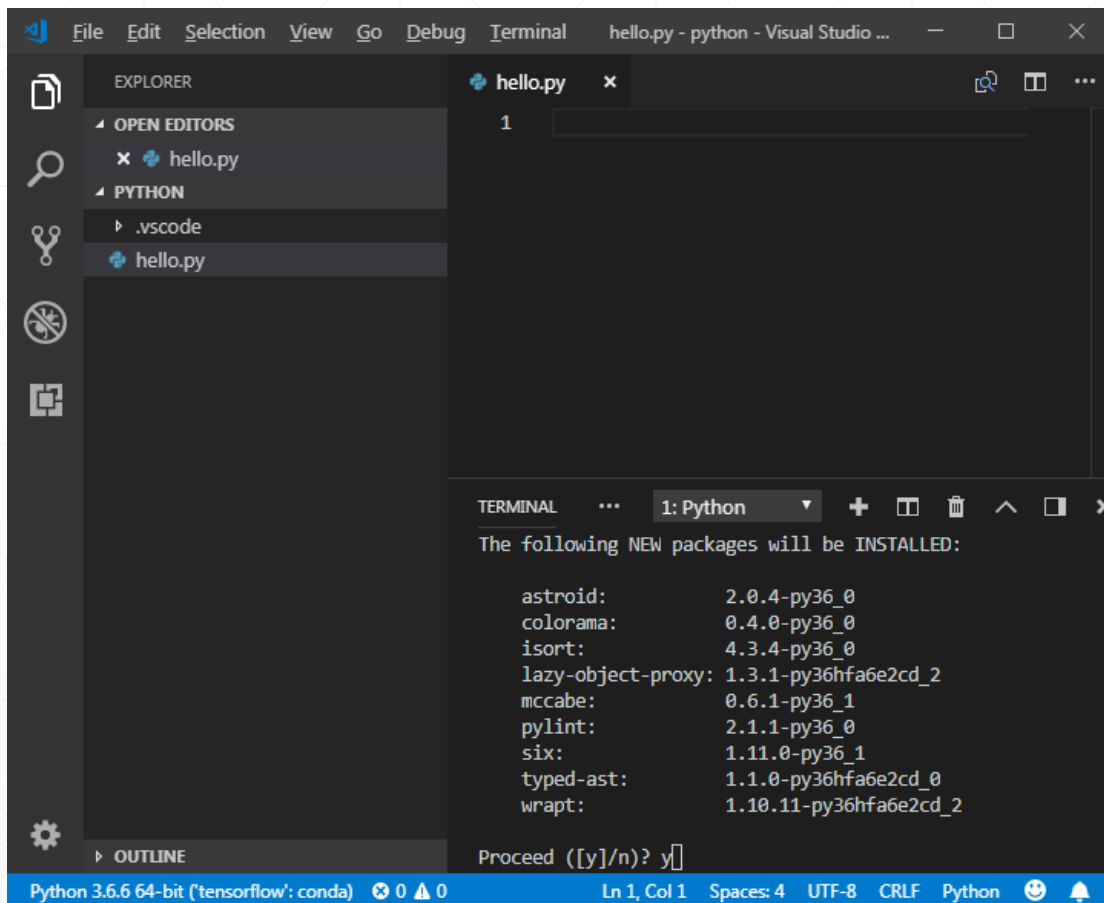
- Python 실행 환경을 선택하면 pylint 라이브러리를 설치할 수 있음

Visual Studio Code 설치 및 사용법 (10/22)



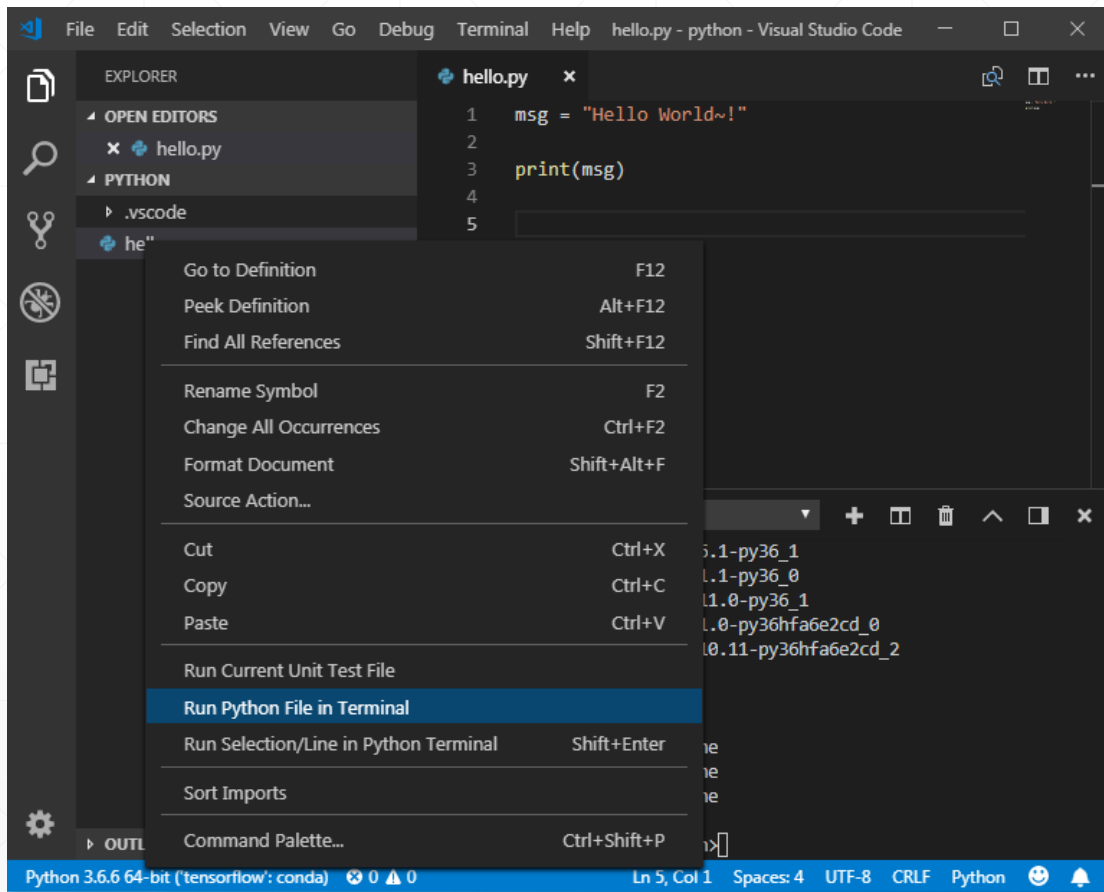
- Python 실행 환경을 선택하면 pylint 라이브러리를 설치할 수 있음

Visual Studio Code 설치 및 사용법 (11/22)



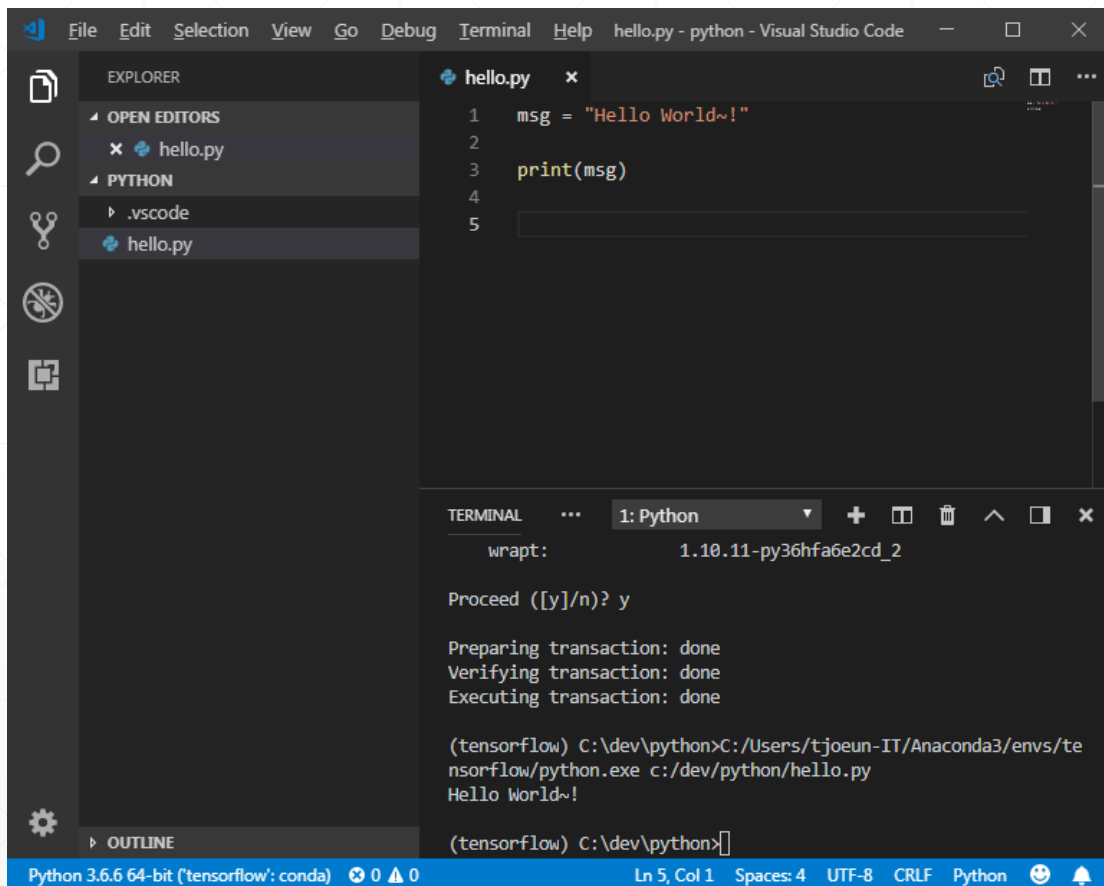
- Python 실행 환경을 선택하면 pylint 라이브러리를 설치할 수 있음

Visual Studio Code 설치 및 사용법 (12/22)



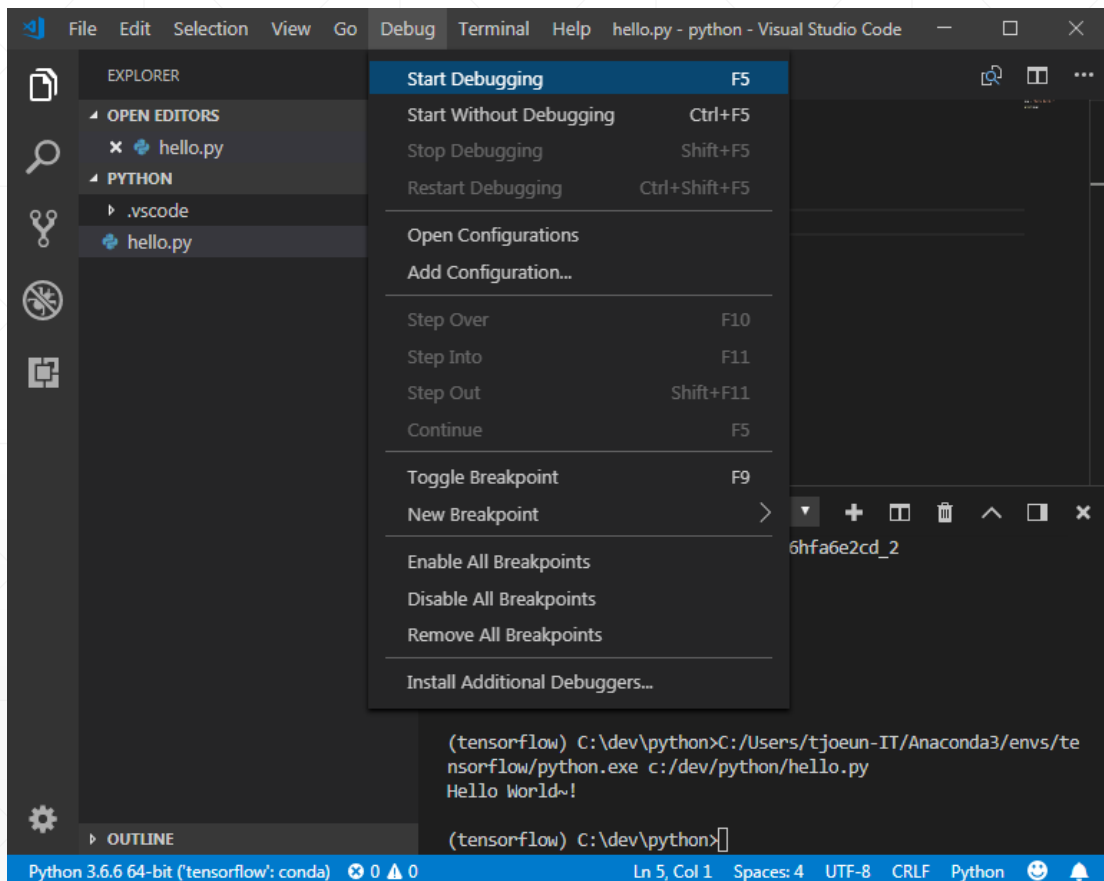
- hello.py 파일을 추가한 후 왼쪽의 이미지에 나타난 코드를 입력
- 소스 코드에서 마우스 오른쪽 버튼을 클릭하면 실행 메뉴가 나타남

Visual Studio Code 설치 및 사용법 (13/22)



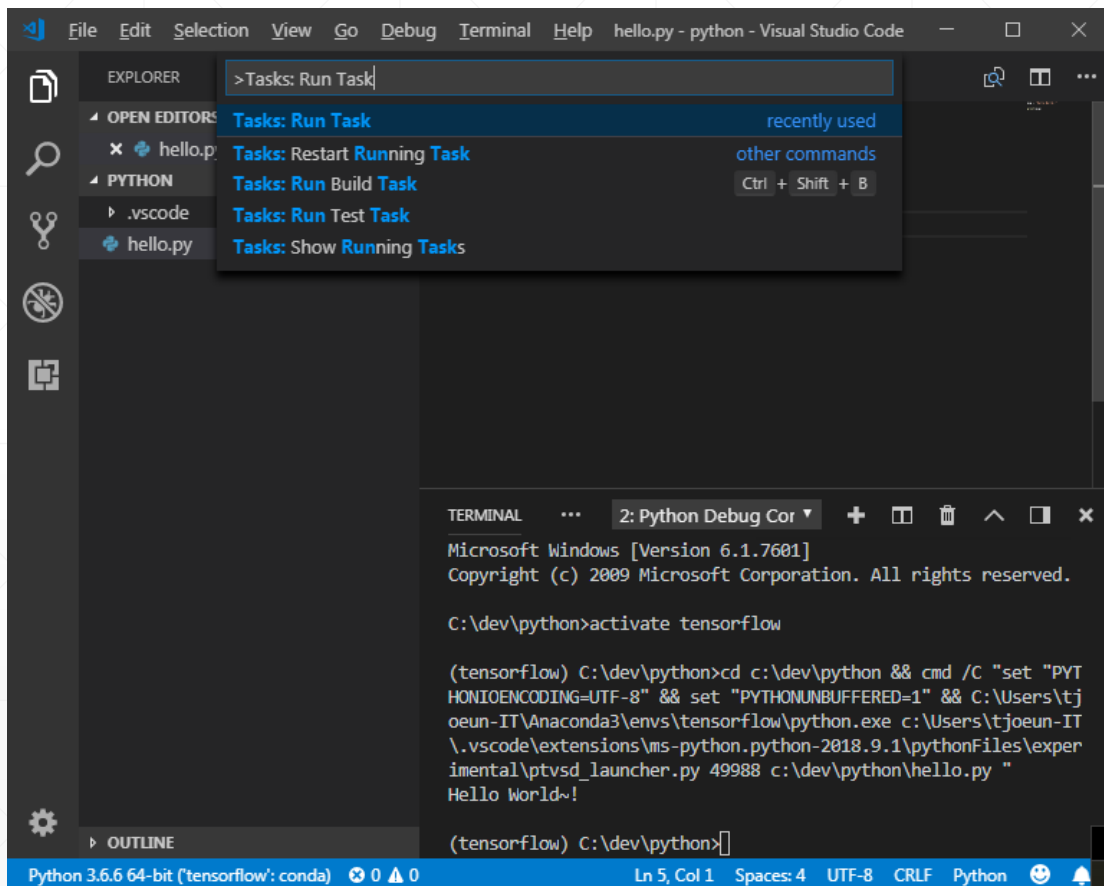
- hello.py 파일을 추가한 후 왼쪽의 이미지에 나타난 코드를 입력
- 소스 코드에서 마우스 오른쪽 버튼을 클릭하면 실행 메뉴가 나타남

Visual Studio Code 설치 및 사용법 (14/22)



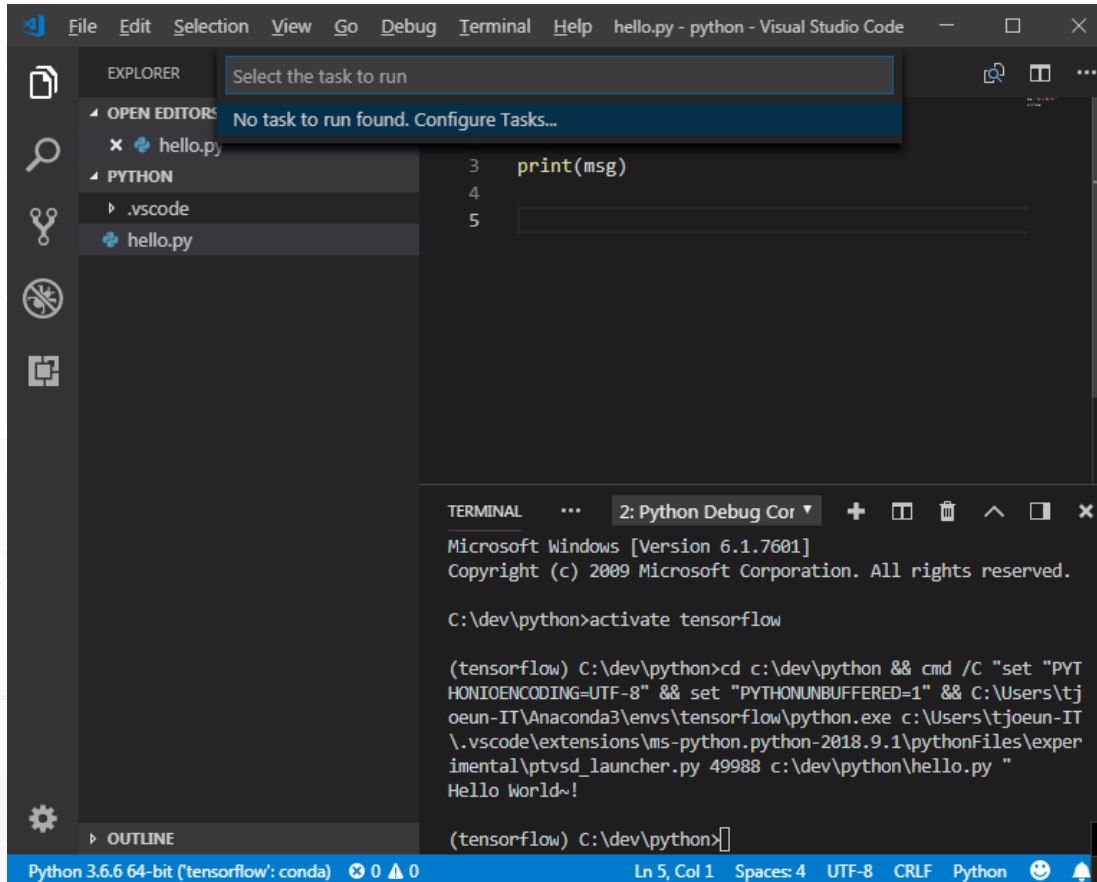
- Debugging(디버깅) 실행
- Debug 메뉴에서 Start Debugging 메뉴를 선택하여 실행
- 브레이크 포인트와 같은 요소를 사용하여 디버깅할 수 있음

Visual Studio Code 설치 및 사용법 (15/22)



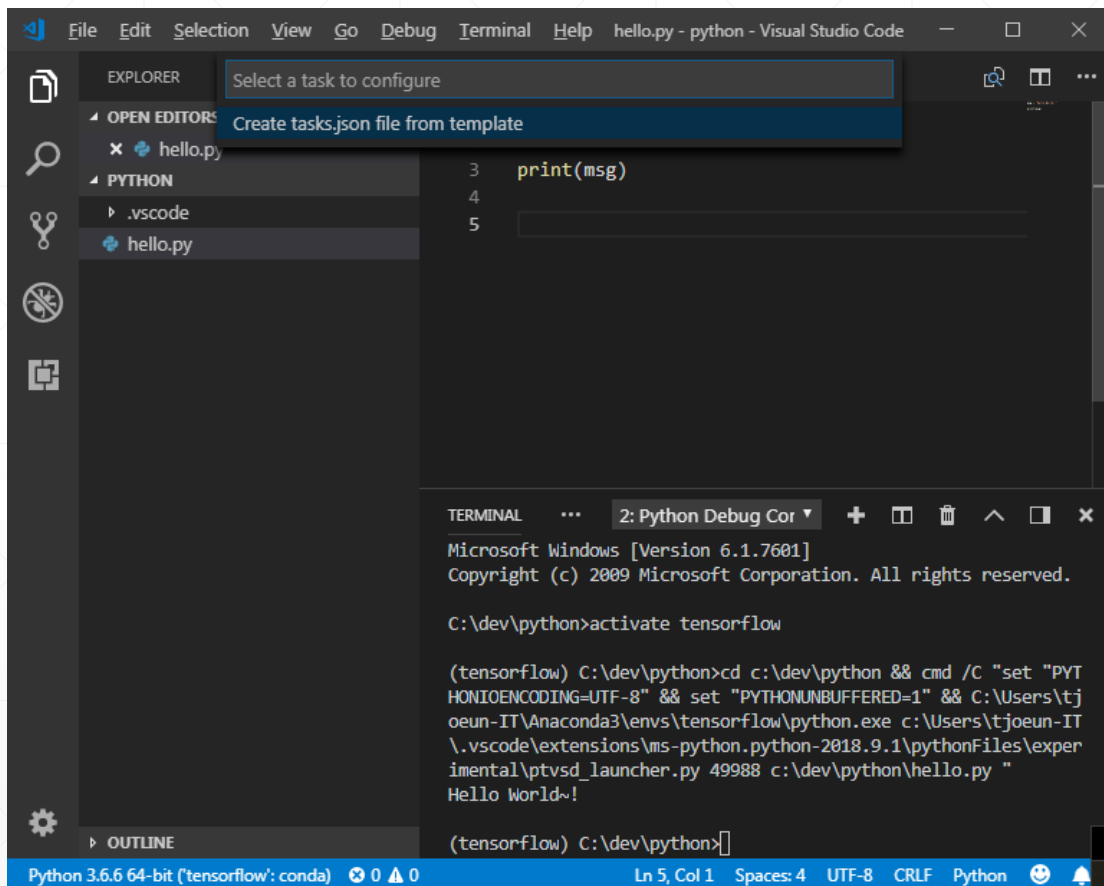
- Task를 활용한 실행 방법
- 소스 코드의 빌드 설정을 직접 작성하여 실행하는 방법
- 실행에 필요한 매개변수를 포함시켜 실행할 수 있음
- Ctrl + Shift + P 입력 후, Tasks: Run Task 입력

Visual Studio Code 설치 및 사용법 (16/22)



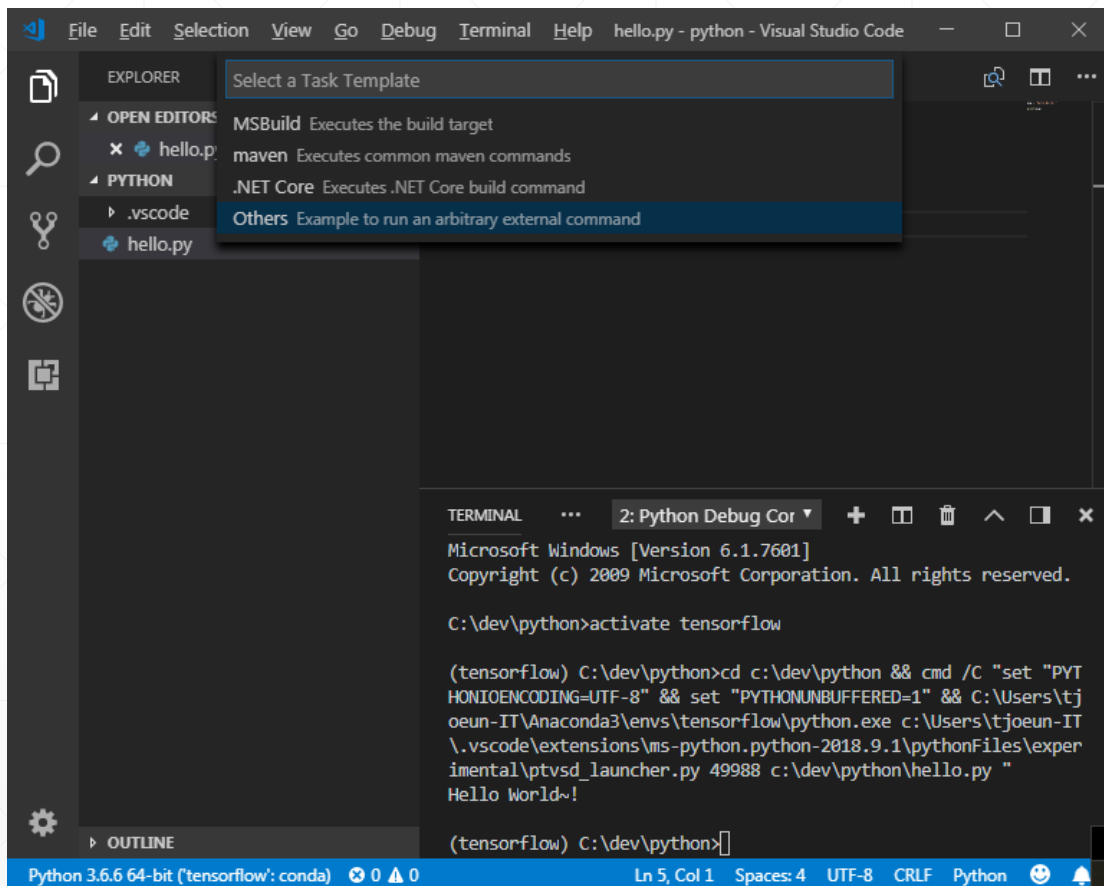
- Task를 활용한 실행 방법
- 소스 코드의 빌드 설정을 직접 작성하여 실행하는 방법
- 실행에 필요한 매개변수를 포함시켜 실행할 수 있음
- Ctrl + Shift + P 입력 후, Tasks: Run Task 입력

Visual Studio Code 설치 및 사용법 (17/22)



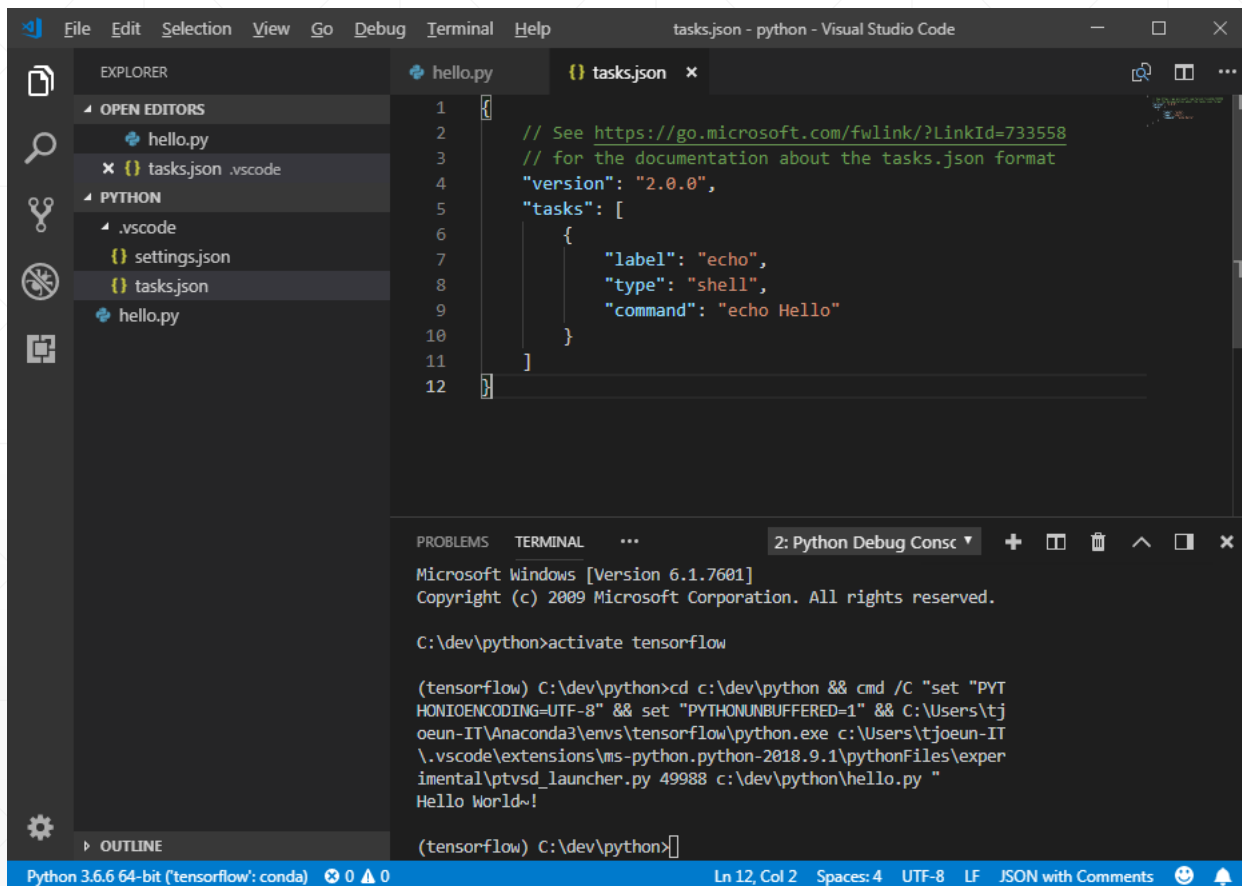
- Task를 활용한 실행 방법
- 소스 코드의 빌드 설정을 직접 작성하여 실행하는 방법
- 실행에 필요한 매개변수를 포함시켜 실행할 수 있음
- Ctrl + Shift + P 입력 후, Tasks: Run Task 입력

Visual Studio Code 설치 및 사용법 (18/22)



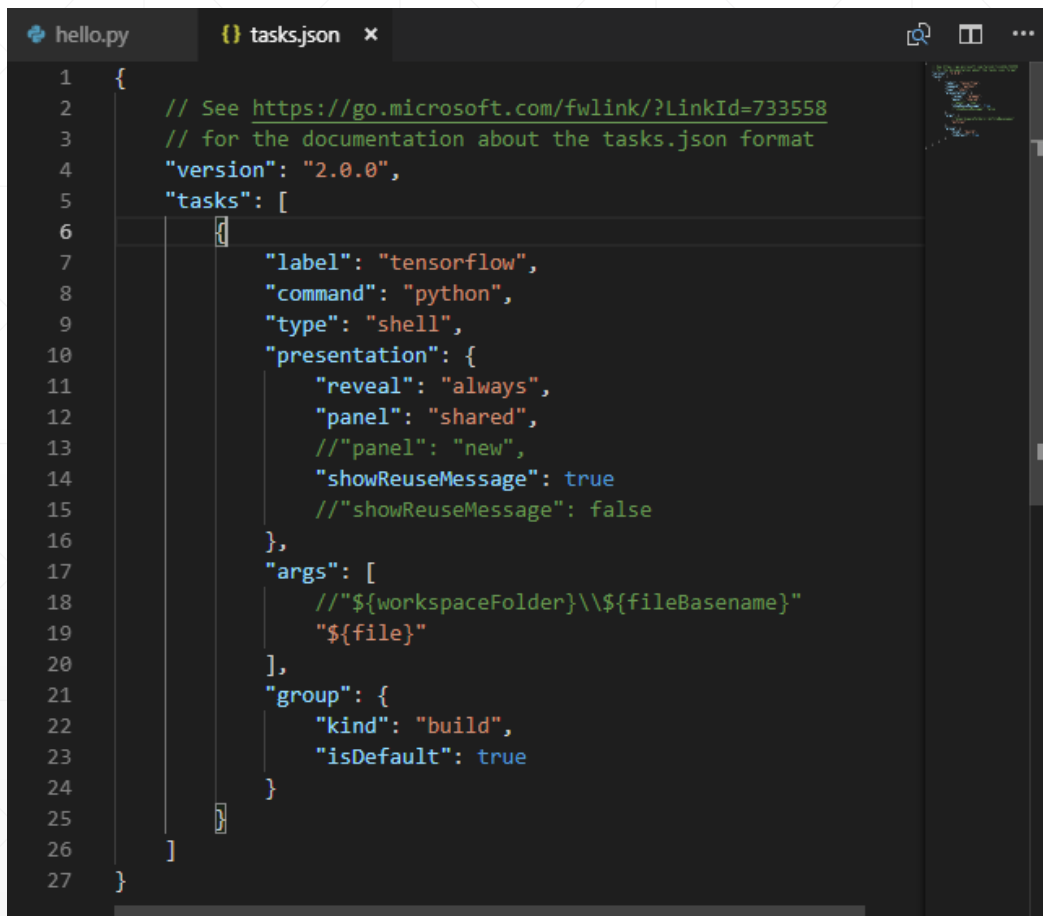
- Task를 활용한 실행 방법
- 소스 코드의 빌드 설정을 직접 작성하여 실행하는 방법
- 실행에 필요한 매개변수를 포함시켜 실행할 수 있음
- Ctrl + Shift + P 입력 후, Tasks: Run Task 입력

Visual Studio Code 설치 및 사용법 (19/22)



- Ctrl + Shift + P 입력 후, Tasks: Run Task 입력
- Task의 생성이 끝나면, tasks.json 파일이 생성됨

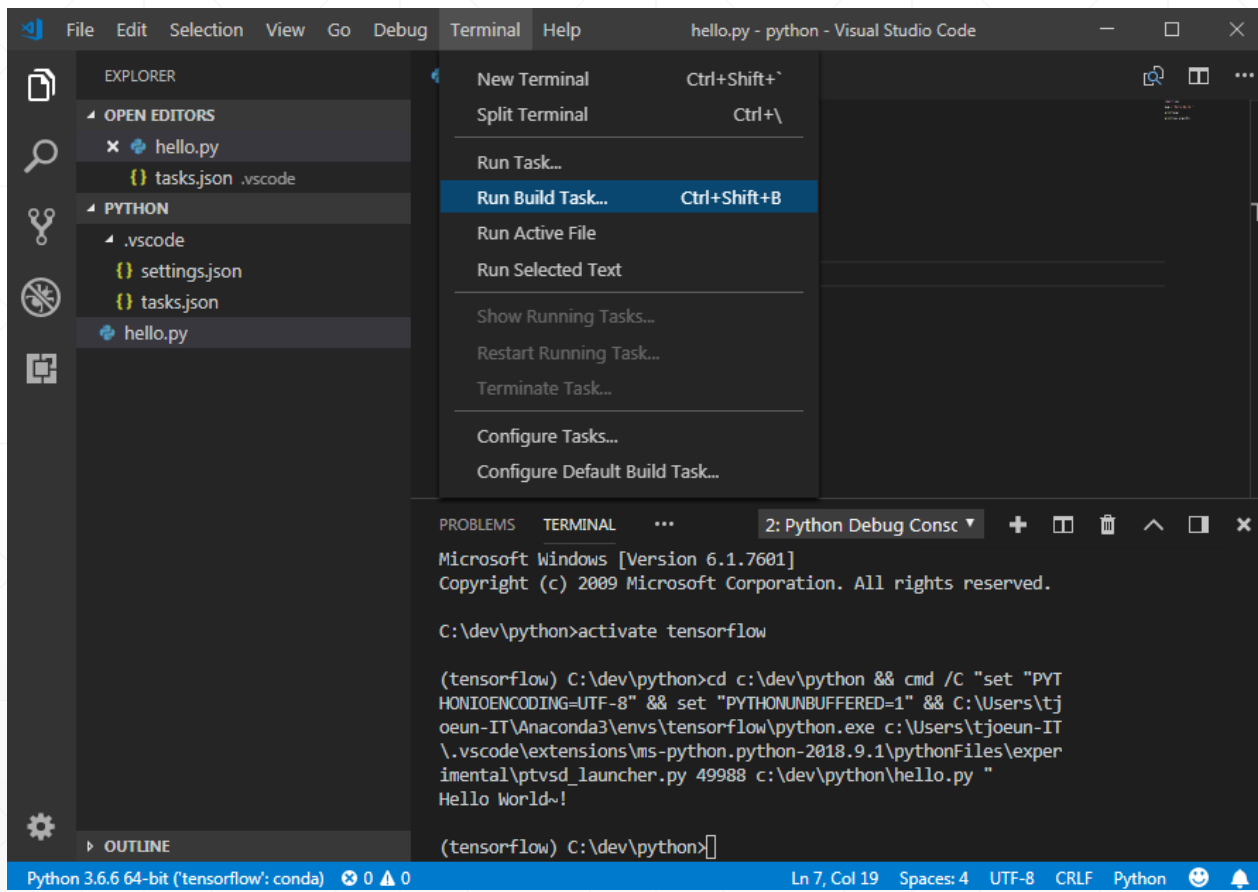
Visual Studio Code 설치 및 사용법 (20/22)

A screenshot of the Visual Studio Code editor interface. The top bar shows two tabs: 'hello.py' and 'tasks.json'. The 'tasks.json' tab is active, displaying a JSON configuration for a task. The code is as follows:

```
1 {  
2   // See https://go.microsoft.com/fwlink/?LinkId=733558  
3   // for the documentation about the tasks.json format  
4   "version": "2.0.0",  
5   "tasks": [  
6     {  
7       "label": "tensorflow",  
8       "command": "python",  
9       "type": "shell",  
10      "presentation": {  
11        "reveal": "always",  
12        "panel": "shared",  
13        // "panel": "new",  
14        "showReuseMessage": true  
15        // "showReuseMessage": false  
16      },  
17      "args": [  
18        // "${workspaceFolder}\\${fileBaseName}"  
19        "${file}"  
20      ],  
21      "group": {  
22        "kind": "build",  
23        "isDefault": true  
24      }  
25    }  
26  ]  
27 }
```

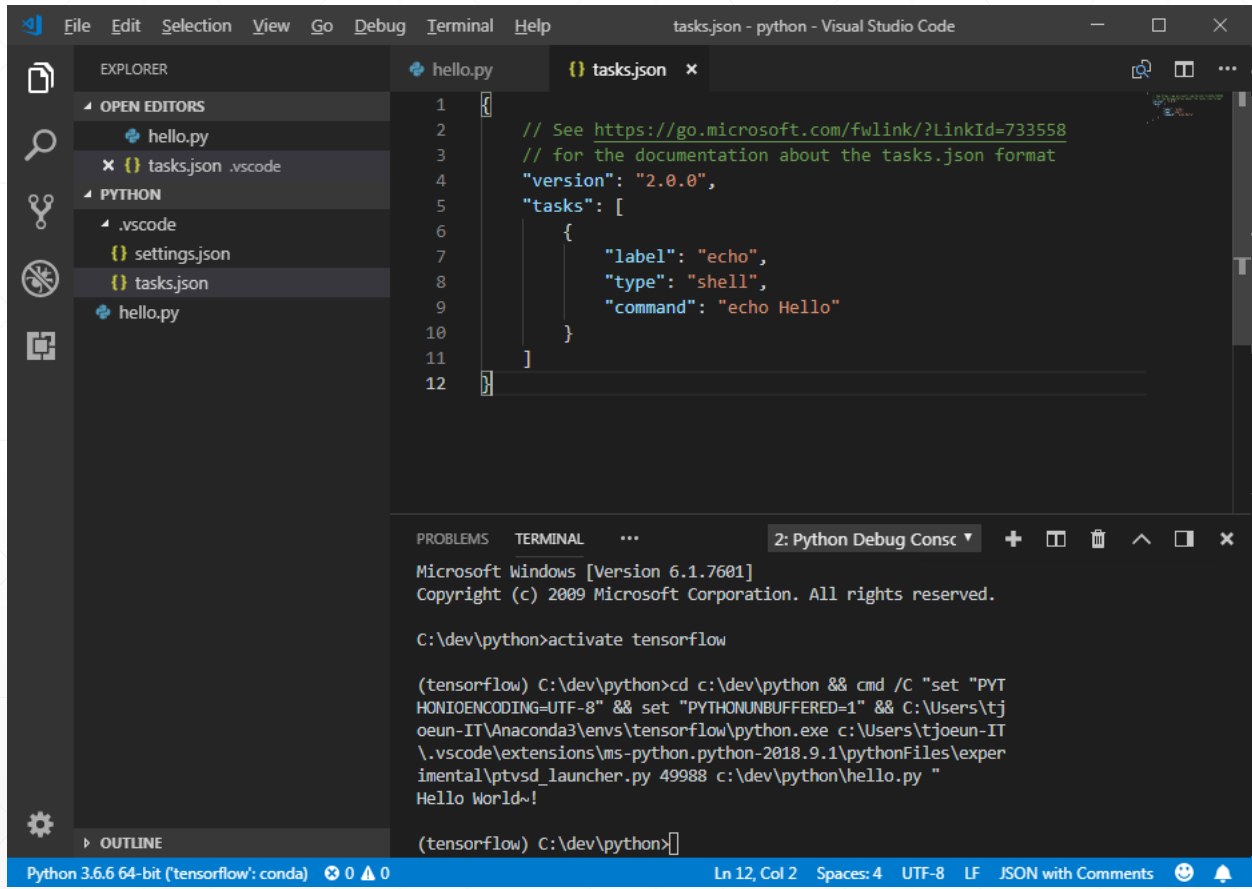
- tasks.json 파일에 왼쪽 이미지의 내용을 입력한 후 저장

Visual Studio Code 설치 및 사용법 (21/22)



- Terminal 메뉴에서 Run Build Task를 실행하여 현재 Python 문서를 실행할 수 있음

Visual Studio Code 설치 및 사용법 (22/22)

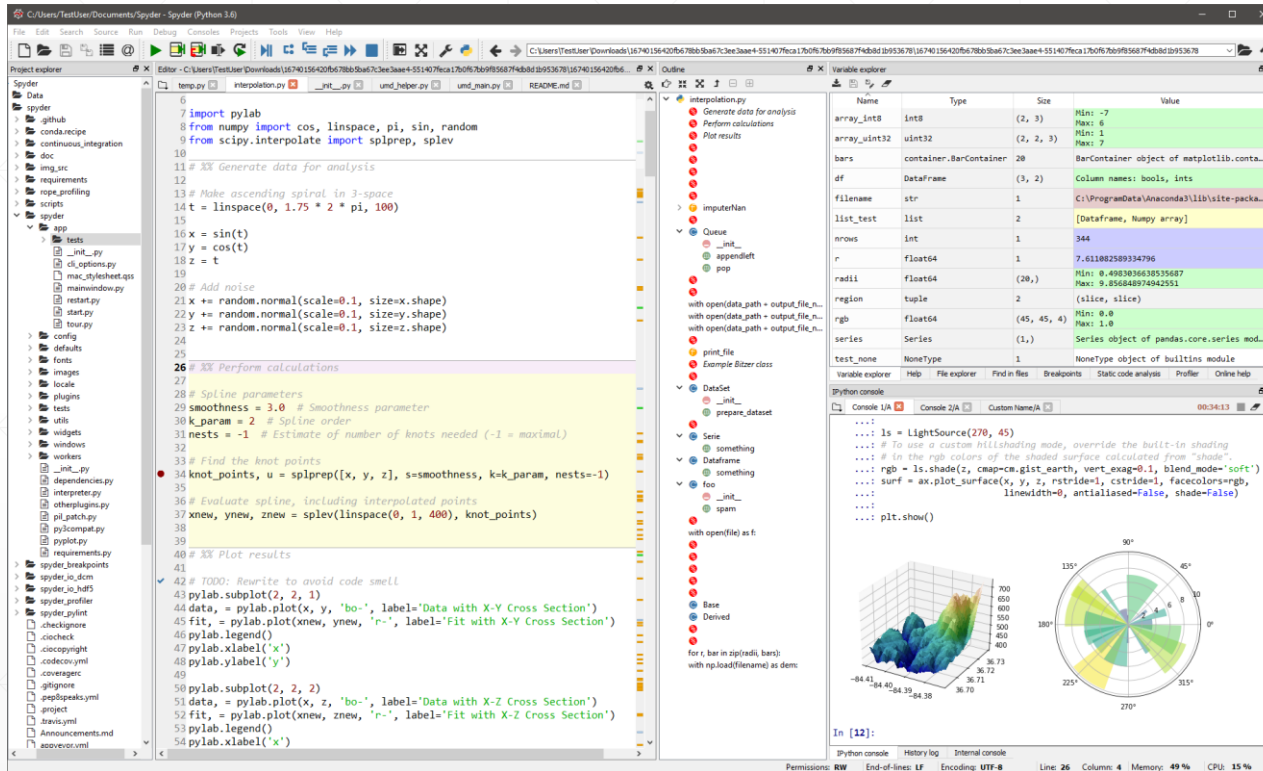


- Terminal 메뉴에서 Run Build Task를 실행하여 현재 Python 문서를 실행할 수 있음

Python 개발도구

Spyder 설치와 사용법

Spyder 설치 및 사용법 (1/12)



- Python 프로그래밍을 위해 무료로 제공되는 통합 개발 환경(IDE)
- Anaconda를 이용하는 경우 기본 제공
- 가상환경을 새롭게 생성하는 경우 각 가상환경에 Spyder를 추가로 설치하여 각 가상환경의 Spyder 개발 환경을 구축할 수 있음

Spyder 설치 및 사용법 (2/12)

```
Anaconda Prompt - conda install spyder

(tensorflow) C:\Users\joeun-IT>conda install spyder
Solving environment: done

## Package Plan ##

  environment location: C:\Users\joeun-IT\AppData\Local\conda\conda\envs\tensorflow

  added / updated specs:
    - spyder

The following packages will be downloaded:
```

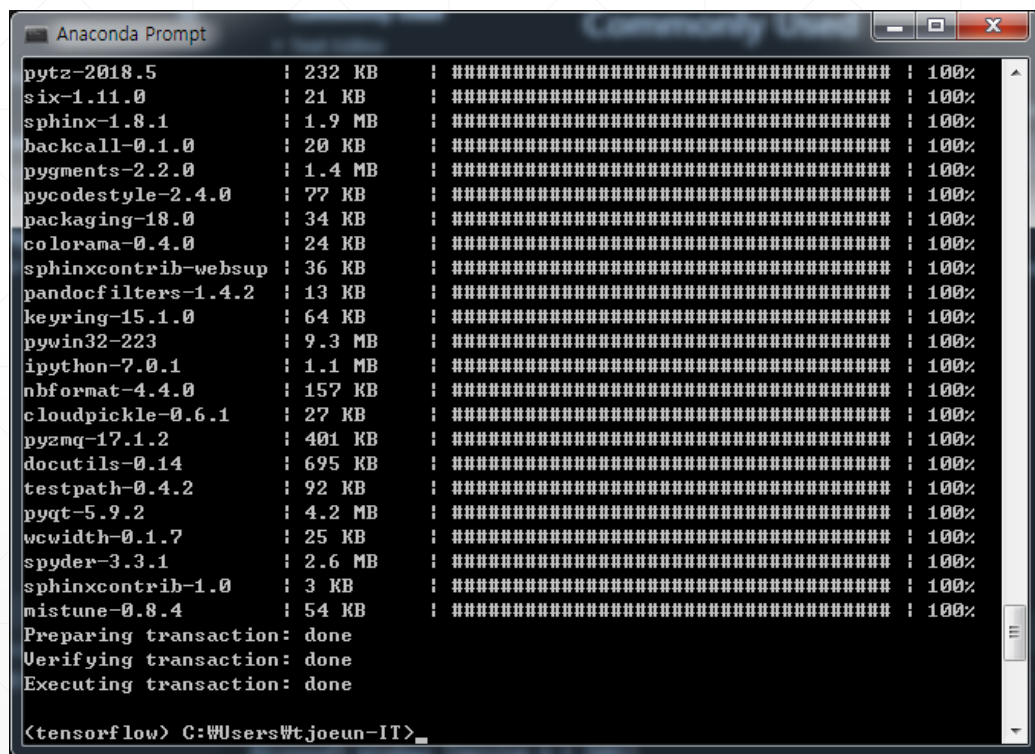
package	build	
prompt_toolkit-2.0.6	py36_0	465 KB
qtawesome-0.5.1	py36_1	570 KB
spyder-kernels-0.2.6	py36_0	70 KB
entrypoints-0.2.3	py36_2	9 KB
lazy-object-proxy-1.3.1	py36hfafa6e2cd_2	31 KB
isort-4.3.4	py36_0	76 KB
simplegeneric-0.8.1	py36_2	10 KB
jsonschema-2.6.0	py36h7636477_0	104 KB
libpng-1.6.35	h2a8f88b_0	589 KB
pickleshare-0.7.5	py36_0	13 KB
jupyter_client-5.2.3	py36_0	188 KB

```
Anaconda Prompt - conda install spyder

qtpy: 1.5.2-py36_0
requests: 2.19.1-py36_0
rope: 0.11.0-py36_0
simplegeneric: 0.8.1-py36_2
sip: 4.19.8-py36h6538335_0
six: 1.11.0-py36_1
snowballstemmer: 1.2.1-py36h763602f_0
sphinx: 1.8.1-py36_0
sphinxcontrib: 1.0-py36_1
sphinxcontrib-websupport: 1.1.0-py36_1
spyder: 3.3.1-py36_1
spyder-kernels: 0.2.6-py36_0
sqlite: 3.25.2-hafa6e2cd_0
testpath: 0.4.2-py36_0
tornado: 5.1.1-py36hafa6e2cd_0
traitlets: 4.3.2-py36h096827d_0
typed-ast: 1.1.0-py36hafa6e2cd_0
urllib3: 1.23-py36_0
wcwidth: 0.1.7-py36h3d5aa90_0
webencodings: 0.5.1-py36_1
win_inet_pton: 1.0.1-py36_1
wrapt: 1.10.11-py36hafa6e2cd_2
zeromq: 4.2.5-he025d50_1
zlib: 1.2.11-h8395fce_2

Proceed [y]/n)? y
```

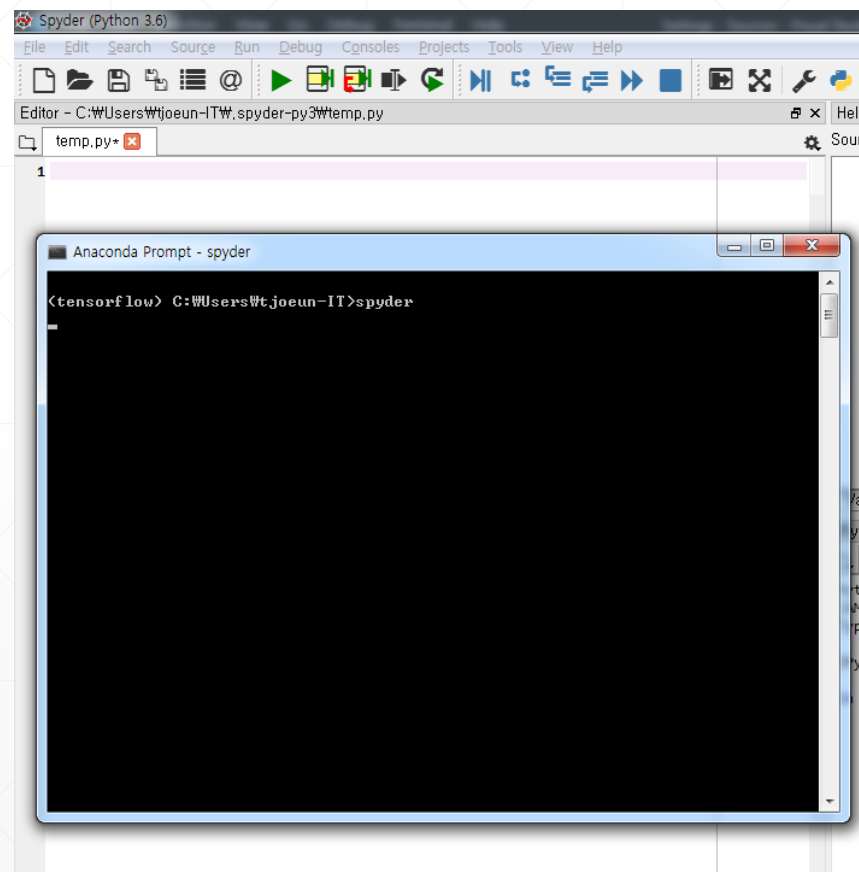
Spyder 설치 및 사용법 (3/12)



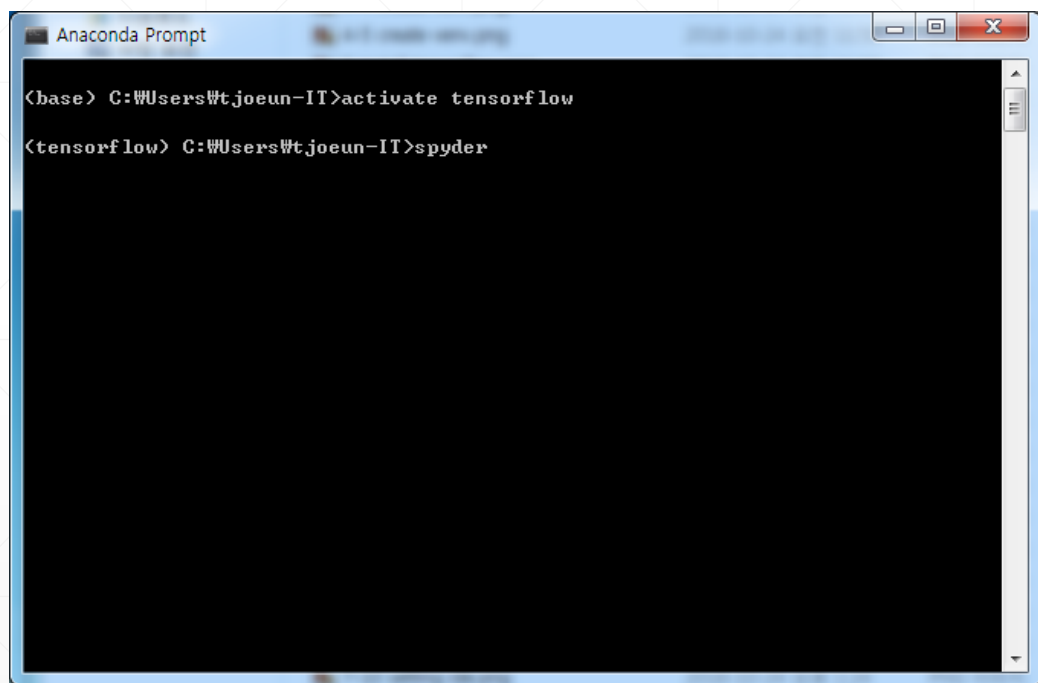
```

Anaconda Prompt
pytz-2018.5      : 232 KB : ##### : 100%
six-1.11.0       : 21 KB : ##### : 100%
sphinx-1.8.1     : 1.9 MB : ##### : 100%
backcall-0.1.0   : 20 KB : ##### : 100%
pygments-2.2.0   : 1.4 MB : ##### : 100%
pycodestyle-2.4.0 : 77 KB : ##### : 100%
packaging-18.0   : 34 KB : ##### : 100%
colorama-0.4.0   : 24 KB : ##### : 100%
sphinxcontrib-websup : 36 KB : ##### : 100%
pandocfilters-1.4.2 : 13 KB : ##### : 100%
keyring-15.1.0   : 64 KB : ##### : 100%
pywin32-223      : 9.3 MB : ##### : 100%
ipython-7.0.1    : 1.1 MB : ##### : 100%
nbformat-4.4.0   : 157 KB : ##### : 100%
cloudpickle-0.6.1 : 27 KB : ##### : 100%
pyzmq-17.1.2     : 401 KB : ##### : 100%
docutils-0.14    : 695 KB : ##### : 100%
testpath-0.4.2   : 92 KB : ##### : 100%
pyqt-5.9.2       : 4.2 MB : ##### : 100%
wcwidth-0.1.7    : 25 KB : ##### : 100%
spyder-3.3.1     : 2.6 MB : ##### : 100%
sphinxcontrib-1.0 : 3 KB : ##### : 100%
mistune-0.8.4    : 54 KB : ##### : 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done

(tensorflow) C:\Users\Wt.joeun-IT>
```



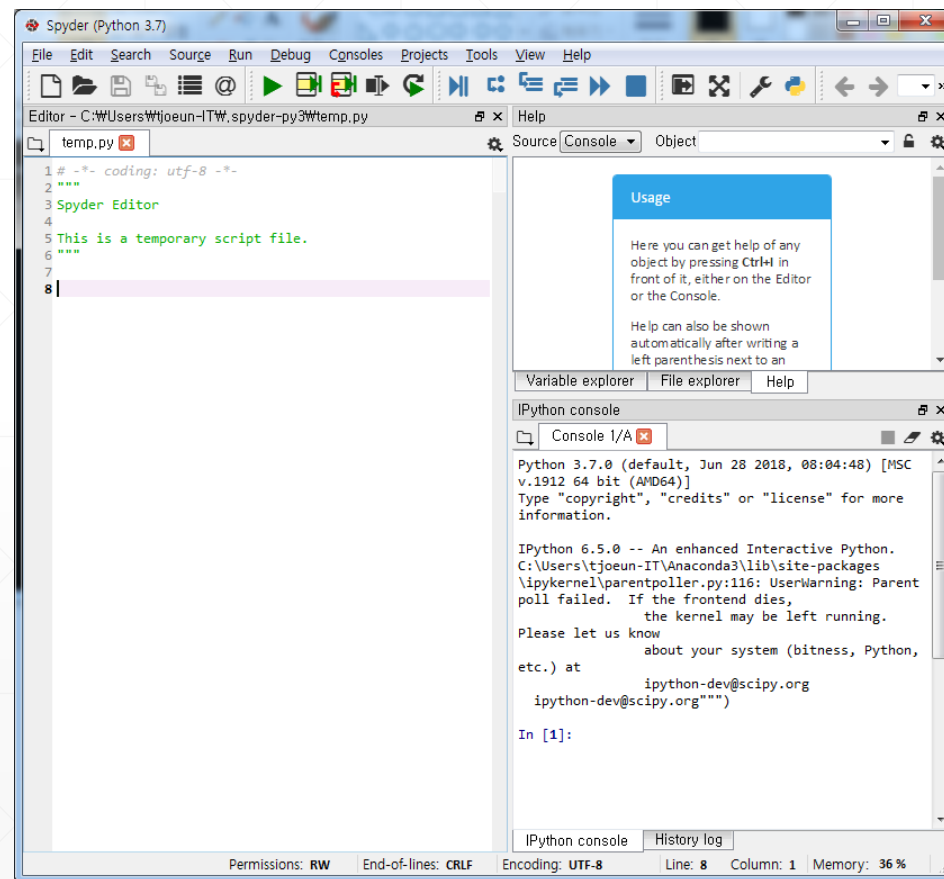
Spyder 설치 및 사용법 (4/12)



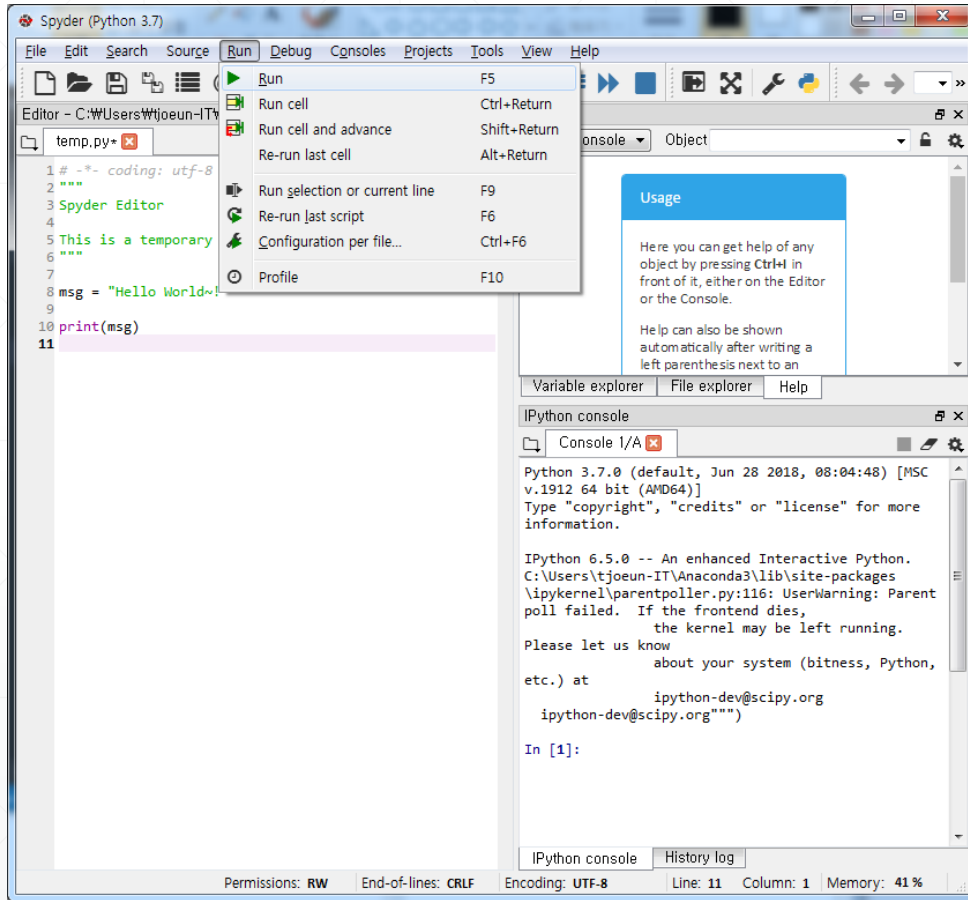
```
Anaconda Prompt

(base) C:\Users\tjoeun-IT>activate tensorflow

(tensorflow) C:\Users\tjoeun-IT>spyder
```

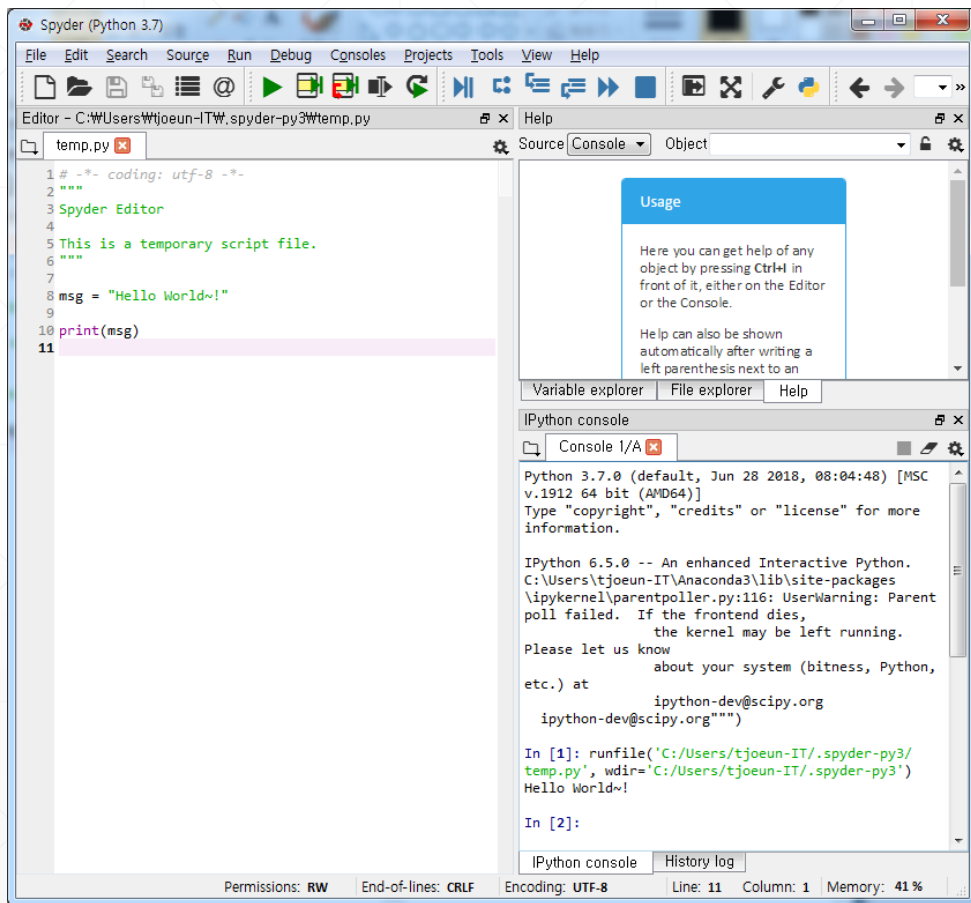


Spyder 설치 및 사용법 (5/12)



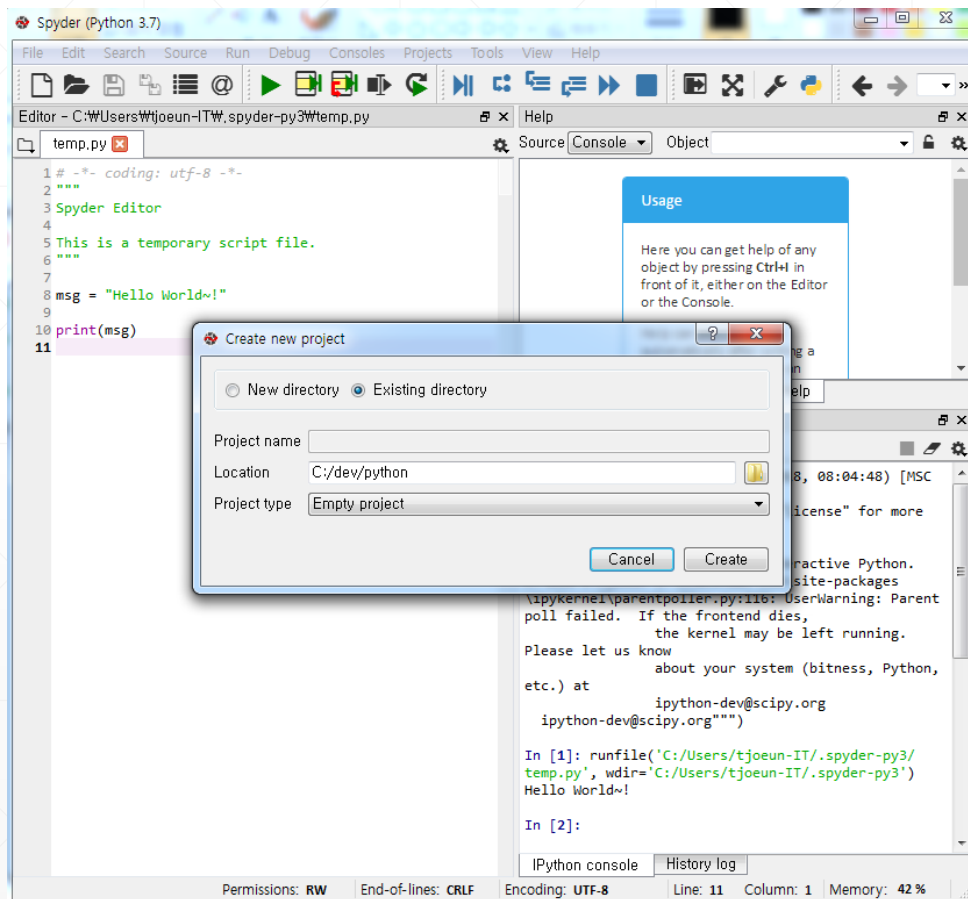
- Run 메뉴에서 Run을 실행하여 현재 Python 문서를 실행할 수 있음
- F5 단축키를 사용할 수 있음
- 실행의 결과는 오른쪽 하단의 콘솔 창에 출력됨

Spyder 설치 및 사용법 (6/12)



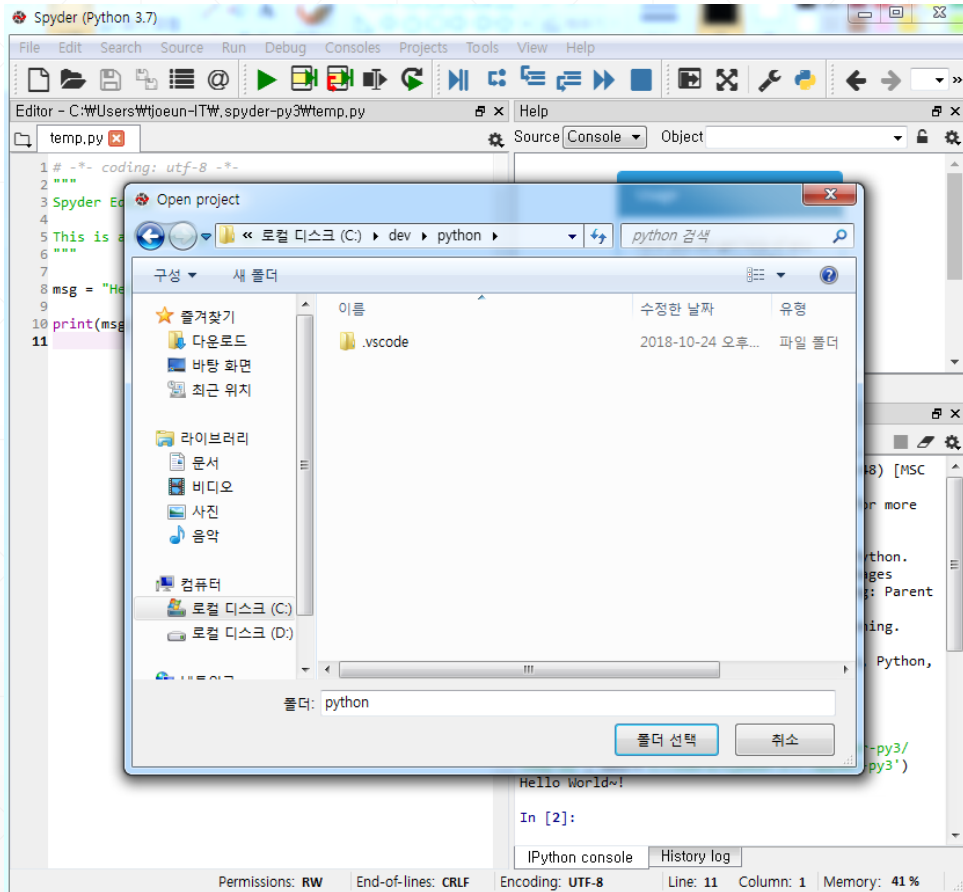
- Run 메뉴에서 Run을 실행하여 현재 Python 문서를 실행할 수 있음
- F5 단축키를 사용할 수 있음
- 실행의 결과는 오른쪽 하단의 콘솔 창에 출력됨

Spyder 설치 및 사용법 (7/12)



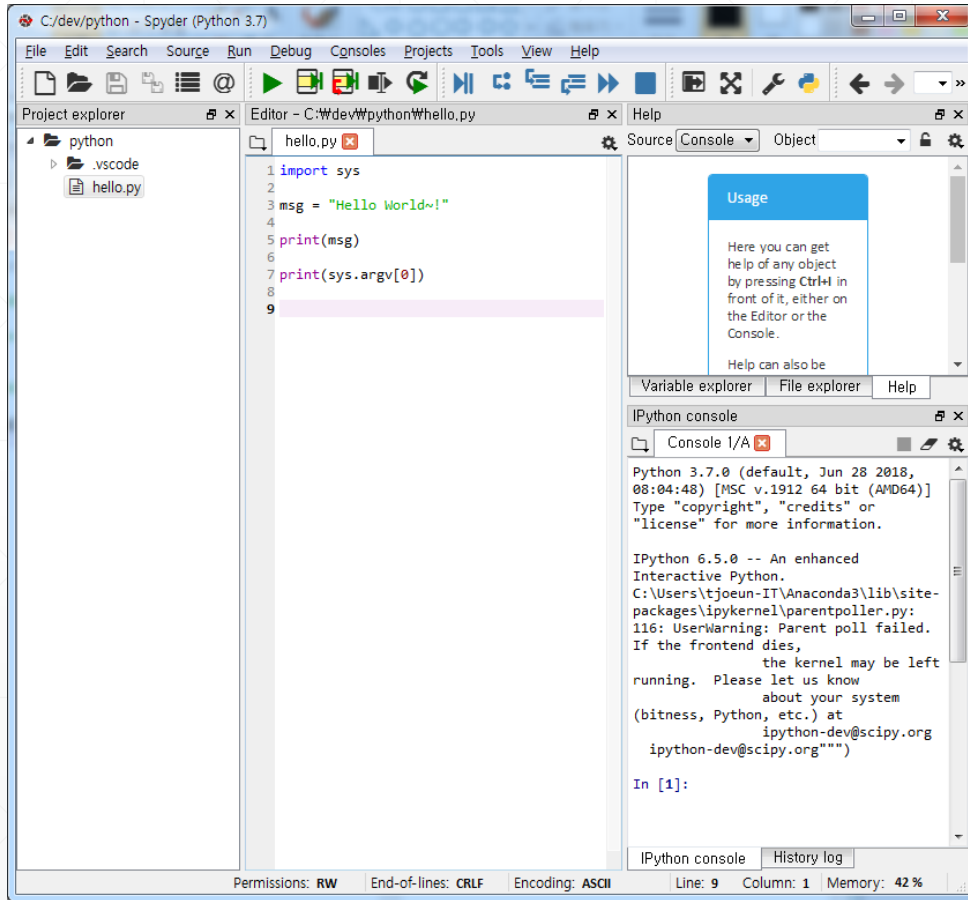
- Spyder 에서의 프로젝트 생성
- 특정 디렉토리를 기준으로 프로젝트를 생성
- Projects 메뉴에서 New Project를 이용하여 프로젝트를 생성할 수 있음
- 새로운 디렉토리를 생성하거나 기존에 존재하는 디렉토리를 이용하여 프로젝트를 생성할 수 있음

Spyder 설치 및 사용법 (8/12)



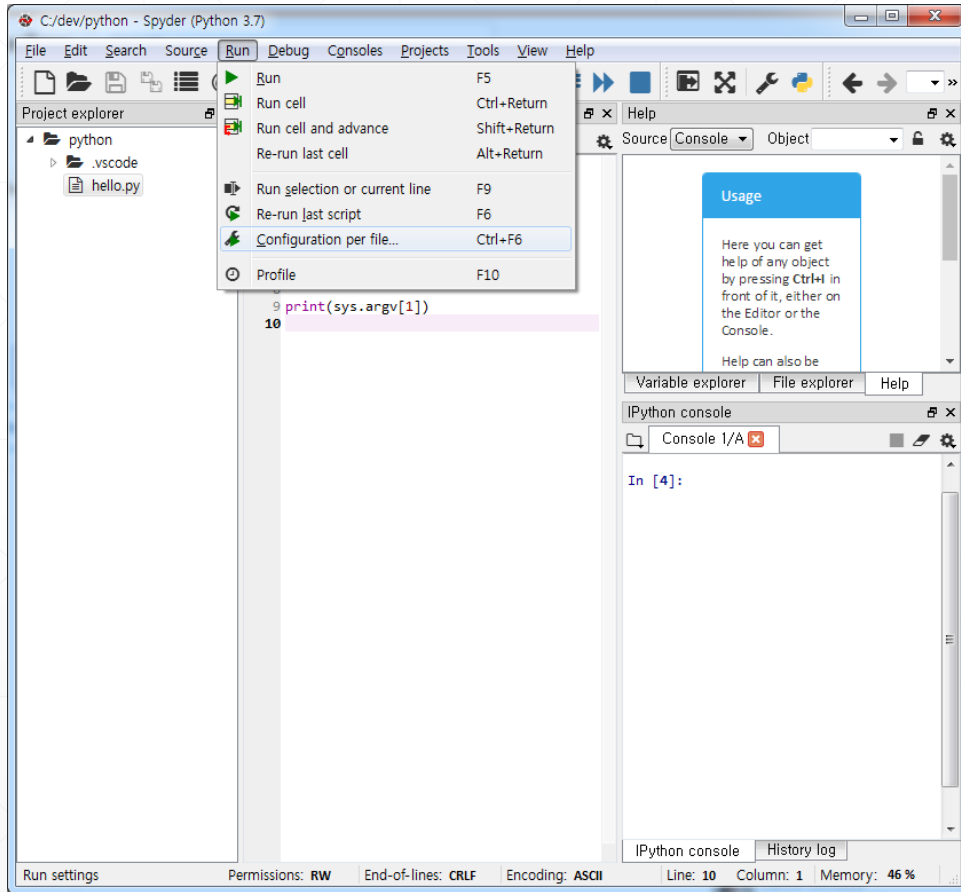
- Spyder 에서의 프로젝트 생성
- 특정 디렉토리를 기준으로 프로젝트를 생성
- Projects 메뉴에서 New Project를 이용하여 프로젝트를 생성할 수 있음
- 새로운 디렉토리를 생성하거나 기존에 존재하는 디렉토리를 이용하여 프로젝트를 생성할 수 있음

Spyder 설치 및 사용법 (9/12)



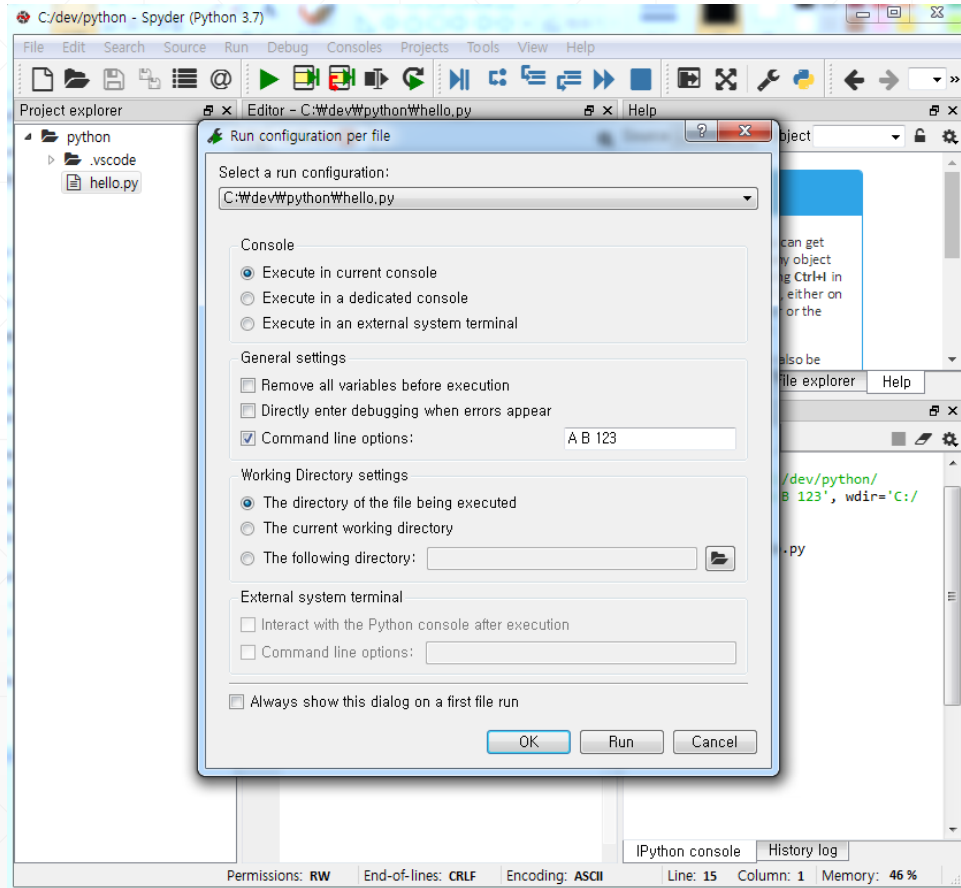
- Spyder 에서의 프로젝트 생성
- 특정 디렉토리를 기준으로 프로젝트를 생성
- Projects 메뉴에서 New Project를 이용하여 프로젝트를 생성할 수 있음
- 새로운 디렉토리를 생성하거나 기존에 존재하는 디렉토리를 이용하여 프로젝트를 생성할 수 있음

Spyder 설치 및 사용법 (10/12)



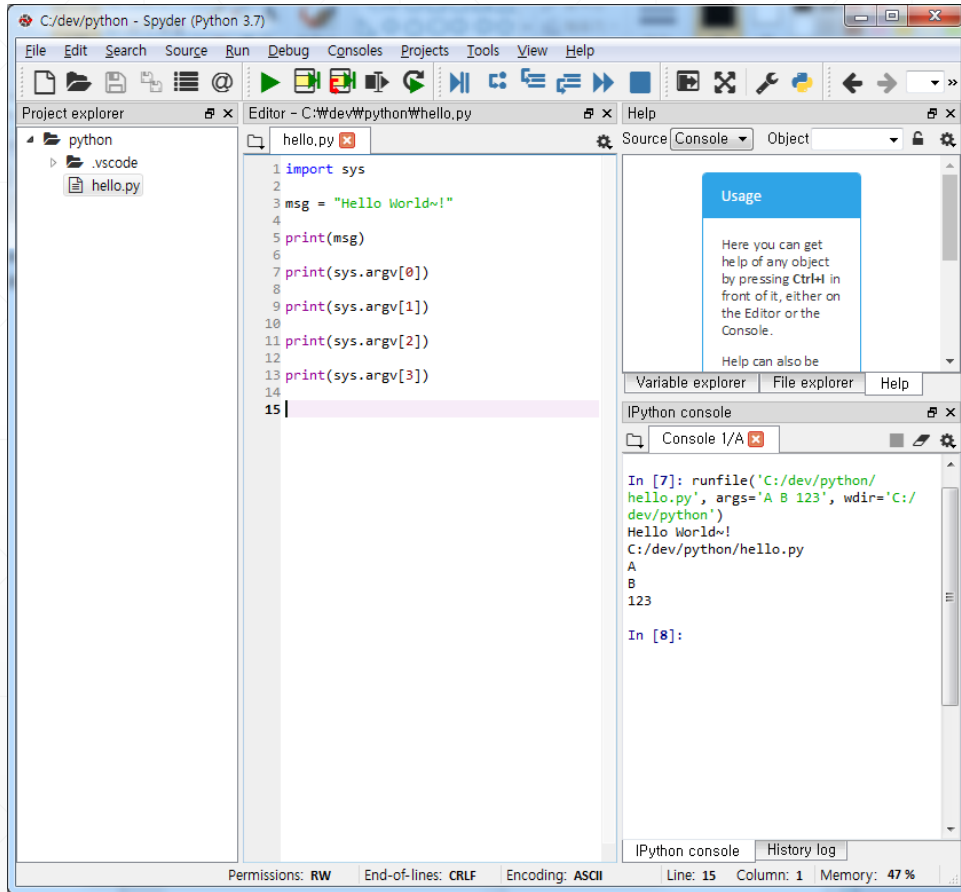
- Spyder를 이용하여 실행할 때 실행 매개변수를 전달하는 방법
- Python 프로그램 구동 시 필요한 정보를 제공할 수 있음
- Run 메뉴에서 Configuration per file을 이용하여 설정할 수 있음
- Command line options에 입력할 매개변수를 공백을 기준으로 입력

Spyder 설치 및 사용법 (11/12)



- Spyder를 이용하여 실행할 때 실행 매개변수를 전달하는 방법
- Python 프로그램 구동 시 필요한 정보를 제공할 수 있음
- Run 메뉴에서 Configuration per file을 이용하여 설정할 수 있음
- Command line options에 입력할 매개변수를 공백을 기준으로 입력

Spyder 설치 및 사용법 (12/12)



- Spyder를 이용하여 실행할 때 실행 매개변수를 전달하는 방법
- Python 프로그램 구동 시 필요한 정보를 제공할 수 있음
- Run 메뉴에서 Configuration per file을 이용하여 설정할 수 있음
- Command line options에 입력할 매개변수를 공백을 기준으로 입력