

1장. 파이썬?

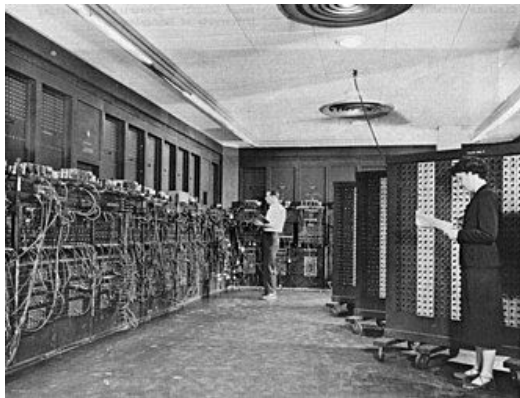
- 파이썬기초 -

컴퓨터 시초

- 계산기



컴퓨터 시초



컴퓨터에 적절한 입력을 주고 원하는 출력을 얻기
위해서 computer language를 사용

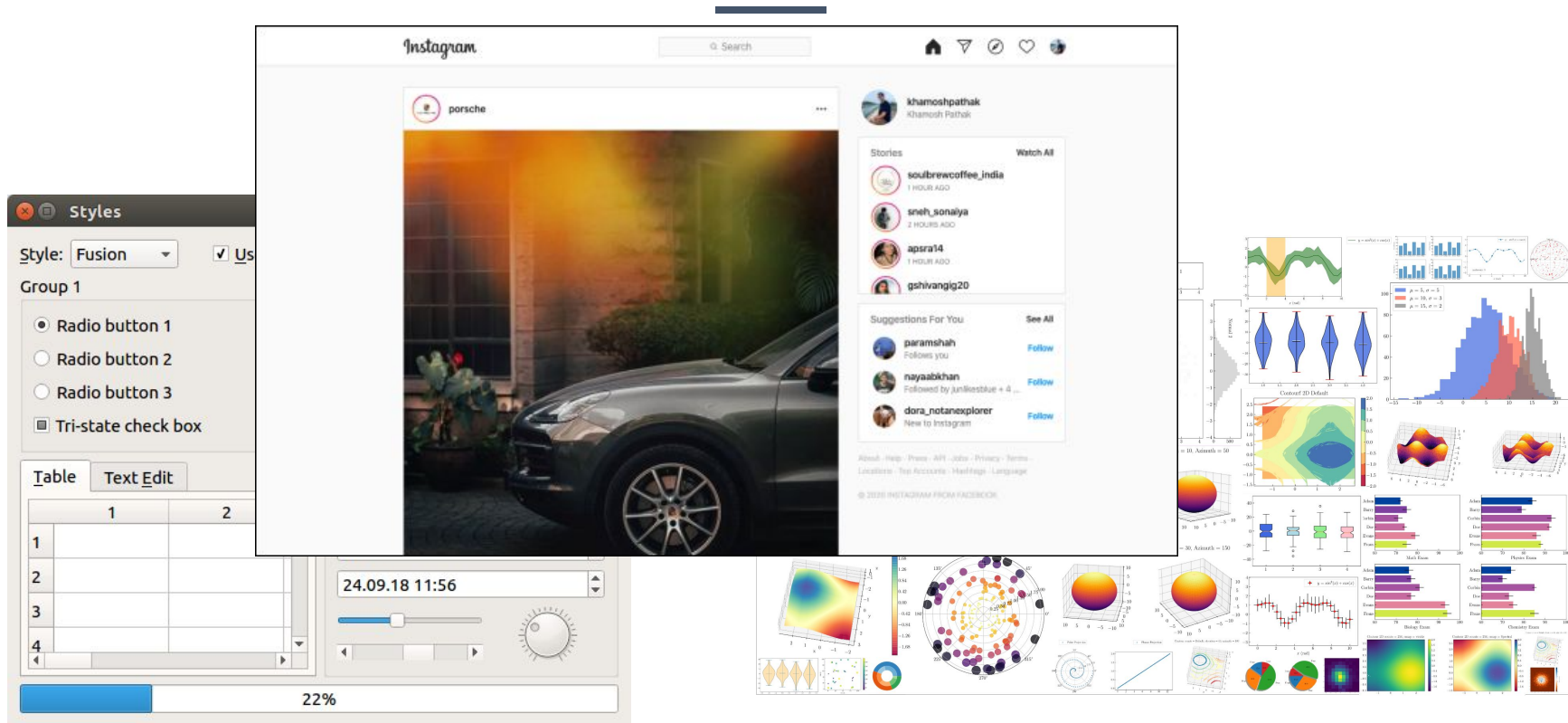
➡ 프로그래밍



복잡한 연산과 메모리를 활용하기 위한 방법?

Language

원하는 출력 예시 (made by Python)



프로그래밍 언어

- 걸작을 만드는 도구



<https://www.dreamstime.com/stock-photo-sculptor-carving-working-gypsum-bust-sculpture-image68258481>

2장. 파이썬 시작하기

- 파이썬기초 -

Reference



이엑스퍼트출판

<https://wikidocs.net/book/1>



<https://wikidocs.net/book/2944>

아나콘다?

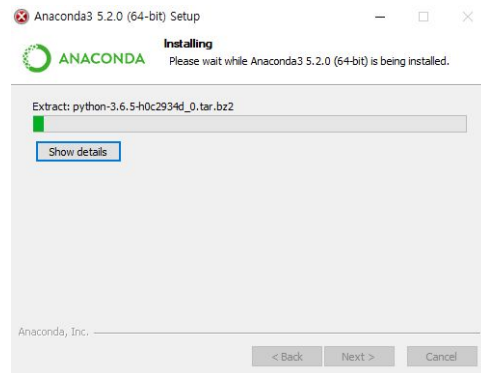
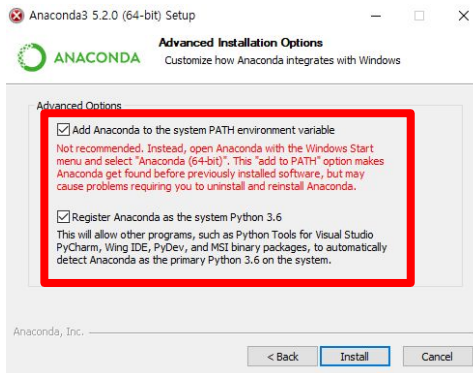
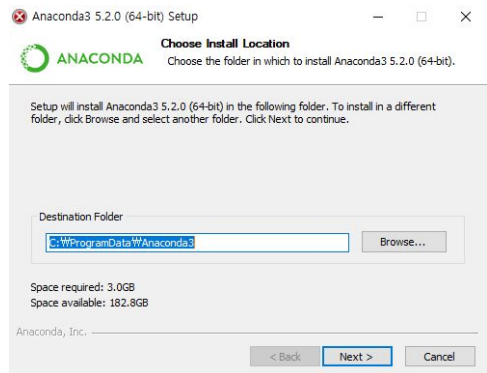
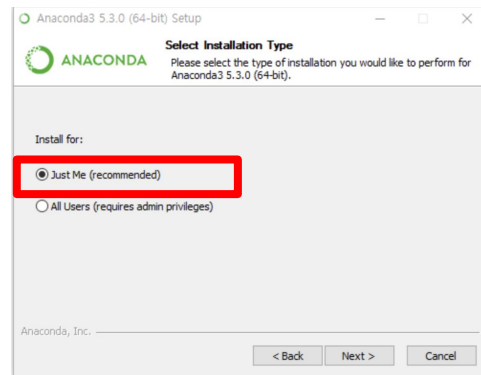
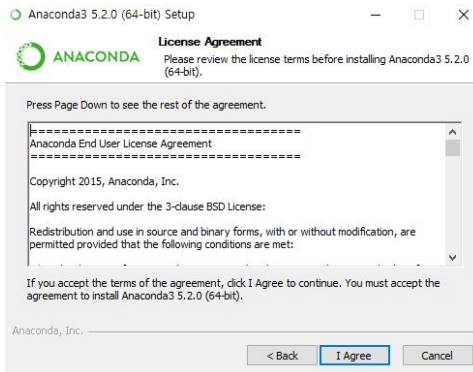
1. 파이썬 내장
2. 과학계산을 위한 패키지 및 툴 포함



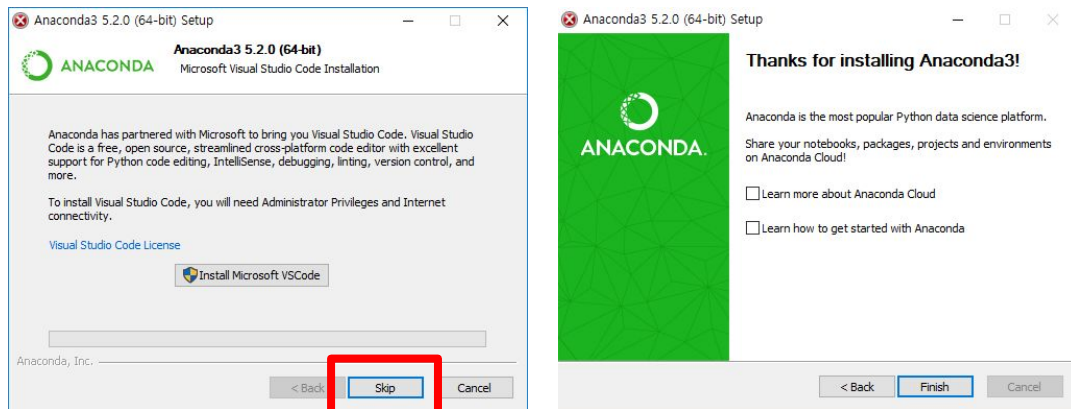
3. 가상환경 제공
(파이썬 다양한버전, 다양한 라이브러리 테스트)

<https://www.anaconda.com/products/individual>

아나콘다 설치

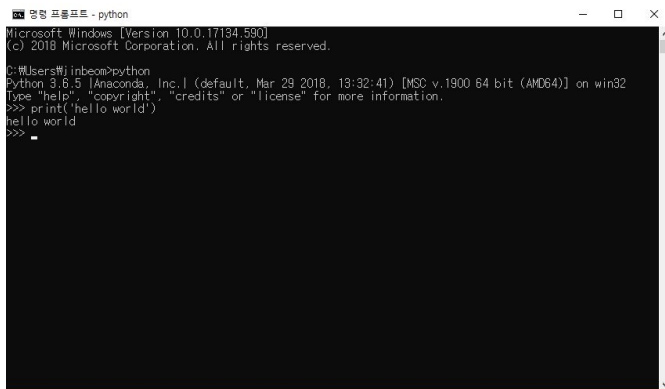
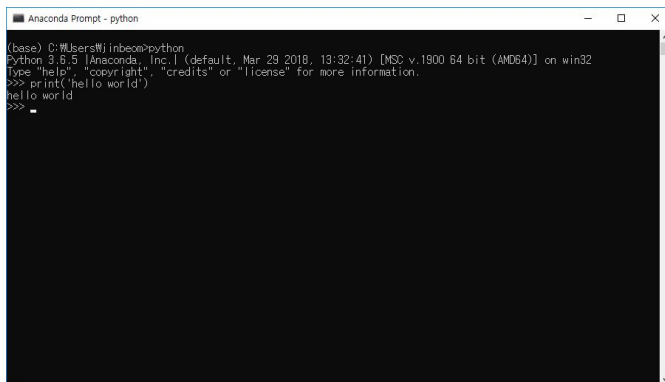
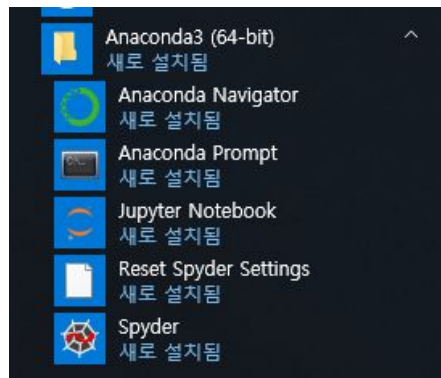


아나콘다 설치완료



IDE (Integrated Development Environment) : 효율적인 소프트웨어 개발을 위한 통합개발환경

아나콘다 설치완료



설치완료 테스트

python

>>> print('hello world')

Visual Studio Code?

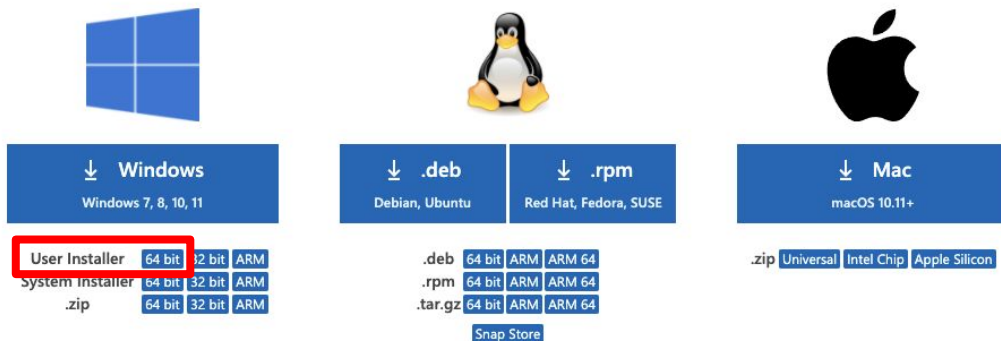
1. IDE(통합 개발 환경)
2. 자동 완성 및 린팅(정적 코드 분석)
3. 다양한 패키지 및 툴 지원 - PyQt, Jupyter notebook, 원격 개발
4. 다양한 테마

<https://code.visualstudio.com/download>

visual studio code 설치

Download Visual Studio Code

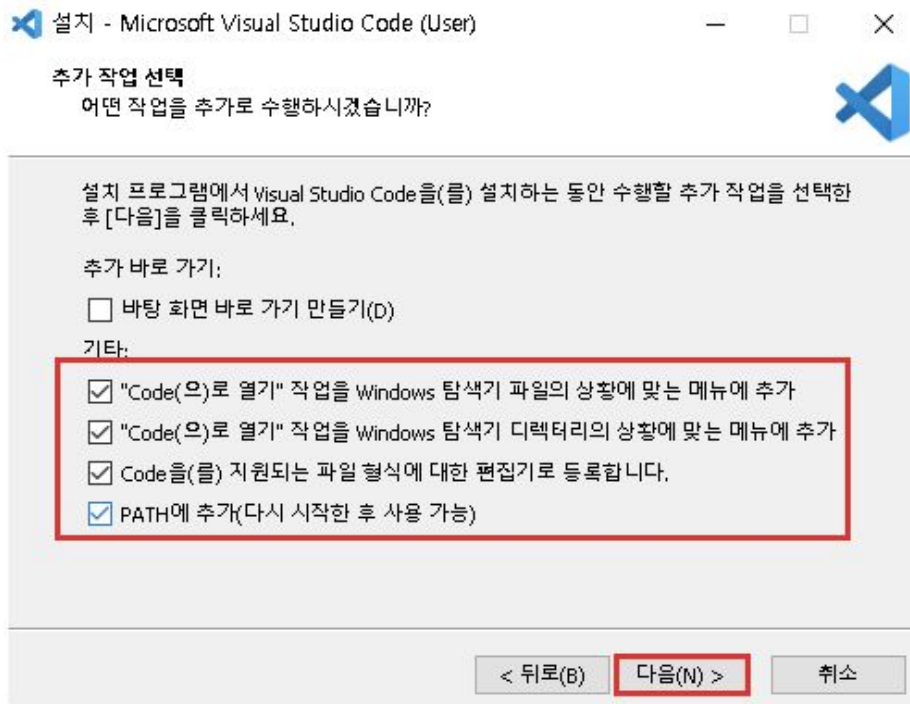
Free and built on open source. Integrated Git, debugging and extensions.



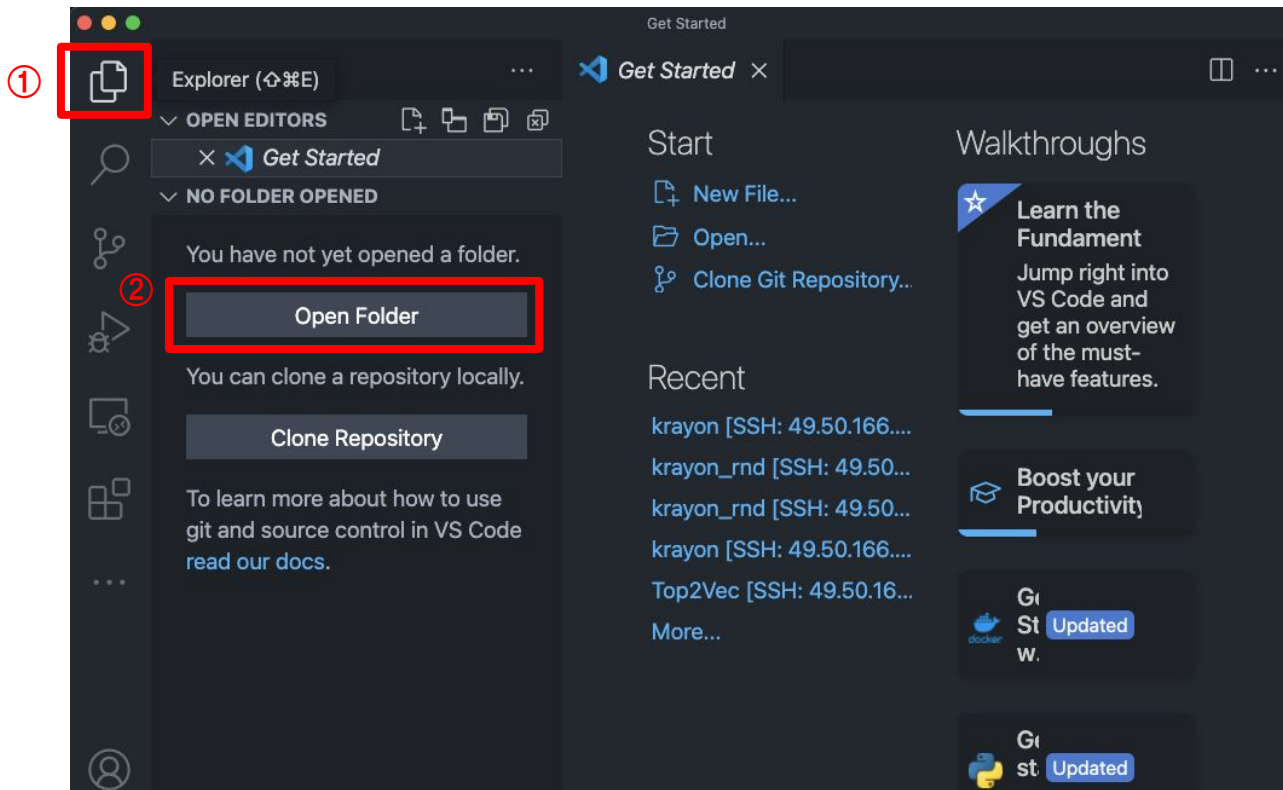
The image shows the Visual Studio Code download page. It features three main sections: Windows, Linux, and Mac. The Windows section has a blue button with a download icon and the text 'Windows' and 'Windows 7, 8, 10, 11'. Below this button, there are three rows of links: 'User Installer' (highlighted with a red box), 'System Installer', and '.zip'. Each row has four sub-links: '64 bit', '32 bit', 'ARM', and 'ARM 64'. The Linux section has a penguin icon and two buttons for '.deb' (Debian, Ubuntu) and '.rpm' (Red Hat, Fedora, SUSE). Below these are links for '.deb', '.rpm', and '.tar.gz', each with sub-links for '64 bit', 'ARM', and 'ARM 64'. There is also a 'Snap Store' link. The Mac section has an Apple logo and a button for 'Mac' (macOS 10.11+). Below this is a '.zip' link with sub-links for 'Universal', 'Intel Chip', and 'Apple Silicon'.

Platform	Installer Type	Architecture	Link	
Windows	User Installer	64 bit	64 bit ARM ARM 64	
		32 bit	32 bit ARM ARM 64	
		ARM	ARM ARM 64	
	System Installer	64 bit	64 bit ARM ARM 64	
		32 bit	32 bit ARM ARM 64	
		ARM	ARM ARM 64	
	.zip	64 bit	64 bit ARM ARM 64	
		32 bit	32 bit ARM ARM 64	
		ARM	ARM ARM 64	
	Linux	.deb	64 bit	64 bit ARM ARM 64
			ARM	ARM ARM 64
		.rpm	64 bit	64 bit ARM ARM 64
ARM			ARM ARM 64	
.tar.gz		64 bit	64 bit ARM ARM 64	
		ARM	ARM ARM 64	
Snap Store				
Mac		.zip	Universal	Intel Chip Apple Silicon

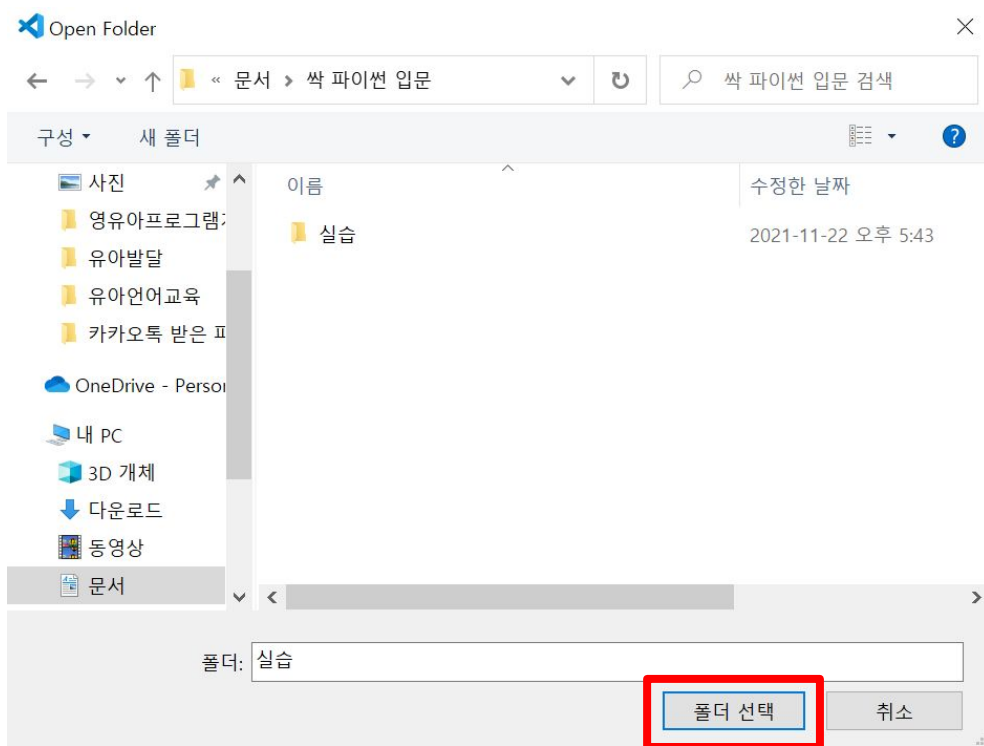
visual studio code 설치



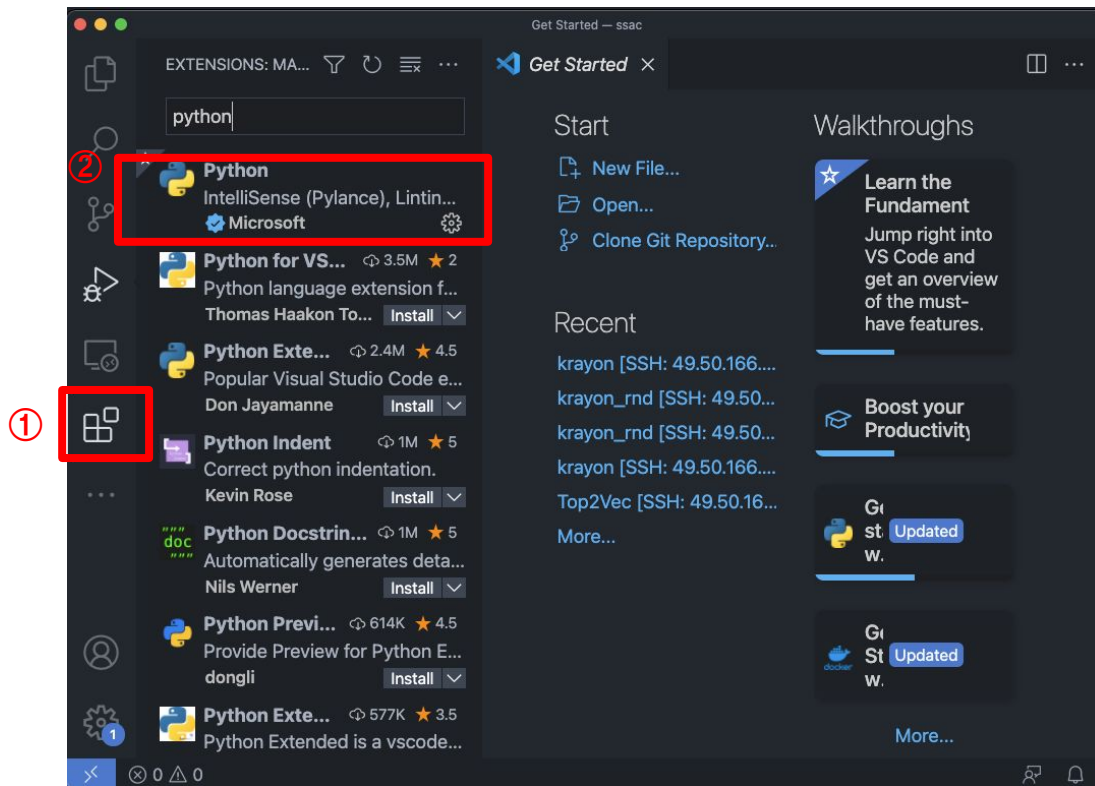
visual studio code 시작하기



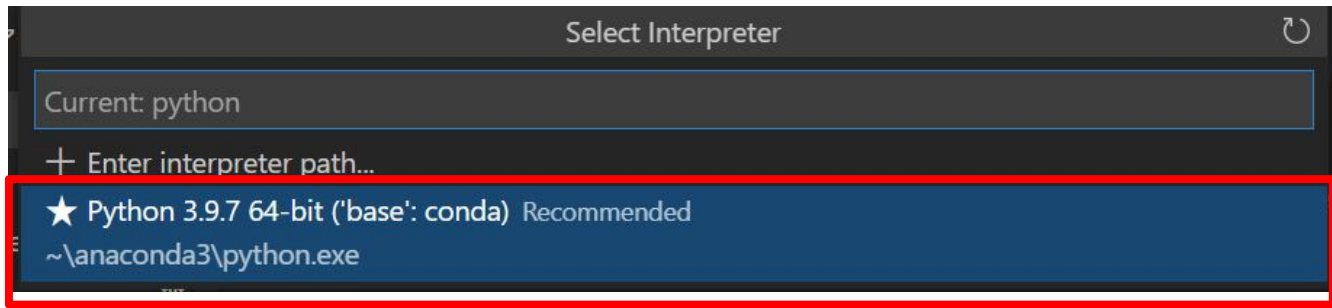
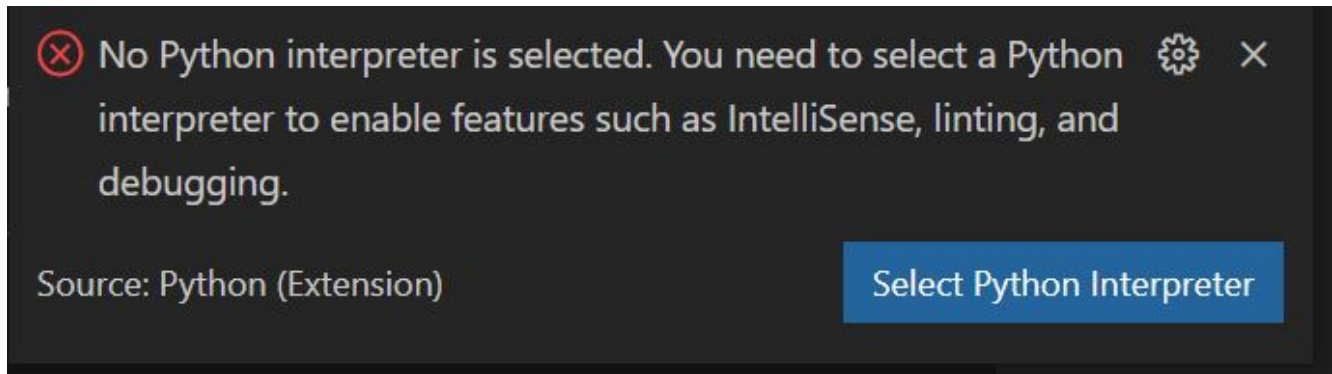
visual studio code 시작하기



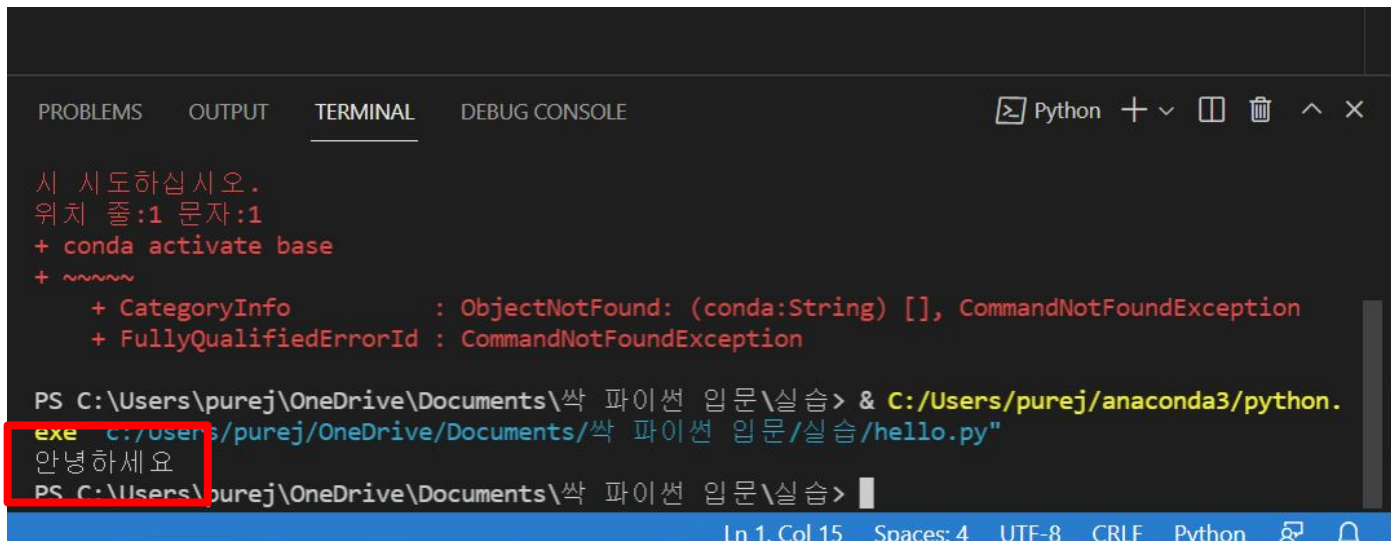
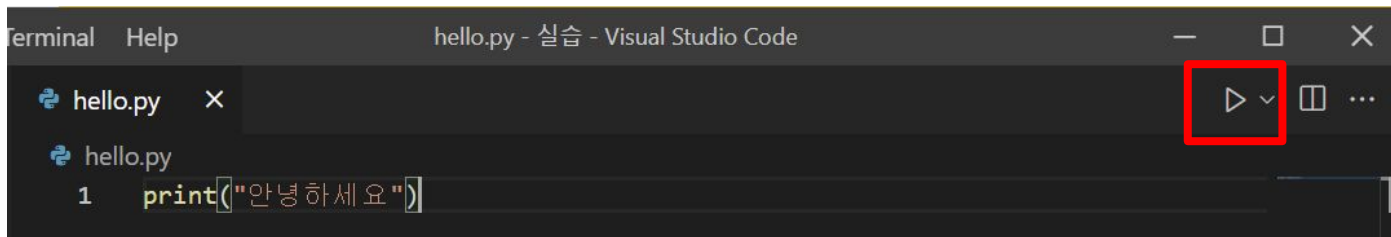
visual studio code 시작하기 - python



visual studio code 시작하기 - python



visual studio code 시작하기 - python



간단한 실습 - 사용자 입력, 출력

```
x = input("입력해주세요 : ")  
print(x)
```

소스코드

02-2 간단한 실습.ipynb

간단한 실습 - 주식

소스 흐름을 쉽게 파악할수 있도록 설명

한줄수식

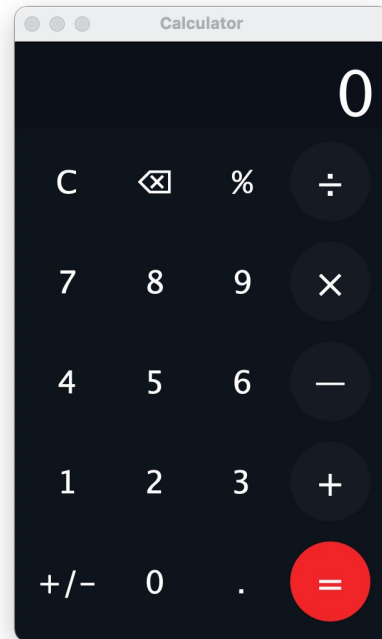
“””

여러줄주식
여러줄주식

“””

PyQt?

- Qt의 레이아웃에 Python의 코드를 연결하여 GUI 프로그램을 만들 수 있게 해주는 프레임워크
- 실행 (<https://wikidocs.net/35479>)



PyQt - 실습 1

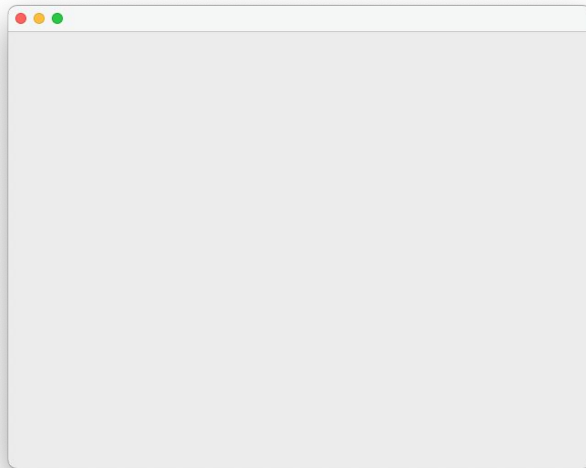
```
import PyQt5.QtWidgets as qtw
import PyQt5.QtGui as qtg

class MainWindow(qtw.QWidget):
    def __init__(self):
        super().__init__()

        self.show()

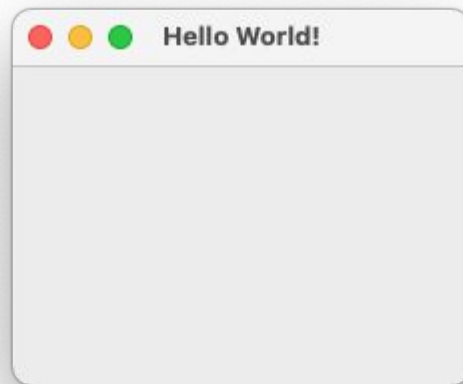
app = qtw.QApplication([])
mw = MainWindow()

app.exec_()
```



PyQt - 실습 2

```
class MainWindow(qtw.QWidget):  
    def __init__(self):  
        super().__init__()  
  
        self.setWindowTitle("Hello World!")  
        self.setLayout(qtw.QVBoxLayout())  
  
    self.show()
```



PyQt - 실습 3

```
class MainWindow(qtw.QWidget):  
    def __init__(self):  
        super().__init__()  
  
        self.setWindowTitle("Hello World!")  
        self.setLayout(qtw.QVBoxLayout())  
  
        my_label = qtw.QLabel("안녕 친구야!\n네 이름이  
뭐야?")  
        self.layout().addWidget(my_label)  
  
        self.show()
```



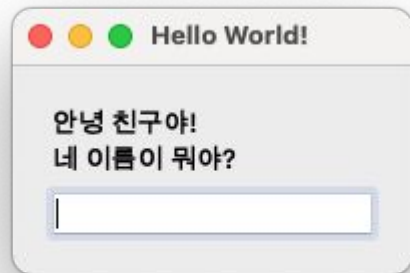
PyQt - 실습 4

```
my_label = QtWidgets.QLabel("안녕 친구야!\n네 이름이  
뭐야?")
```

```
self.layout().addWidget(my_label)
```

```
my_entry = QtWidgets.QLineEdit()  
my_entry.setObjectName("name_field")  
my_entry.setText("")  
self.layout().addWidget(my_entry)
```

```
self.show()
```



PyQt - 실습 5

```
my_entry = QtWidgets.QLineEdit()
my_entry.setObjectName("name_field")
my_entry.setText("")
self.layout().addWidget(my_entry)

my_button = QtWidgets.QPushButton("확인")
self.layout().addWidget(my_button)
```

```
def press_it():
    print(my_entry.text())
    my_entry.setText("")
my_button.clicked.connect(press_it)
self.show()
```



3장. 자료형

- 파이썬기초 -

사칙연산

구분	예제	결과
덧셈	$1 + 2$	3
뺄셈	$1 - 2$	-1
곱셈	$5 * 2$	10
나눗셈	$5 / 2$	2.5
나눗셈 (몫)	$5 // 2$	2
나눗셈 (나머지)	$9 \% 2$	1
제곱	$2 ** 3$	8
괄호 (우선순위)	$(2 + 3) * 2$	10

숫자의 자료형

자료의 형태를 확인하는 함수

type(값)

```
>> type(10)
```

결과 : int

자료형

int - 정수 (-2,147,483,648 ~ 2,147,483,647)

float - 실수

bool - True or False

str - 문자

```
>> type(10.5)
```

결과 : float

서로 변환 가능

list - list

dict - dictionary

tuple - tuple

set - 집합

숫자형 변환

자료형 변환함수

int(값) - 정수로 변환

float(값) - 실수로 변환

```
>>> float(3) #정수를 실수로
```

결과 : 3.0

```
>>> int(3.5) #실수를 정수로
```

결과 : 3

```
>>> int(5 / 2) #형변환을 이용한 몫구하기
```

결과 : 2

실습01

원주율을 3.1415926535 라고 하면 반지름을 입력으로 받아서
원의 면적을 구하는 코드를 작성하시오.

원의 면적 = (반지름) x (반지름) x (원주율)

실행 :

반지름 : 20

결과 : 1256.6370614

4장. 변수

- 파이썬기초 -

변수

NamesandNumbers.com	
Wood River Valley	
522-7481	BATES Paul 118 Willow Rd.....Hailey 788-1206
788-3933	BATES Steve 105 Audubon Pl.....Hailey 788-6222
788-9263	BATES VICKY - INTERIOR MOTIVES PO Box 1820.....Sun Valley 788-5950
788-9933	BATHUM Roy 235 Spur Ln.....Ketchum 726-0722
578-0595	BATMAN.....See West Adam
788-8979	BATT Jeffrey & Camille.....726-7494
788-2515	BATTERSBY Patricia 116 Ritchie Dr.....Ketchum 726-8896
720-5661	BAUER Charlotte 621 Northstar Dr.....Hailey 788-4279
28-7219	BAUER CHARLOTTE LINDBERG
38-2317	Radiance Skin Care Studio.....Hailey 578-2214
	BAUER Matt 3340 Woodside Blvd.....Hailey 578-0703
	BAUER Rich.....720-0165

변수사용방법

$$X = 10$$

변수명

값

$$X, Y, Z = 10, 20, 30$$

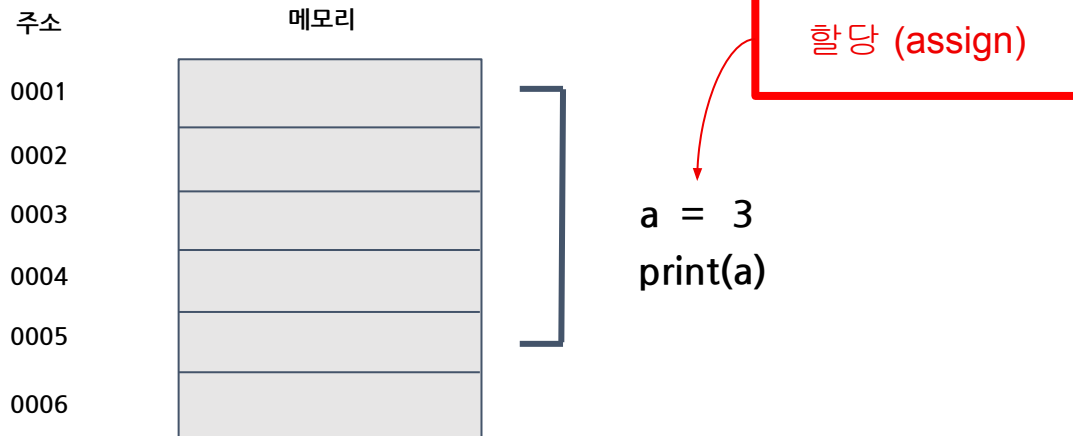
변수명

값

영문, 숫자 사용가능
 대소문자 구분
 숫자부터 시작할수 없음
 특수문자 사용불가 (+, -, *, /, \$ 등)
 파이썬의 키워드 사용불가

변수란?

하나의 데이터를 저장할 수 있는 메모리 공간, 변할수있는 수



프로그래밍이란 데이터(숫자, 문자열)를 기반으로 컴퓨터에 명령을 내리는 것인데 이 데이터를 저장하는 곳이 변수이다.

변수로 계산하기

```
>>> x = 20
>>> y = 30
>>> c = x + y
>>> c
결과 : 50
```

```
>>> x = 10
>>> x + 10
>>> x
결과 : 10
```

```
>>> x = 10
>>> x = x + 10
>>> x
결과 : 20
```

```
>>> x = 10
>>> x += 10
>>> x
결과 : 20
```

비어있는 변수 (Null)

None

아무값도없는

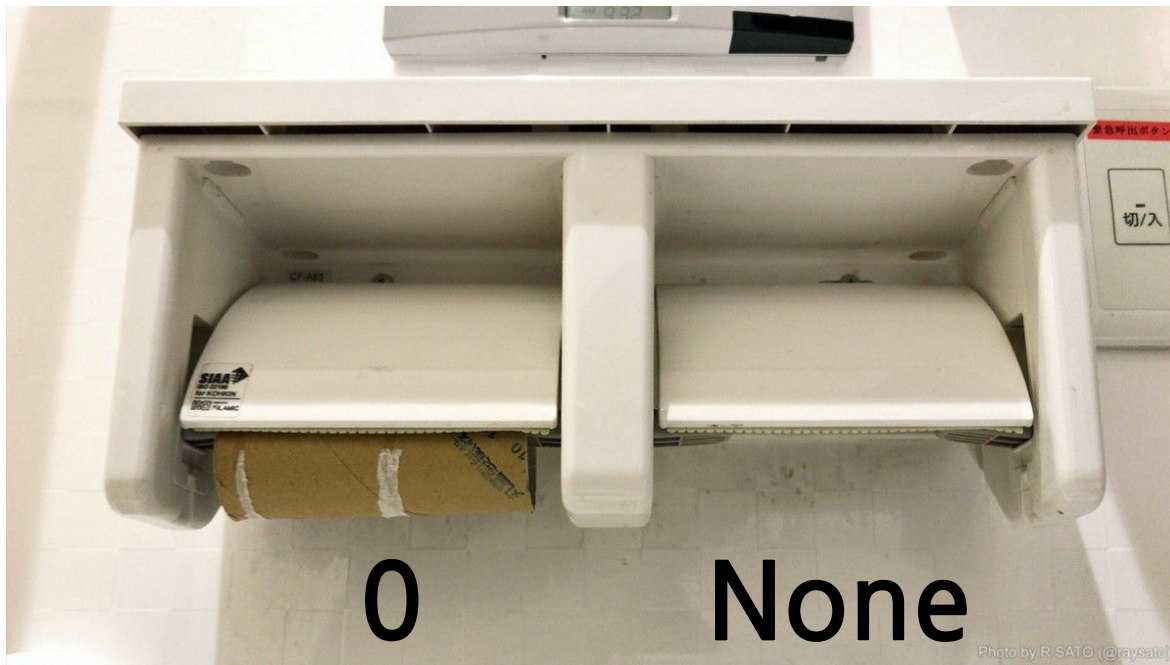
```
>>> x = None
```

```
>>> print(x)
```

결과 : None

```
>>> type(x)
```

결과 : None Type



실습01

사용자로부터 변수 x , 변수 y 를 입력받아
더하고 곱하는 프로그램을 작성하세요

실행 :

x 값을 입력해주세요 : 20

y 값을 입력해주세요 : 30

결과 :

50

5장. Boolean 과 비교, 논리연산자

- 파이썬기초 -

Boolean형

True

참

False

거짓

```
>>> print( True)
```

결과 : True

```
>>> print(False)
```

결과 : False

```
>>> print((type(True)))
```

결과 : bool

비교연산자

>

크다

>=

크거나같다

<

작다

<=

작거나같다

==

같다

!=

같지않다

연산결과는 Boolean

```
>>> print(3 > 1)
```

결과 : True

```
>>> print(10 != 10)
```

결과 : False

```
>>> print(type(10==10))
```

결과 : bool

논리연산자

and(&)

모두가 True 면 True
하나라도 False 면 False

>>> True and True

결과 : True

>>> True and False

결과 : False

>>> False and False

결과 : False

or (|)

하나만 True 여도 True
모두가 False 면 False

>>> True or True

결과 : True

>>> True or False

결과 : True

>>> False or False

결과 : False

not

반대

>>> not True

결과 : False

>>> not False

결과 : True

비교 + 논리 연산자

$5 > 10$ and $20 == 20$

False

True

False

$5 == 5$ or $10 != 10$

True

False

True

실습01

사용자로 점수를 3개 입력받아
모든 점수가 65점보다 클 경우 True
점수가 하나라도 65점이 안넘을 경우 False
를 출력하세요

실행 :

첫번째 점수 : 20

두번째 점수 : 70

세번째 점수 : 80

결과 :

False

실행:

첫번째 점수 : 70

두번째 점수 : 80

세번째 점수 : 90

결과:

True

6장. 문자형

- 파이썬기초 -

문자형

```
print(“안녕, 친구야”)
```

```
print(‘안녕, 친구야’)
```

```
print( “““안녕,  
친구야””” )
```


문자열 연산

“과일” + “사과”
과일사과

“과일” * 3
과일과일과일

Format 함수

“문자열 {0}, {1} 문자열”.format(값, 값)

```
>>> “I like {0}, {1} !”.format(“apple”, “banana”)
```

```
I like apple, banana
```

```
>>> “Number {0} {2} {1}”.format(1,2,3)
```

```
Number 1 3 2
```

```
>>> “Number {0} {0} {1}”.format(1,2,3)
```

```
Number 1 1 2
```

```
>>> “Number {} {} {}”.format(1,2,3)
```

```
Number 1 2 3
```

변수사용

```
>>> f1, f2 = “apple”, “banana”
```

```
>>> “I like {0}, {1} !”.format(f1, f2)
```

```
I like apple, banana
```

{0:.숫자f} - 자리수 맞추기

```
>>> “Number {:.2f}!”.format(3.1456)
```

```
Number 3.15!
```

f String

f“문자열 {변수}, {변수} 문자열”

```
>>> f1, f2 = “apple”, “banana”  
>>> f “I like {f1}, {f2} !!”  
I like apple, banana
```

{변수:.숫자f} - 자리수 맞추기

```
>>> pi = 3.141592  
>>> f"Number {pi:.2f}!"
```

변수와 문자열

```
print("Python is awesome programming language. You can get more information from  
https://docs.python.org/3/")
```

```
print("Python is awesome programming language." + "You can get more information from" +  
"https://docs.python.org/3/")
```

```
print("Python is awesome programming language.", "You can get more information from",  
"https://docs.python.org/3/")
```

변수사용

```
language = "JAVA"  
site = "https://docs.oracle.com/javase/8/docs/api/overview-summary.html"
```

```
print(f"{language} is awesome programming language. You can get more information from  
{site}")
```

문자열 함수소개

len()

문자 길이

```
>>> word = "apple banana"
>>> len(word) #문자길이
12
```

upper()

대문자로 바꾸기

```
>>> word.upper()
1RANG2 BANANA
```

lower()

소문자 바꾸기

```
>>> word.lower()
1rang2 banana
```

replace(바꿀문자, 새문자)

문자열 바꾸기

```
>>> word.replace("apple", "orange")
orange banana
```

실습01

(1) 문자를 입력받아 공백을 모두 제거하고 출력하세요

실행 :

아무 문자나 입력해주세요 : **안녕하세요 반갑습니다.**
안녕하세요반갑습니다.

(2) 이름과 점수3개를 입력받아 아래와 같이 출력하세요

실행 :

이름을 입력해 주세요 : **홍길동**
첫번째 점수를 입력해 주세요 : **80**
두번째 점수를 입력해 주세요 : **70**
세번째 점수를 입력해 주세요 : **60**
저의 이름은 홍길동 이고 총점은 210 입니다.

실습02

아래와 같이 이름을 입력받아 “안녕 {이름}” 형식으로 출력하세요



Hello World!

안녕 친구야!
네 이름이 뭐야?

경수

확인



Hello World!

안녕 경수!

확인