


# Python 01

김은진



# Computer

---

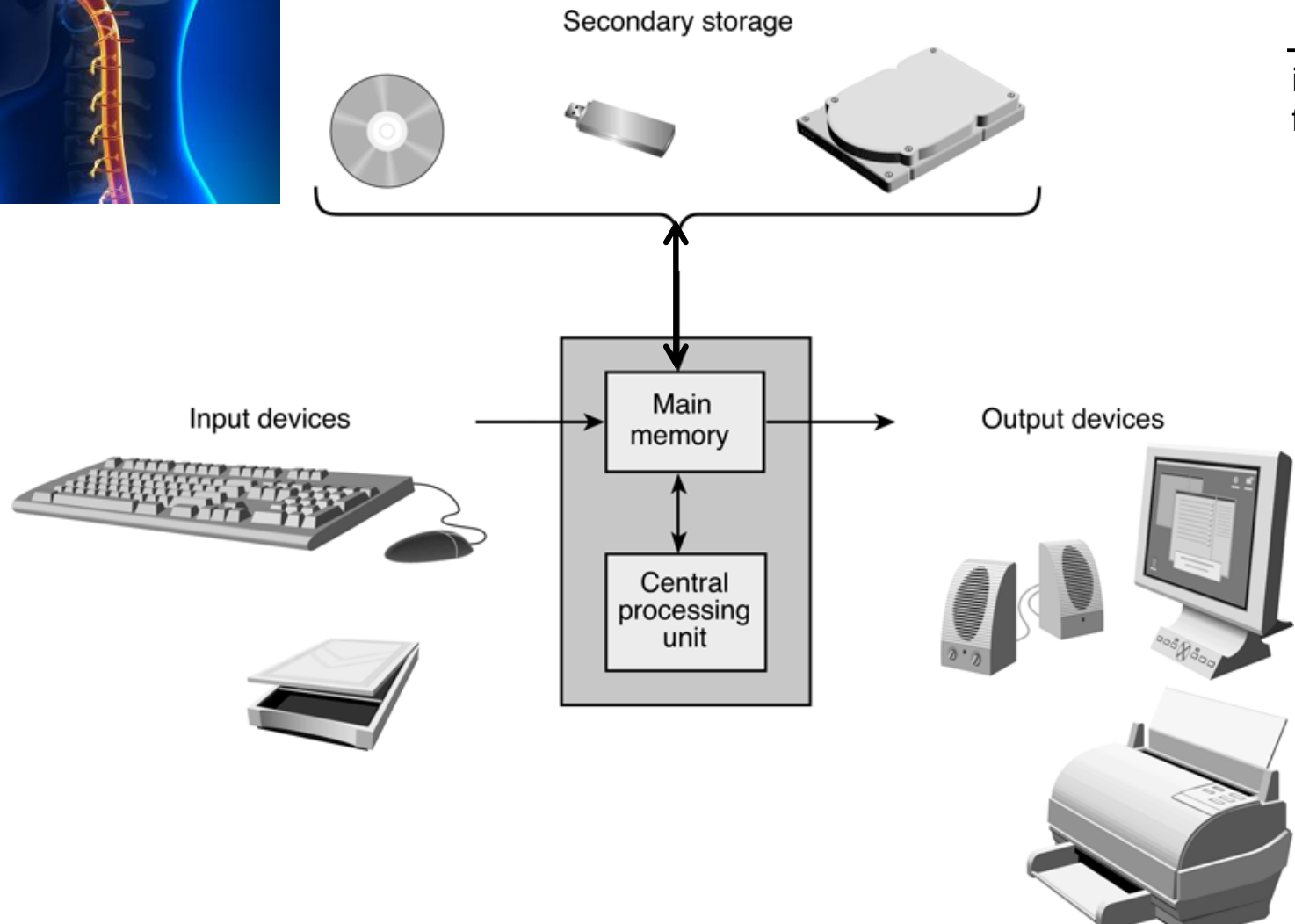
- *a machine that can*
  - *receive,*
  - *store,*
  - *transform,*
  - *and output*

**data of all kinds**
- Computer system
  - Hardware
  - Software
    - 저장장치에 저장된 특정한 목적의 하나 또는 다수의 컴퓨터 프로그램  
램





# Computer Hardware



# Memory

---

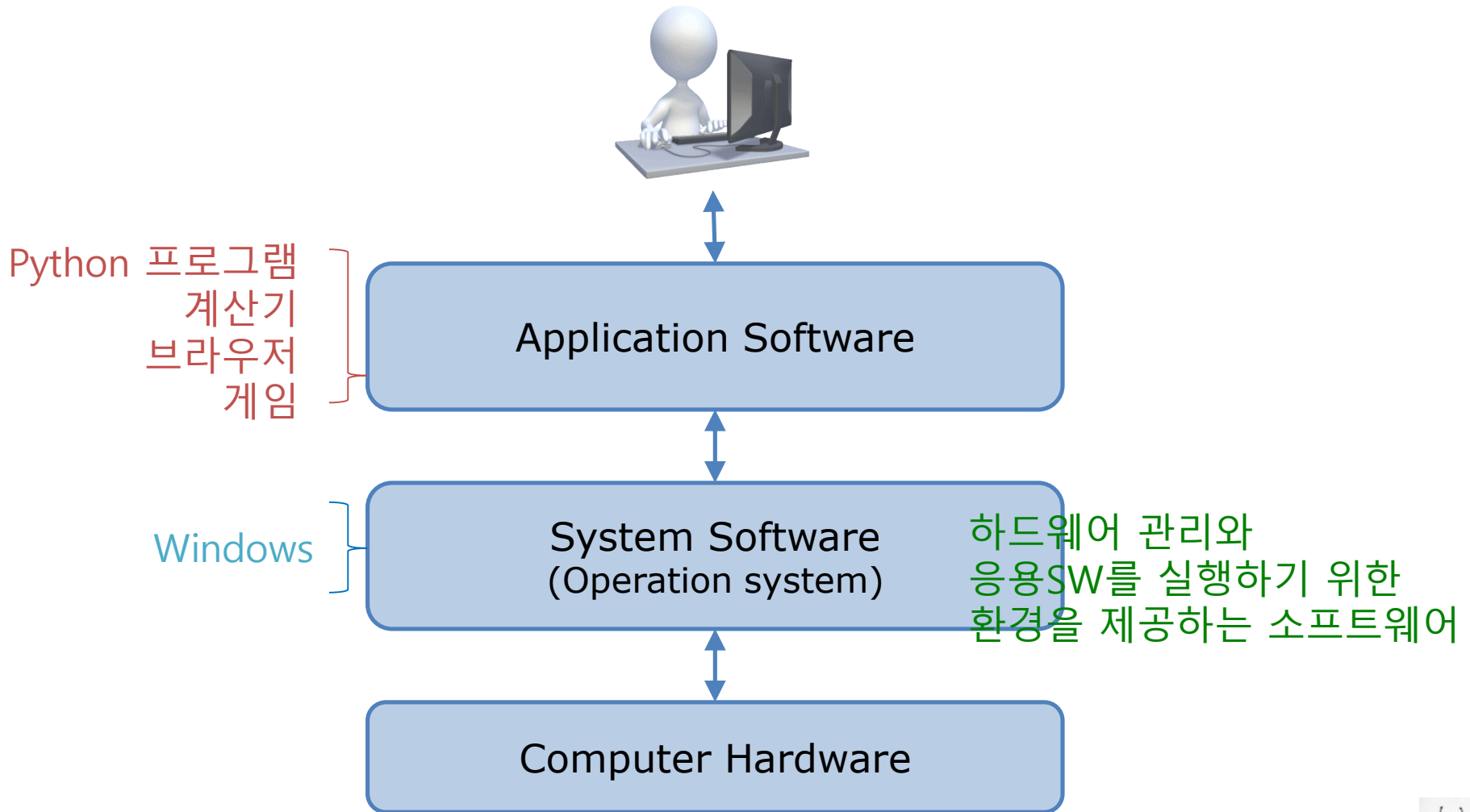
- Main memory stores programs, data, and results.
- Secondary storage retains data even when the power to the drive is off.

Term	Abbreviation	Equivalent to	Comparison to power of 10
Byte	B	8 bits	
Kilobyte	KB	1,024( $2^{10}$ )bytes	$>10^3$
Megabyte	MB	1,048,576( $2^{20}$ )bytes	$>10^6$
Gigabyte	GB	1,073,741,824( $2^{30}$ )bytes	$>10^9$
Terabyte	TB	1,099,511,627,776( $2^{40}$ )bytes	$>10^{12}$
Petabyte	PB	( $2^{50}$ )bytes	$>10^{15}$
Exabyte	EB	( $2^{60}$ )bytes	$>10^{18}$



# Computer Software

---



# Program-Process

프로그램(Program): 보조기억장치에 저장된 파일



notepad.exe

메모리  
로드

프로세스(Process) : CPU와 메모리에 로드 된 파일

제목 없음 - 메모장 파일(F) 편집(E) 서식(O) 보기(V)	서비스 호스트: Microsoft Acco...	0%	3.2MB	0MB/s	0Mbps	0%
	메모장	0.1%	2.9MB	0MB/s	0Mbps	0%
	제목 없음 - 메모장					
	서비스 및 컨트롤러 응용 프	0.8%	2.8MB	0MB/s	0Mbps	0%



CTRL + ALT + DEL

프로세스

작업 관리자						
파일(F) 옵션(O) 보기(V)						
프로세스 성능 앱 기록 시작프로그램 사용자 세부 정보 서비스						
이름	9% CPU	69% 메모리	0% 디스크	0% 네트워크	1% GPU	GPU 엔진
> Google Chrome(11)	0%	360.4MB	0.1MB/s	0Mbps	0%	GPU 0
> eclipse.exe	0.1%	165.6MB	0MB/s	0Mbps	0%	
> Microsoft PowerPoint	0.1%	102.7MB	0MB/s	0Mbps	0%	
> Java(TM) Platform SE binary	0.4%	87.4MB	0MB/s	0Mbps	0%	
> Antimalware Service Executable	0.3%	47.2MB	0MB/s	0Mbps	0%	
Kited	0%	47.0MB	0MB/s	0Mbps	0%	
데스크톱 창 관리자	1.0%	26.8MB	0MB/s	0Mbps	0.9%	GPU 0
> 작업 관리자	1.2%	21.3MB	0.1MB/s	0Mbps	0%	
Windows 탐색기	2.0%	20.5MB	0MB/s	0Mbps	0%	
> 서비스 호스트: UtcSvc	0%	18.4MB	0MB/s	0Mbps	0%	
> 서비스 호스트: Diagnostic Poli...	0%	10.6MB	0MB/s	0Mbps	0%	
Intel Driver & Support Assistan...	0%	8.8MB	0MB/s	0Mbps	0%	
> Microsoft Windows Search 인...	0%	8.1MB	0MB/s	0Mbps	0%	
> Windows 탐색기	0%	7.0MB	0MB/s	0Mbps	0%	
< >						
간단히(D)						작업 끝내기(E)



# Can computers understand commands?

```
materials = []
currentMaterial = defaultMaterial
for line in self.contents.split("\n"):
    if line[:6] == 'mtllib':
        filename = ' '.join(line.split(' ')[1:])
        materials.extend(self.loader.load(filename, silent))

    if line[:6] == 'usemtl':
        name = line.split(' ')[1][0]
        if name == '(null)':
            currentMaterial = defaultMaterial
            continue
        for material in materials:
            if material.name == name:
                currentMaterial = material
                break
        else:
            currentMaterial = defaultMaterial

    if materials[3].
        diffuse
        name
    if line[:2] == '\
        coords = line[2:].split(' ')
        self.vertices.append(v([float(coords[0]), float(coords[1]), float(coords[2])])
```



?

실제 컴퓨터는 사람이 작성한  
'영어'명령어를 직접 실행하지 못한다!





# Programming Languages

## Python

```
def fib(n):  
    a, b = 1, 1  
    for i in range(n-1):  
        a, b = b, a + b  
    return a
```

## C

```
unsigned int fib(unsigned int n)  
{  
    if (n <= 0)  
        return 0;  
    else if (n <= 2)  
        return 1;  
    else {  
        unsigned int a,b,c;  
        a = 1;  
        b = 1;  
        while (1) {  
            c = a + b;  
            if (n <= 3) return c;  
            a = b;  
            b = c;  
            n--;  
        }  
    }  
}
```

## Assembler

```
fib:  
    mov edx, [esp+8]  
    cmp edx, 0  
    ja @f  
    mov eax, 0  
    ret  
  
@@:  
    cmp edx, 2  
    ja @f  
    mov eax, 1  
    ret  
  
@@:  
    push ebx  
    mov ebx, 1  
    mov ecx, 1  
  
@@:  
    lea eax, [ebx+ecx]  
    cmp edx, 3  
    jbe @f  
    mov ebx, ecx  
    mov ecx, eax  
    dec edx  
    jmp @b
```

Machine Language(기계어)



# Machine Code Example

---

$Y = Y + X$

## Machine Language Code

---

```
1100 0000 0010 0000 0000 0000
1011 0000 0001 0000 0000 0000
1001 0000 0010 0000 0000 0000
```

## Assembly Language Code

---

```
LOAD  Y
ADD   X
STORE Y
```

## Memory Cell Definitions:

Address	Name	Cell Contents
1000	X	32
2000	Y	16



# Processing high-level languages into machine languages

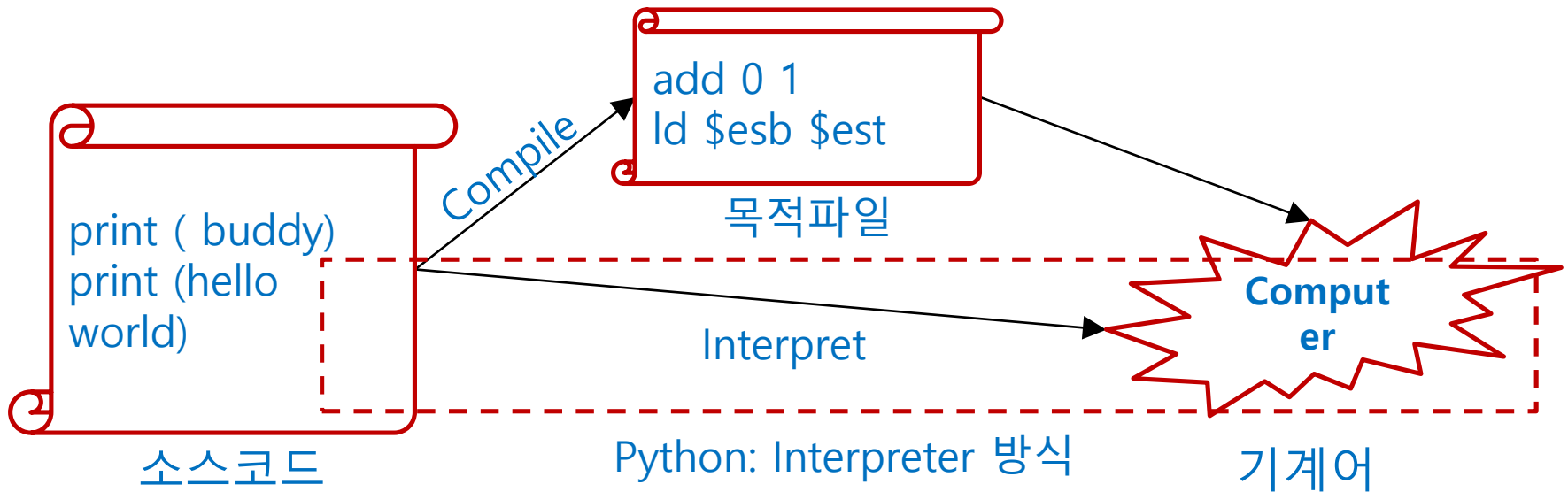
- Interpreters

Python

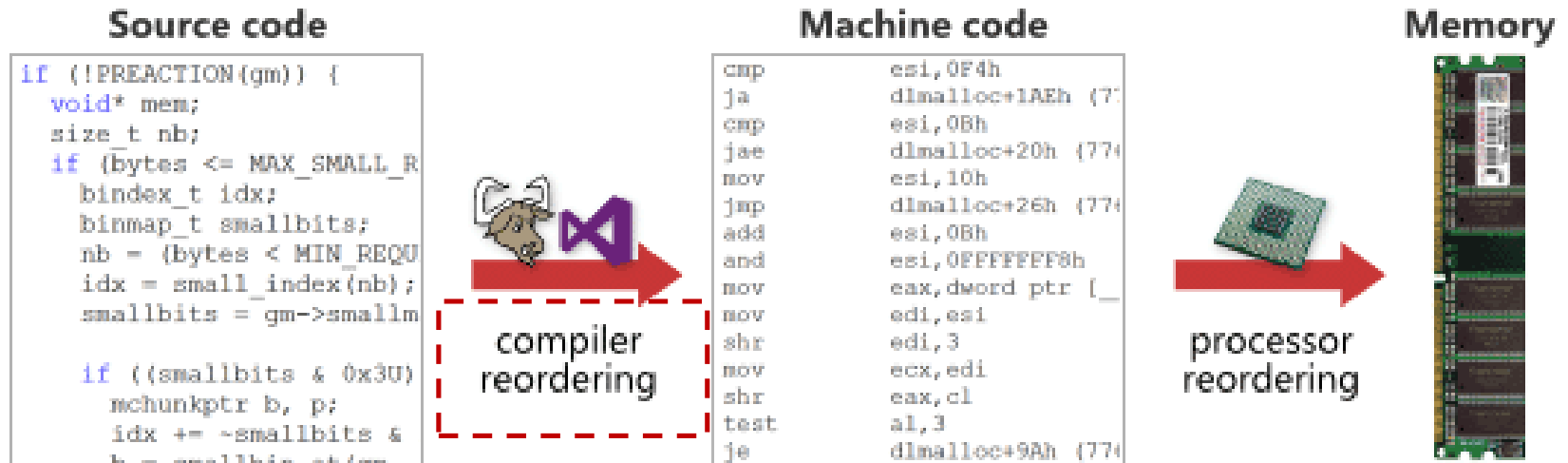
- 한 줄 씩 프로그램언어를 읽고 기계어로 번역하여 실행하는 방식

- Compiler

- 실행 전에 프로그램언어를 모두 기계어로 번역 해놓고 실행 하는 방식



# SW, HW

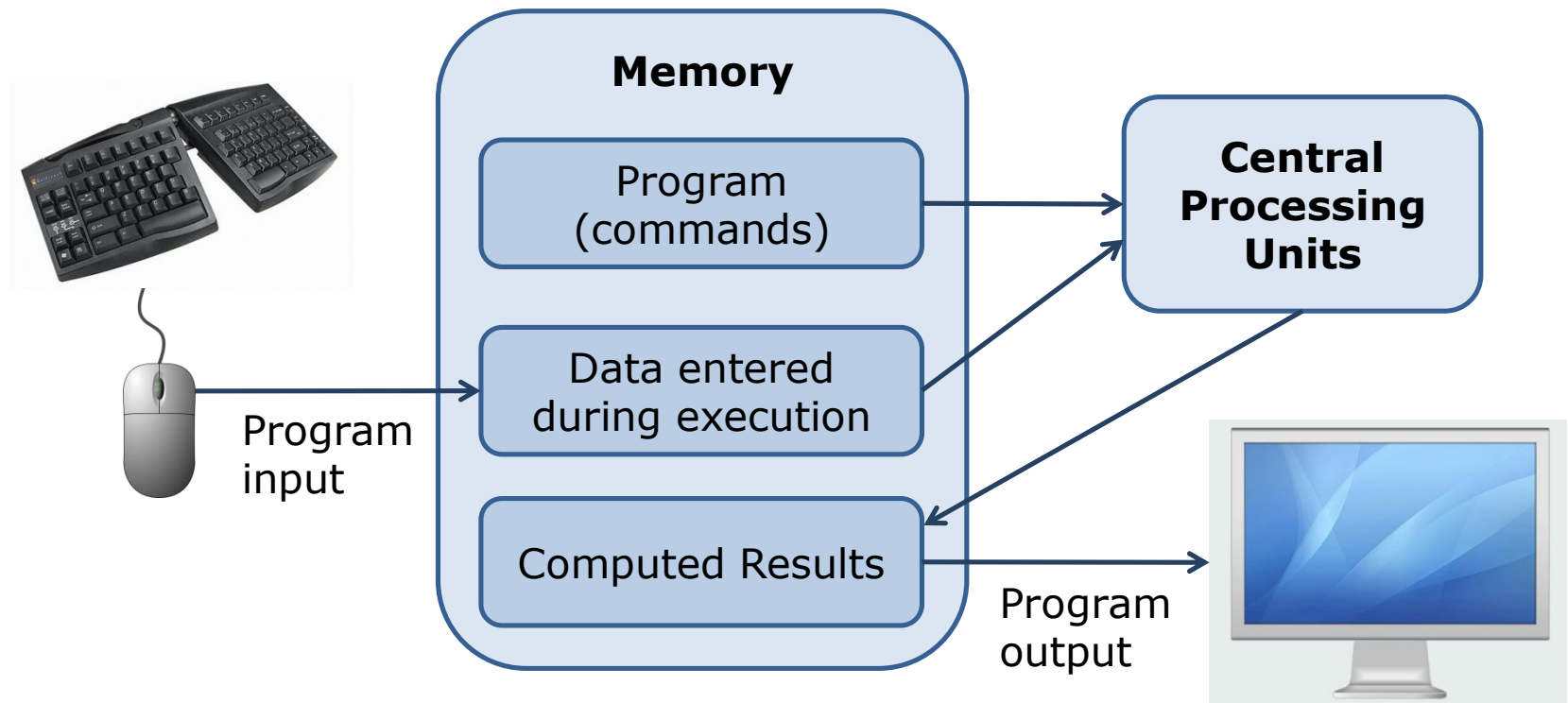


소스코드에서 기계어로 변환된 SW은 HW를 동작 시킴



# Flow of information during Program Execution

---



# Python(파이썬)

- 1990 네덜란드 출신 프로그래머 Guido van Rossum가 발표한 프로그래밍 언어



Guido van  
Rossum



몬티 파이트 비행 서커스

TV 프로그램

비단뱀 [Pythonidae](#)



# Features of Python

---

- Simple
  - 엄청난 개발자 층을 갖게 했고,수많은 라이브러리를 만들어 냈다



# Features of Python

- Cross platform

기존 언어들



1972



1985



1995



2000

운영체제



Ubuntu



Red hat



macOS

Python은 모든 언어와 함께 동작 할 수 있고  
다양한 운영체제 환경에서도 실행 될 수 있다





# Python 활용분야

- Web server

- 적게는 수십, 많게는 수천의 요청을 받아 웹페이지를 서비스하는 컴퓨터



<http://flask.pocoo.org/>



<https://www.djangoproject.com/>



Google 검색

I'm Feeling Lucky

Google 제공 서비스: English

- Crawling

- Web page에서 필요한 데이터만 추출하는 것



An open source and collaborative framework for extracting the data you need from websites. In a fast, simple, yet extensible way.

<https://scrapy.org/>



Beautiful Soup

<https://www.crummy.com/software/BeautifulSoup/>



# Python 활용분야

---

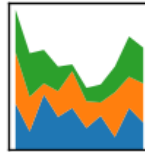
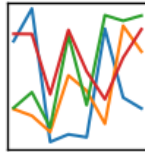
- Data science

- Big Data, Machine learning 등 데이터를 분석해서 활용하는 분야

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$

<https://pandas.pydata.org/>



<http://www.numpy.org/>



<https://www.scipy.org/>



<http://scikit-learn.org/>

- Artificial Intelligence

- Deep learning : 사람의 사고방식을 컴퓨터에게 가르치는 기계학습의 한 분야



<https://www.tensorflow.org/>



# 개발환경 설치

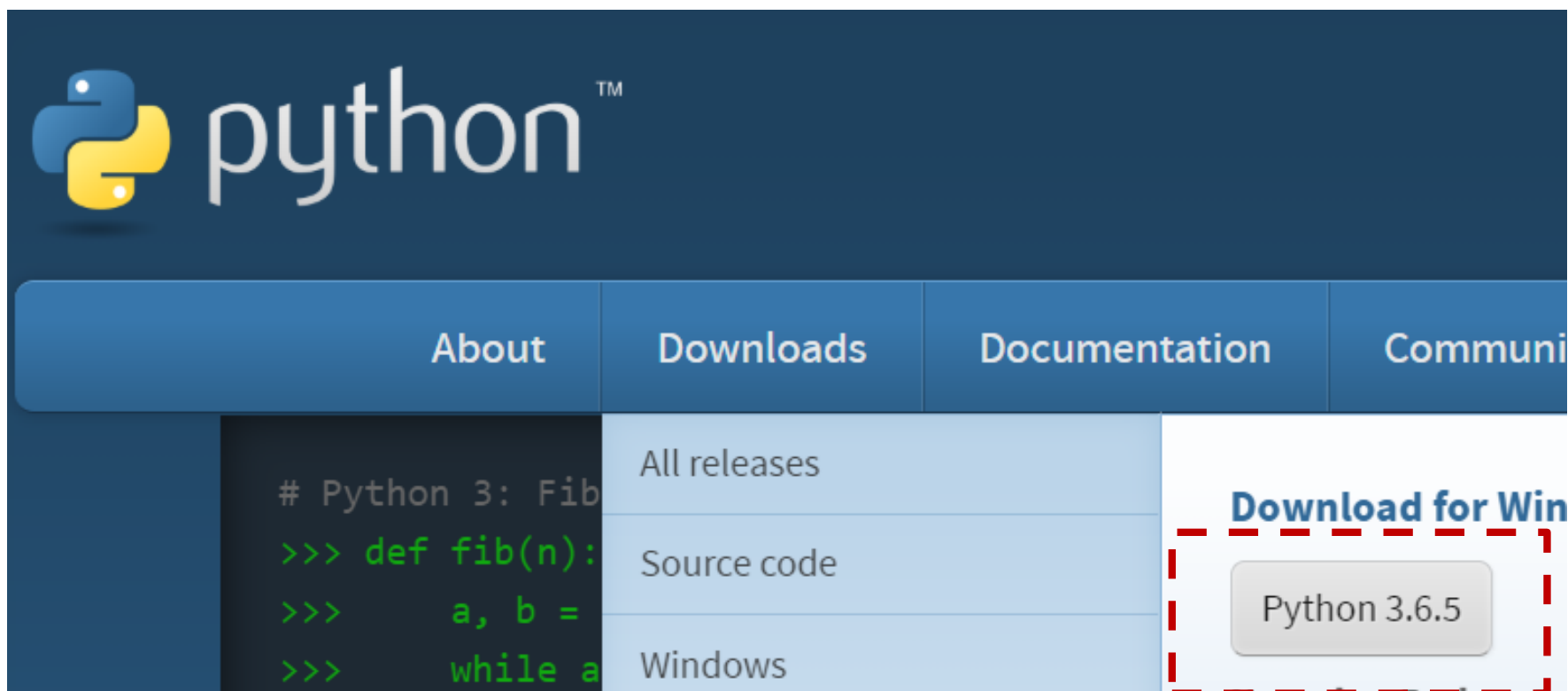
---

- Python 3
- Anaconda



# Python 3 다운로드

- <https://www.python.org>



# Python 3 설치

Python 3.5.1 (64-bit) Setup



## Install Python 3.5.1 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

→ Install Now

C:\Users\Username\AppData\Local\Programs\Python\Python35

Includes IDLE, pip and documentation  
Creates shortcuts and file associations

② 선택

→ Customize installation

Choose location and features

① 반드시 선택

☒ Install launcher for all users (recommended)

☒ Add Python 3.5 to PATH

Cancel

python  
for  
windows



# Anaconda5 다운로드

---

- <https://www.anaconda.com/download/>

Anaconda 5.2 For Windows Installer

Python 3.6 version \*

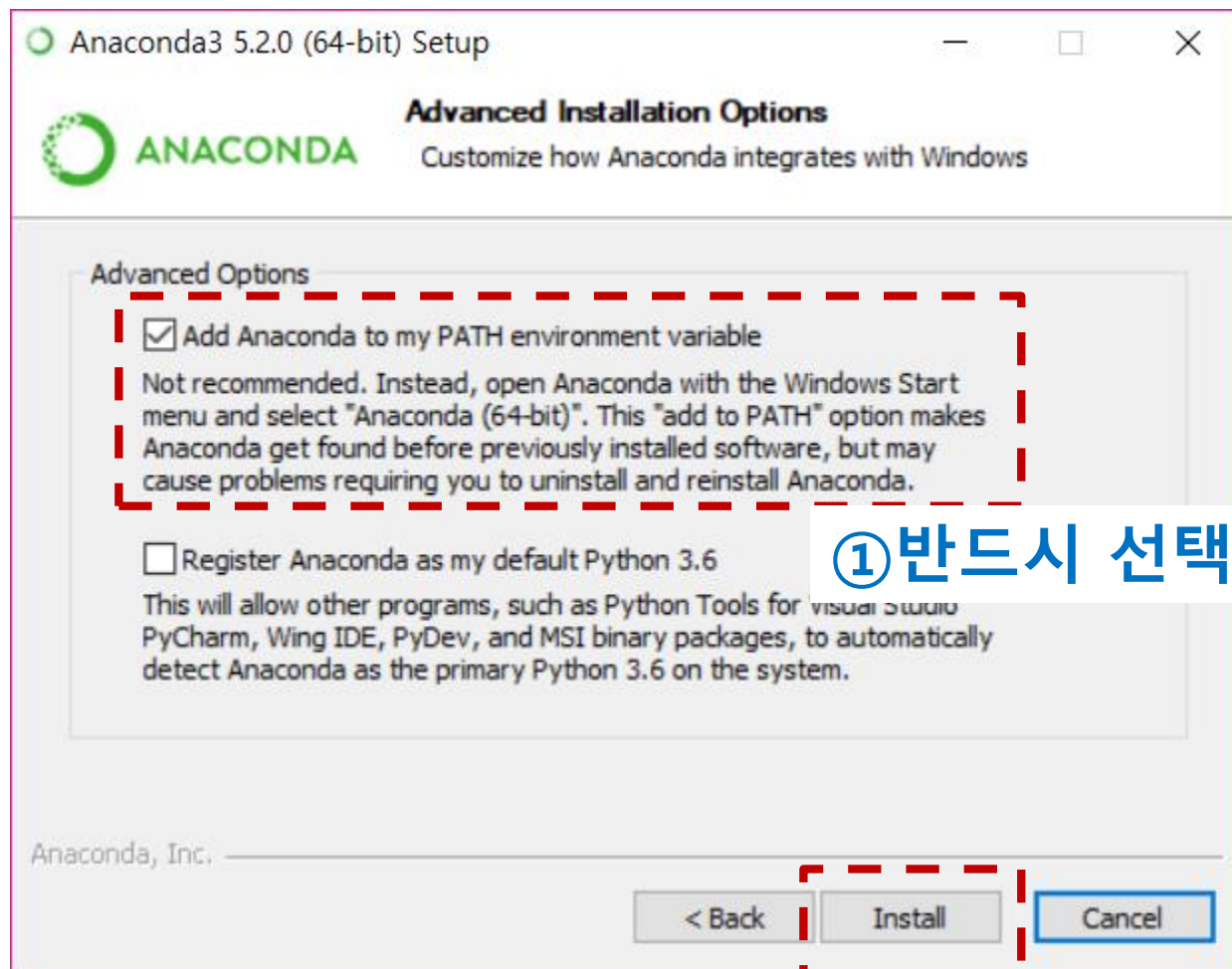
↓ Download

[64-Bit Graphical Installer \(631 MB\)](#) ?

[32-Bit Graphical Installer \(506 MB\)](#)



# Anaconda5 설치



① 반드시 선택

② 선택



# 편집기 Jupyter notebook 설치

The image shows a Windows Settings window with the '실행' (Task Scheduler) section selected. A red dashed box highlights the '열기(O):' (Open) field, which contains the text 'cmd'. An orange arrow points from this field to a Command Prompt window below. The Command Prompt window shows the command 'pip install jupyter' being executed, with the output indicating that the requirements are already satisfied.

실행 > cmd 입력

"pip install jupyter" 입력

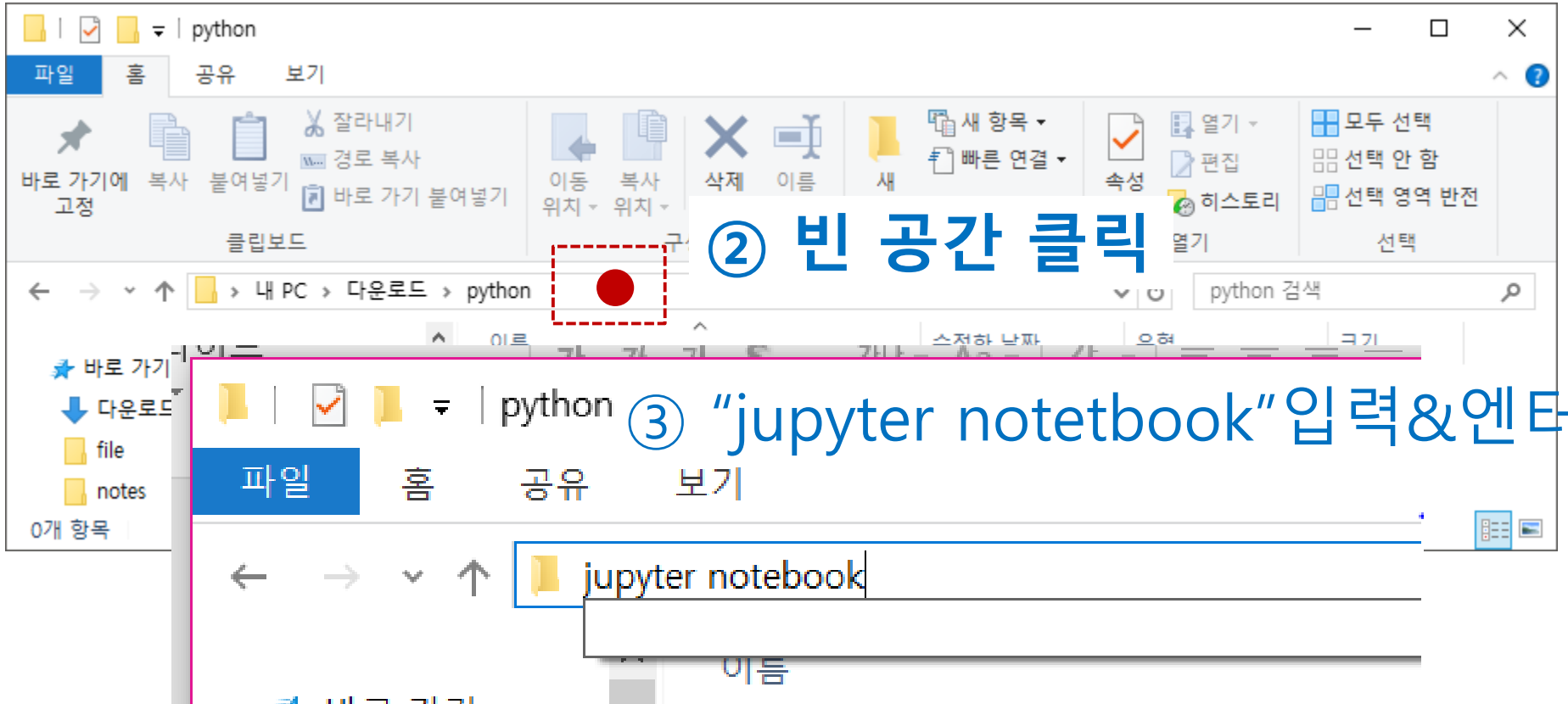
```
Microsoft Windows [Version 10.0.16299.371]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\mailo>pip install jupyter
Requirement already satisfied: jupyter in c:\Users\mailo\AppData\Local\Programs\Python\Python36-32\lib\site-packages
Requirement already satisfied: jupyter-console in c:\Users\mailo\AppData\Local\Programs\Python\Python36-32\lib\site-packages (from jupyter)
Requirement already satisfied: ipykernel in c:\Users\mailo\AppData\Local\Programs\Python\Python36-32\lib\site-packages (from jupyter)
```



# 개발 작업환경

- ① 작업 디렉터리 [python] 폴더를 생성하고 들어가기  
예제) C:\Users\{UserName}\Downloads\python



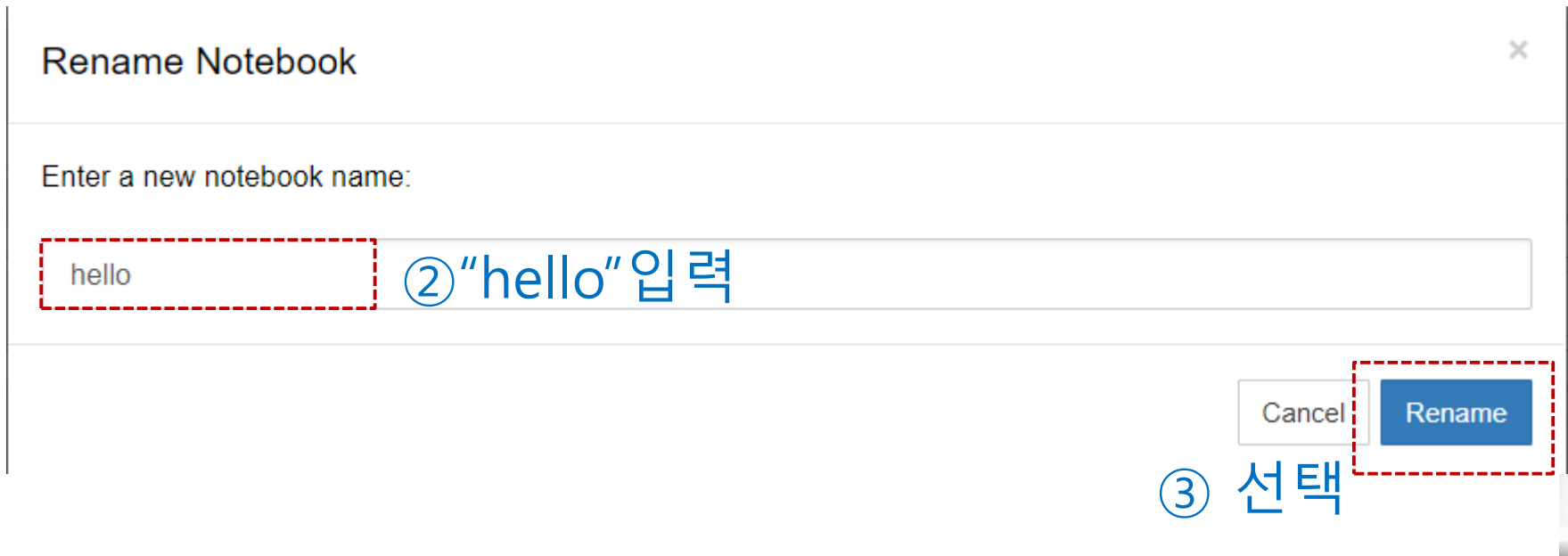
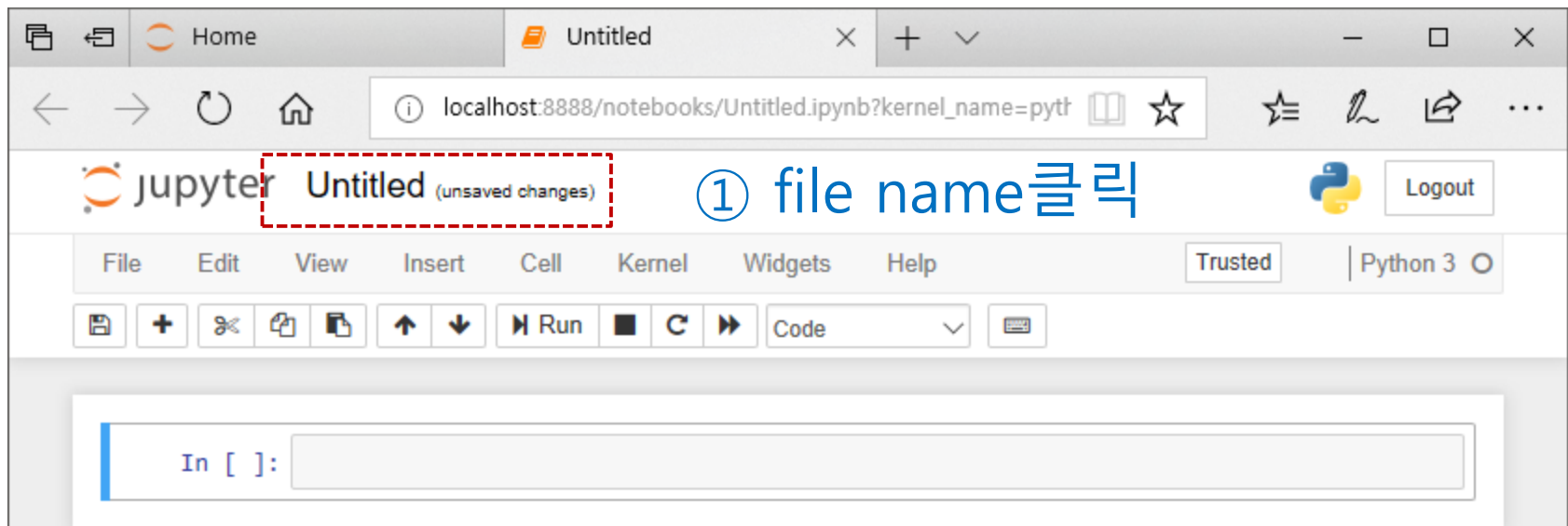
# Jupyter notebook

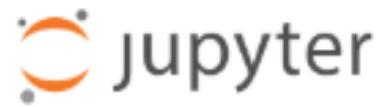
The screenshot shows the Jupyter Notebook web interface in a browser window. The address bar shows 'localhost:8888/tree'. The interface has tabs for 'Files', 'Running', and 'Clusters'. Below these, there's a section for 'Select items to perform actions on them.' with buttons for 'Upload', 'New', and a refresh icon. The 'New' button is highlighted with a red dashed box and labeled with a blue circle and text '① "New" 선택'. Below this, there's a table header with 'Name' and 'Last Modified'. The table is empty, with the text 'The notebook list is empty.' below it. A dropdown menu is open from the 'New' button, showing options: 'Notebook:', 'Python 3', 'Other:', 'Text File', 'Folder', and 'Terminal'. The 'Python 3' option is highlighted with a red dashed box and labeled with a blue circle and text '② "Python 3" 선택'.

① "New" 선택

② "Python 3" 선택





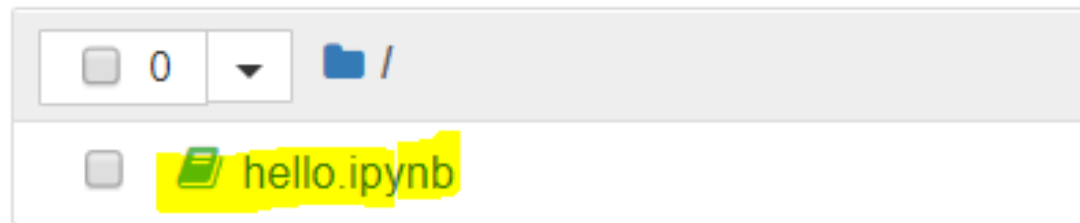


Files

Running

Clusters

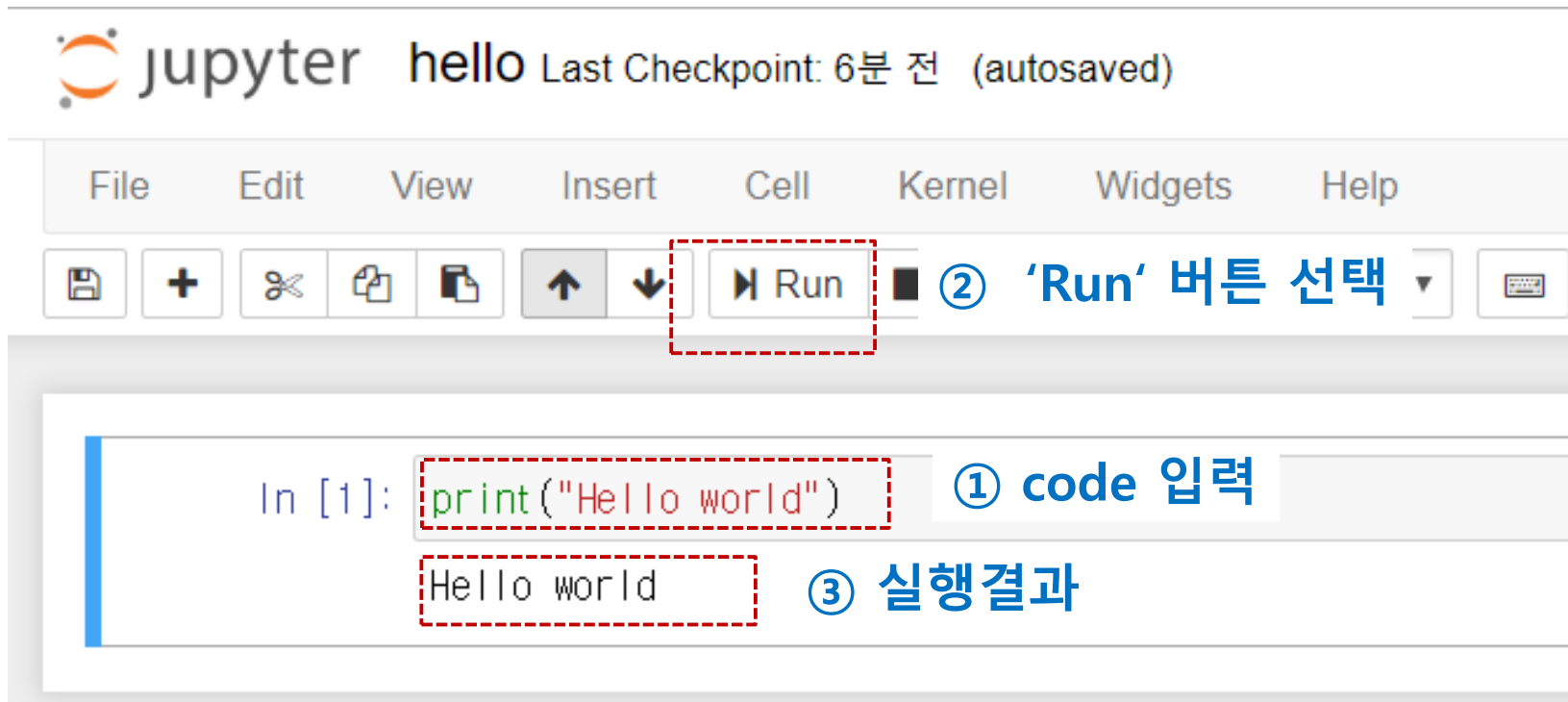
Select items to perform actions on them.



① file name이 바뀐 것을 확인



# “Hello World” 화면 출력하기



The screenshot displays the JupyterLab interface. At the top, the Jupyter logo is followed by the text 'jupyter hello' and 'Last Checkpoint: 6분 전 (autosaved)'. Below this is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. Under the menu bar is a toolbar with icons for saving, adding, deleting, copying, pasting, and running. The 'Run' button, which has a play icon and the text 'Run', is highlighted with a red dashed box and labeled with a blue circled '2' and the text '“Run” 버튼 선택'. Below the toolbar is a code cell. The code cell has a blue vertical bar on the left. Inside the cell, the prompt 'In [1]:' is followed by the code 'print("Hello world")', which is highlighted with a red dashed box and labeled with a blue circled '1' and the text 'code 입력'. Below the code, the output 'Hello world' is displayed, highlighted with a red dashed box and labeled with a blue circled '3' and the text '실행결과'.

jupyter hello Last Checkpoint: 6분 전 (autosaved)

File Edit View Insert Cell Kernel Widgets Help

② ‘Run’ 버튼 선택

In [1]: `print("Hello world")` ① code 입력

Hello world ③ 실행결과



## 출력 - print

---

print ('Hello world')

함수명

인자값(Argument)



'Hello World'을 화면에 출력해라!

- Function (함수)
  - 프로그램 명령문들의 모임
  - 특정 목적을 위해 작성된 작은 프로그램
  - 재사용 목적
- 함수 실행(Function Call) : 함수이름( )



# print

---

>>> print('Hello world') 작은 따옴표

**Hello World**

>>> print("Hello world") 큰 따옴표

**Hello World**

>>> print('Hello' + 'World') 두개 문자열 출력하기

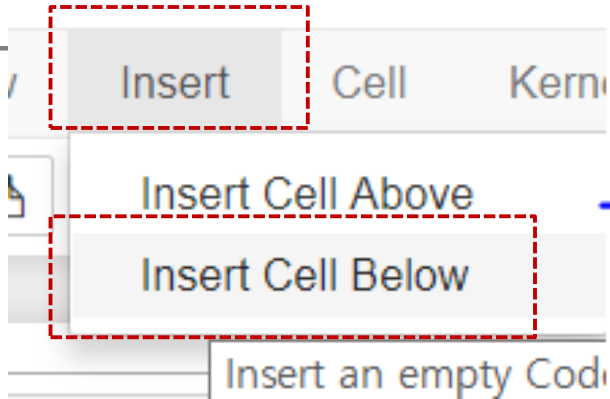
**Hello World**

- String(문자열) : 문자(character)들의 모임

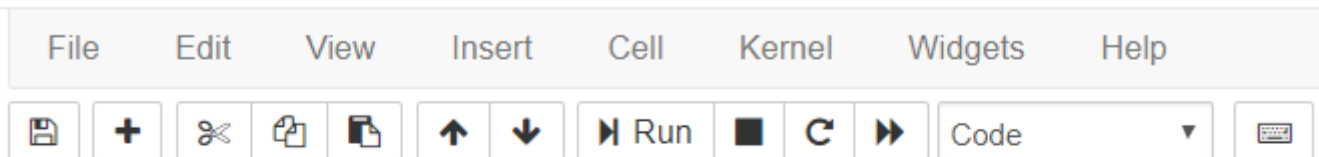


# 새로운 Cell로 편집창 확장하기

① [Insert] / [Insert Cell Below] 선택



jupyter hello Last Checkpoint: 11분 전 (unsaved changes)

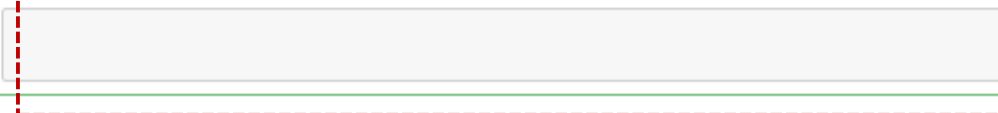


```
In [1]: print("Hello world")
```

Hello world

② 새로운 Cell생성

```
In [ ]:
```





# 입력-input

- 화면에서 사용자로부터 값을 입력 받아 출력하기



② 'Run' 버튼 선택

In [3]:

```
value = input('값을 입력하세요: ')\nprint (value)
```

① code 입력

```
값을 입력하세요: 2018\n2018
```

③ 실행결과



# 입력-input

---

value=input ('값을 입력하세요:')

함수명

인자값(Argument)



1. 인자 값을 화면에 출력하고 사용자로부터 입력 받아라
2. 그 입력 값을 반환 값으로 돌려줘라



# 기본 제공 함수: Built-in 함수

Built-in Functions				
abs()	dict()	help()	min()	setattr()
all()	dir()	hex()	next()	slice()
any()	divmod()	id()	object()	sorted()
ascii()	enumerate()	input()	oct()	staticmethod()
bin()	eval()	isinstance()	open()	str()
bool()	exec()	화면입력	ord()	sum()
bytearray()	filter()		pow()	super()
bytes()	float()	issubclass()	print()	tuple()
callable()	format()	iter()	property()	type()
chr()	frozenset()	len()	화면출력	vars()
classmethod()	getattr()	locals()		zip()
compile()	globals()	map()	reversed()	__import__()
complex()	hasattr()	max()	round()	
delattr()	hash()	memoryview()	set()	

Built-in함수는 언제나 어디서나 사용(call)할 수 있다

