

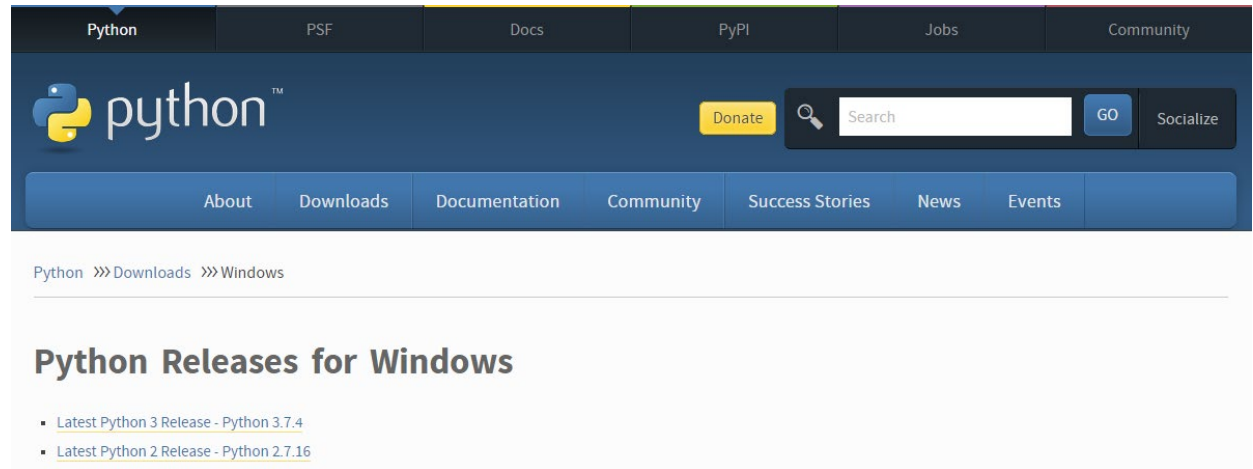
# 파이썬 소개와 설치

오동인

경희대학교 의과대학 의공학교실

# 파이썬 설치

- [www.python.org](http://www.python.org)
- 파이썬 설치 파일 다운로드



- 설치 프로그램 실행



- 어느 폴더 위치에서도 파이썬이 실행될 수 있도록 Add Python 3.7 to PATH 항목 체크
- 설치 진행 → 완료
- 시작 > Python 3.7 >

# 파이썬 실행 : 커맨드라인의 인터프리터 방식

- Python 3.7 폴더의 'Python 3.7 (32-bit)' 실행

Python 3.7 (32-bit)

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

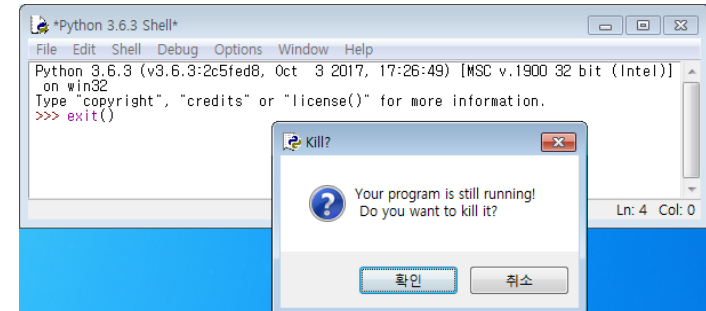
프롬프트(>>>) 커서(cursor)

- 종료 시 `exit()`, `quit()`
- 윈도우 명령 프롬프트를 통한 'python' 명령 실행

```
명령 프롬프트 - python
Microsoft Windows [Version 10.0.18362.295]
(c) 2019 Microsoft Corporation. All rights reserved.

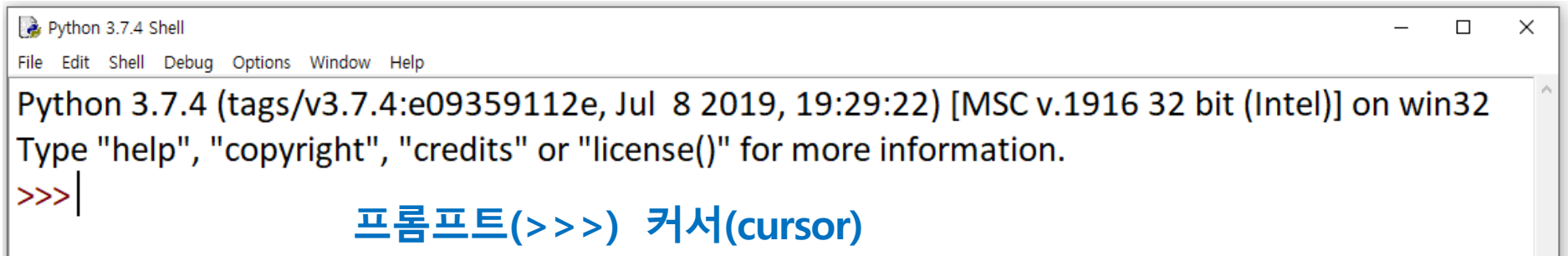
C:\Users\USER>python
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 22:22:05) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

- 모든 내부 동작을 먼저 종료한 후 종료 (중요)



# 파이썬 실행 : 통합 개발환경 방식

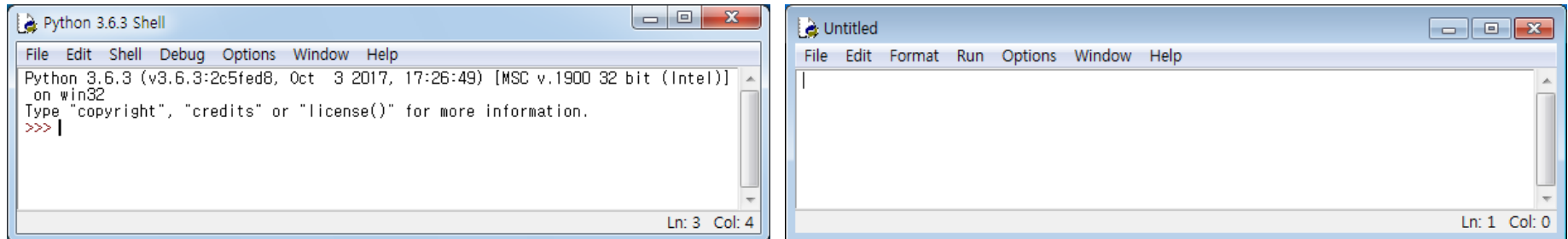
- Python 3.7 폴더의 IDLE (Python 3.7 32-bit) 실행



```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> |
```

프롬프트(>>>) 커서(cursor)

- 종료 시 Ctrl+D, exit(), quit()
- 파이썬 IDLE
  - IDLE(Integrated Development and Learning Environment)
  - 파이썬 프로그램 작성을 도와주는 통합 개발환경
  - 셸 창(shell window)과 에디터 창(editor window)으로 구성



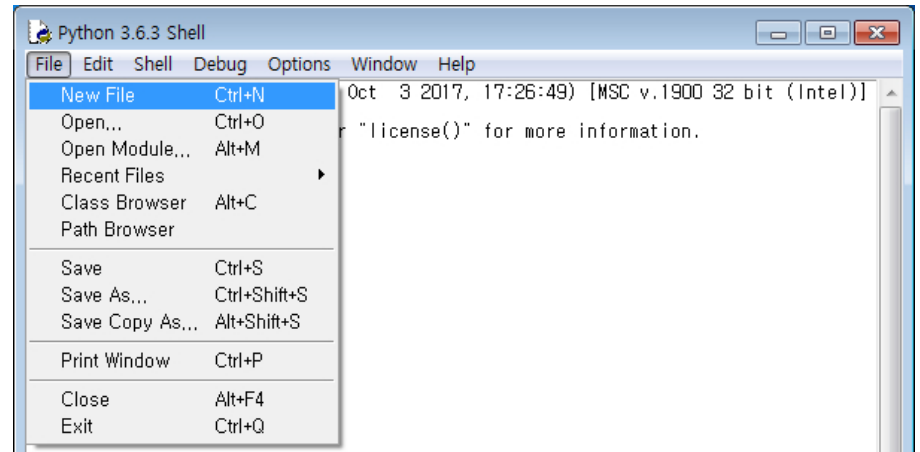
# 파이썬 셸 (Python Shell)

- 대화형 인터프리터
- 프롬프트에 입력된 명령, 문장들은 파이썬 인터프리터가 해석하고 실행하여 결과 출력
- REPL 툴
  - 1) Read :       사용자로부터 프롬프트에서 명령이나 문장을 입력 받아,
  - 2) Evaluate :   명령이나 문장을 즉시 해석하고 평가하며,
  - 3) Print :       출력 결과를 표시한 후,
  - 4) Loop :       다시 프롬프트로 돌아가는 프로그램

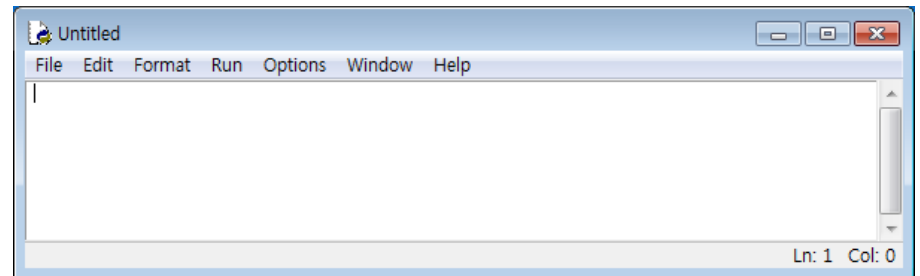
# 파이썬 에디터 (1)

- 파이썬 에디터를 통한 코드 입력 및 실행

1) 파이썬 셸의 메뉴에서  
[File] → [New File]을 실행

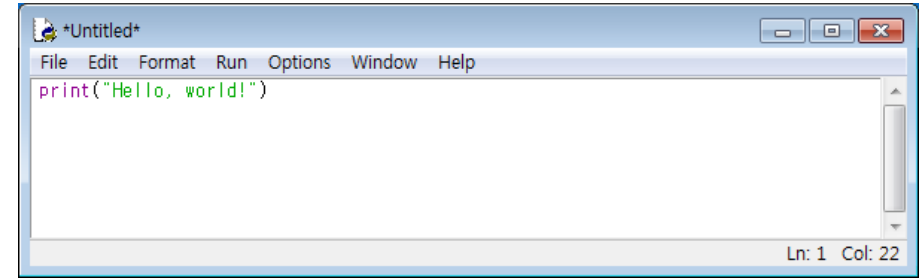


2) 파이썬 에디터가 나타나며,  
파이썬 셸과는 달리  
프롬프트가 존재하지 않음

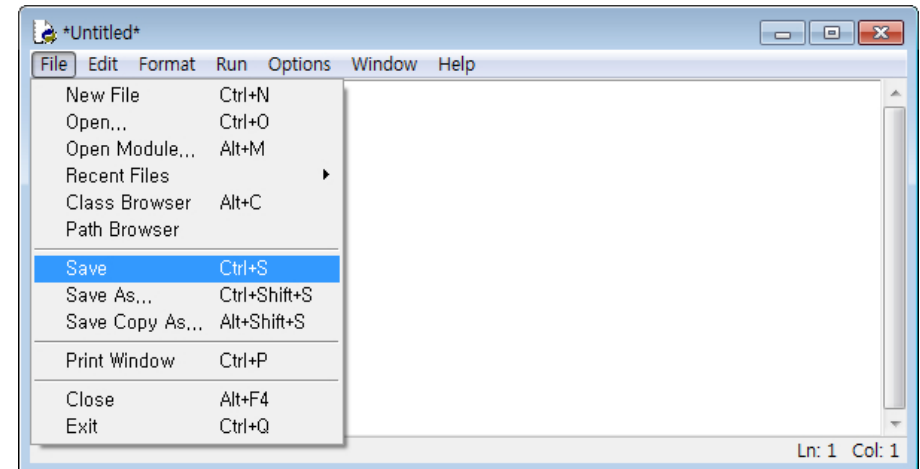


# 파이썬 에디터 (2)

- 파이썬 에디터를 통한 코드 입력 및 실행
  - 3) `print("Hello, world!")` 입력

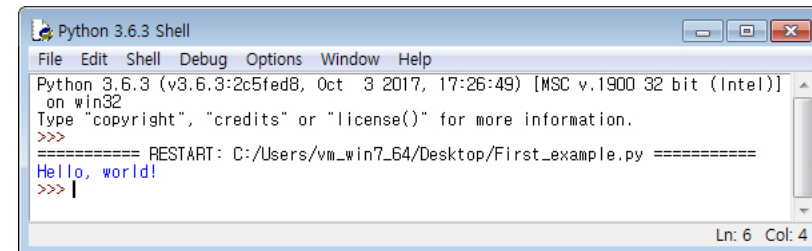
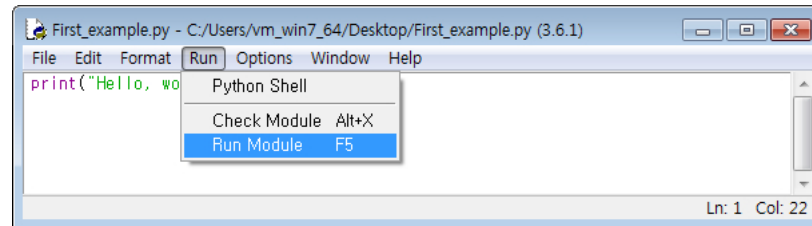
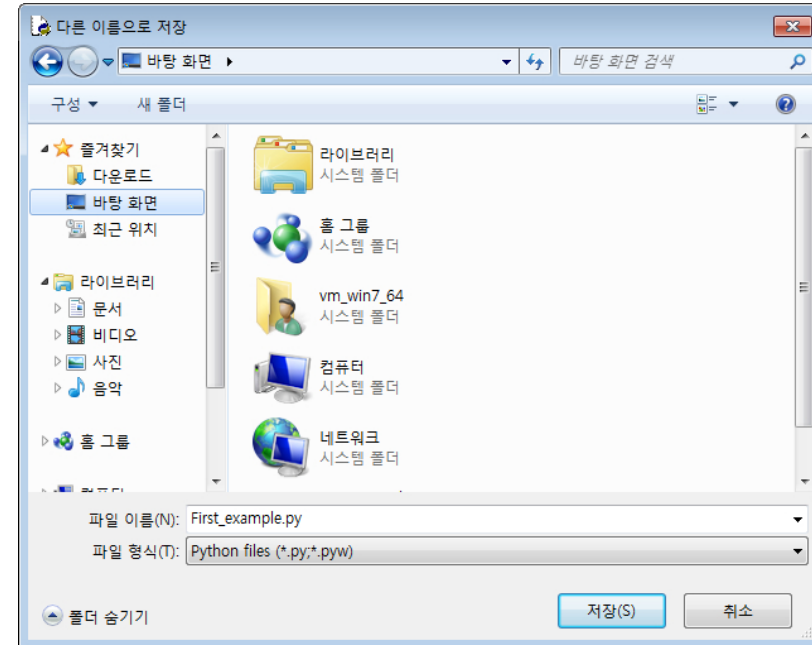


- 4) 파일로 저장하기 위하여  
Ctrl+S 단축키를 누르거나,  
[File] → [Save] 메뉴 실행



# 파이썬 에디터 (3)

- 파이썬 에디터를 통한 코드 입력 및 실행
  - 5) [다른 이름으로 저장] 대화 상자에서 저장할 폴더를 지정한 후 파일 이름을 입력하고, [저장] 단추 클릭.  
파일 이름을 지정할 때 주의할 점은 파일 형식이 py로 되어 있는지 확인
  - 6) [Run] → [Run Module] 메뉴, 또는 F5 단축키 실행
  - 7) 실행 결과가 파이썬 셸에 출력됨





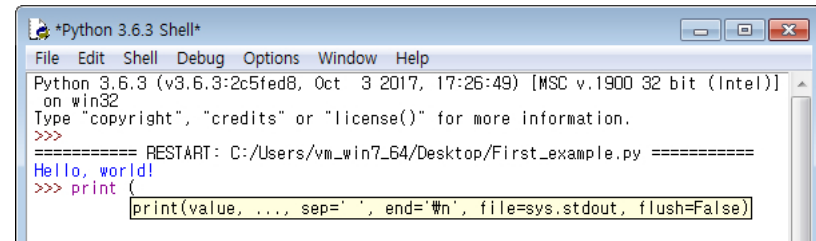
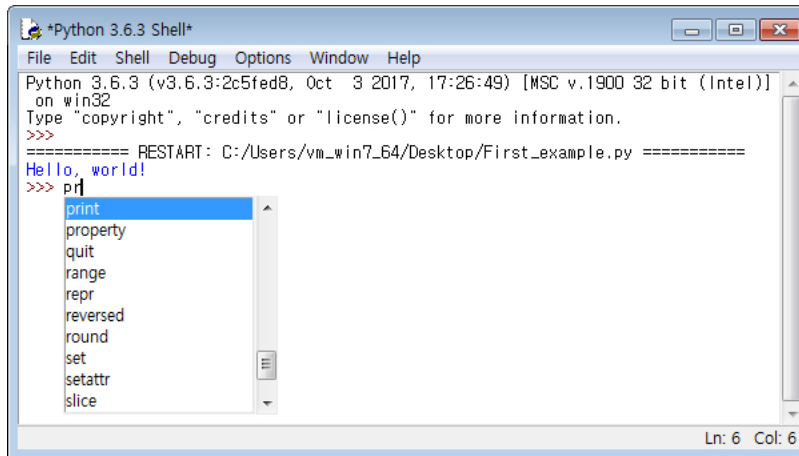
# 들여쓰기, 컬러링, 함수명 자동완성

- 들여쓰기
- 컬러링

```
>>> kor_score = 90                                # kor_score : 변수
>>> if kor_score >= 80:                             # if, else : 예약어
    print("합격입니다.")                          # 자동 들여쓰기 기능
else :
    print("불합격입니다.")                        # print : 함수, "불합격입니다." : 문자열

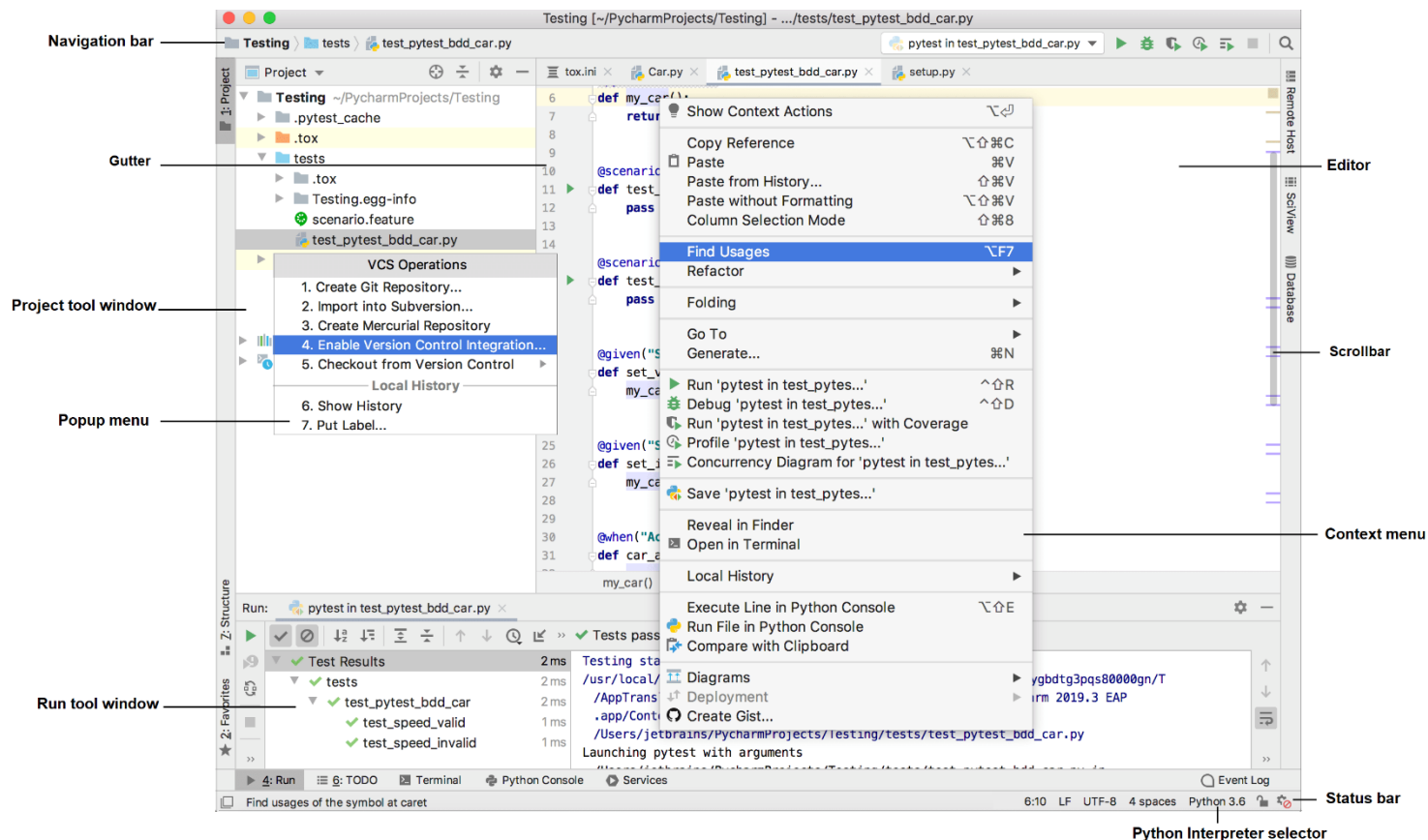
합격입니다.                                         # 합격입니다 : 결과
```

- 함수명 자동완성



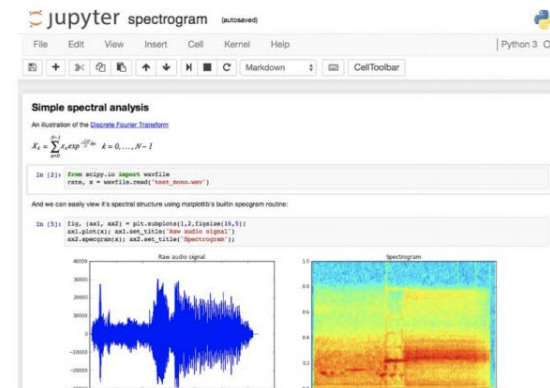
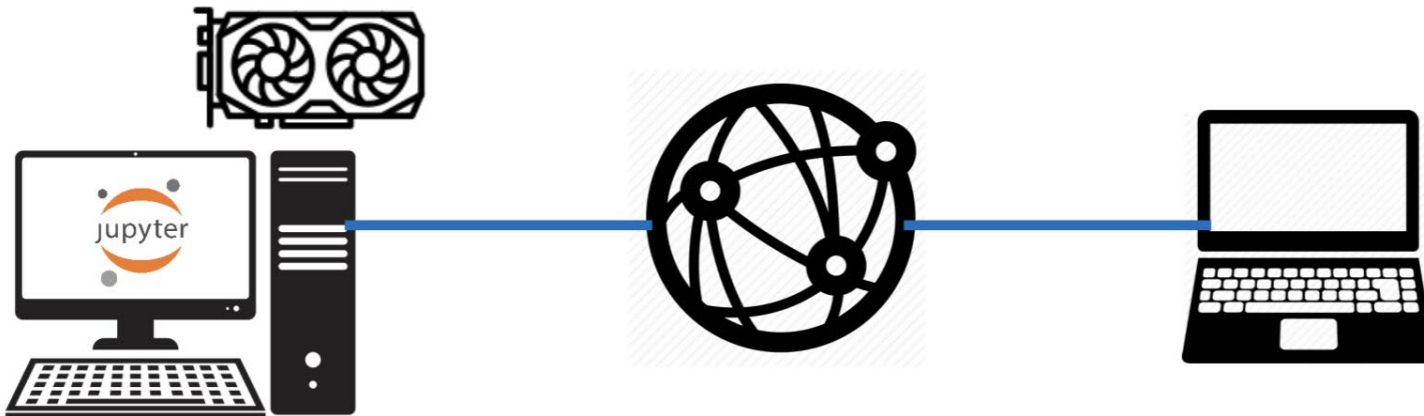
# PyCharm

- Dedicated Python Integrated Development Environment (IDE)
- <https://www.jetbrains.com/ko-kr/pycharm/>
- 무료, 오픈소스인 Community 사용 (등록 후 이용가능)



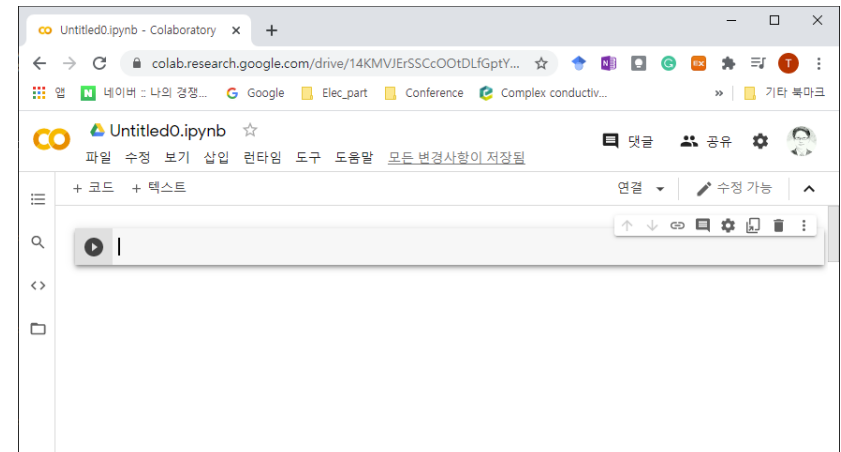
# 주피터 노트북 사용하기

- Anaconda 설치
- <https://www.anaconda.com/products/individual>
- Jupyter Notebook
- 웹 브라우저에서 파이썬 코드를 작성하고 실행해 볼 수 있는 개발 도구
  - 원격 코딩 가능
  - 코드 블록 단위로 실행/디버깅
  - Text block을 이용한 문서화
  - Figure plotting 등 GUI



# 구글 코랩 사용하기

- Google Colaboratory = Google Drive + Jupyter Notebook
  - 구글 계정 전용의 가상 머신 지원 - GPU 포함
  - Google drive 문서와 같이 링크만으로 접근/협업 가능
  - 코드 실행 시 delay 존재
- 개인 구글 계정 필요
- Colab과 Jupyter Notebook 사용 방법은 매우 유사
- 파일 생성 방법
  - 개인구글계정으로 접속
  - <https://colab.research.google.com> 접속
  - Google 드라이브 탭으로 이동
  - 새 python 노트 선택
- .ipynb 파일
  - Code cell과 text cell로 구성



# 화면구성

파일이름

Cell 추가

Text Cell

Code Cell

The screenshot shows the Google Colaboratory web interface. The browser tab is titled 'Ex01.ipynb - Colaboratory'. The address bar shows the URL 'colab.research.google.com/drive/14KMVJErSSCcOOtDLfGptY...'. The interface includes a top navigation bar with the Colab logo, the file name 'Ex01.ipynb', and a star icon. Below this is a toolbar with icons for file management, running, and help. The main workspace contains a 'Text Cell' and a 'Code Cell'. The 'Text Cell' contains the text '여러 줄의 주석을 효과적으로 시각화' and '예시'. The 'Code Cell' contains the code '# Code Cell', 'print("Hello World!")', and '# 코드 삽입 후 블럭 단위로 실행 가능'. The output of the code cell is 'Hello World!'. The interface also shows a 'RAM 디스크' indicator and a '수정 가능' status.

Ex01.ipynb

파일 수정 보기 삽입 런타임 도구 도움말 모든 변경사항이 저장됨

코드 + 텍스트

Text Cell

- 여러 줄의 주석을 효과적으로 시각화
- 예시

# Code Cell  
print("Hello World!")  
# 코드 삽입 후 블럭 단위로 실행 가능

Hello World!

실행: Ctrl + Enter

실행결과

더블클릭 또는 Enter 키를 눌러 수정

[ ]

# 단축키

- 단축키 설정 가능 (Ctrl + M H)
- 유용한 단축키
  - 코드 셀 생성 : Ctrl + M A(B)
  - 코드 셀 실행 : Ctrl + Enter
  - 셀 삭제 : Ctrl + M D
  - 실행중인 셀 중단 : Ctrl + M I
  - 런타임 다시 시작 : Ctrl + M .
  - 코드(텍스트) 셀로 변환 : Ctrl + M Y(M)
  - 마지막 셀 작업 실행취소 : Ctrl + Shift + Z

## 키보드 환경설정

편집기 키 바인딩  
default

☒ Enter 키를 눌러 제안 수락

### 단축키

단축키를 추가하거나 변경하려면 키 조합을 클릭한 다음 새 키를 입력하세요. Ctrl+M을(를) 다중 키-이벤트 단축키의 접두어로 사용할 수 있습니다.

단축키 설정	.ipynb 다운로드	Ctrl+Shift+Enter	선택항목 실행
단축키 설정	.py 다운로드	단축키 설정	설정 열기
단축키 설정	GitHub Gist로 사본 저장	단축키 설정	세로형 탭 레이아웃 보기
단축키 설정	GitHub에 사본 저장	단축키 설정	세션 관리
단축키 설정	Google 드라이브의 노트 별표표시/별표표시 해제	단축키 설정	섹션 제목 셀 추가
단축키 설정	Unmount ColabFS	단축키 설정	셀 또는 선택 항목 복사
단축키 설정	가로형 탭 레이아웃 보기	단축키 설정	셀 또는 선택 항목 잘라내기
단축키 설정	노트 공유	Ctrl+Click	셀 선택 토글
단축키 설정	노트 설정 열기	Ctrl+Shift+Y	셀 작업 다시 실행
단축키 설정	노트 업로드	Ctrl+M D	셀/선택항목 삭제
Ctrl+O	노트 열기	Shift+Enter	셀을 실행하고 다음 셀 선택
		Alt+Enter	셀을 실행하고 새 셀 삽입

[기본값 복원](#) [취소](#) [저장](#)

# 런타임

---

- 런타임
  - 실행시간을 기록
  - 런타임 메뉴에서 런타임 유형 변경 시 GPU/TPU 등 가속기 사용 가능
  - 12시간 실행 지원
  - Idle 상태 진입 후 90분 shutdown

# EOD