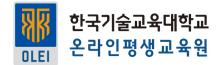


파이썬 시작하기



학습목표

- 컴파일 언어와 스크립트 언어에 대해 설명할 수 있다.
- 파이썬의 특징과 활용처에 대해 설명할 수 있다.
- <mark>파이썬 코딩 환경</mark>을 스스로 구축하고 코딩을 할 수 있다.

학습내용

- 컴파일 언어와 스크립트 언어의 이해
- 파이썬 언어의 이해와 특징
- 파이썬 코딩 환경 구축 및 실행하기

- 1 컴파일(Compile) 언어의 이해
 - 1 컴파일 언어란?

기계어로 컴파일 되는 언어 (컴퓨터가 이해할 수 있도록 번역)

- 컴파일 된 코드는 컴퓨터가 가장 이해하기 쉬운 형태로 변형됨
- 컴파일러는 플랫폼마다 다른 특징을 가지고 있음
 - ➡ 똑같은 소스코드일지라도 운영체제마다 따로 컴파일을 해야 함

2 컴파일 언어의 종류



C 언어, C ++, C#



자바(JAVA)

- ② 스크립트(Script) 언어의 이해
 - 1) 스크립트 언어란?

응용 소프트웨어를 제어하는 컴퓨터 프로그래밍 언어

- 연극의 대사 등이 적혀 있는 스크립트에서 유래
- 소스코드를 한 줄 한 줄 읽어 바로 바로 실행하는 인터프리터 방식
 - 소스코드를 컴파일하지 않고도 실행할 수 있는 프로그래밍 언어

'해석기'라는 뜻으로 프로그래밍 언어의 소스코드를 바로 실행하는 컴퓨터 프로그램 또는 환경

- ② 스크립트(Script) 언어의 이해
 - 2) 스크립트 언어 활용
 - 1 파이썬 프로그래밍에 활용
 - 2 리눅스의 쉘(Shell), 윈도우의 배치(Batch) 등 시스템에 사용
 - 3 HTML, 자바 스크립트 등의 웹 애플리케이션에 사용
 - 3 컴파일 언어와 스크립트 언어의 차이

컴파일 언어		스크립트 언어
일괄 번역, 일괄 수행	···수행 과정 ···	줄 단위로 번역 및 수행
일괄 수행을 위해 저장	목적 	불필요
빠름	····실행 시간 ···	느림
많이 필요		적게 필요

- ② 스크립트(Script) 언어의 이해
 - 4 스크립트 언어의 장점
 - 1 매번 컴파일 과정이 필요 없음
 - 간단한 알고리즘 및 테스트 등 간단한 코드 작성을 하기 좋음
 - 2 출 단위로 실행하기 때문에 많은 메모리를 필요로 하지 않음
 - 문법이 비교적 쉬워 초보자가 배우기 용이함
 - 프로그래밍을 처음 시작하는 초보자에게 적합한 언어

- ② 스크립트(Script) 언어의 이해
 - 5 스크립트 언어의 단점
 - 1 실행 시간이 오래 걸림
 - 고로그램을 실행할 때마다 번역을 새롭게 해야 함
 - 3 보안성이 컴파일 언어보다 떨어짐

- 1 파이썬(Python) 언어의 특징
 - 1 파이썬이란?

사전적 의미

독이 없는 뱀으로 먹이를 몸으로 감아서 압사시키는 큰 뱀, 또는 그리스 신화에 나오는 악마

- 1) 파이썬(Python) 언어의 특징
 - 1 파이썬이란?
 - 1 파이썬의 유래
 - 1970년대 영국 BBC에서 방영한 'Monty Python's Flying Circus'라는 프로그램에서 유래
 - 매우 어색하고 이상한 행동을 보이는 사람들을 쇼 형태로 방영하며 많은 인기를 누림
 - 파이썬을 만든 귀도 반 로섬(Guido van Rossum)이 이 프로그램의 엄청난 팬이었기에, 파이썬이 됨
 - 1989년 12월 크리스마스 휴가를 보내는 중이던 귀도 반 로섬(Guido van Rossum)은 연휴동안 무료함을 달래줄 무언가를 찾음
 - 마침 연휴라 연구실 또한 닫혀 있었고, 집에서 예전부터 생각했던 인터프리터를 만들어 보기로 결심함
 - → 무료한 한 때를 보내기 위해 만들기 시작했던 언어가 오늘날 많은 사람들이 사용하는 파이썬(Python)이 됨

- 1) 파이썬(Python) 언어의 특징
 - 1 파이썬이란?
 - 2 파이썬의 핵심 철학

80년대 초 ABC라는 교육용 언어를 기반으로 설계·발전한 언어

효율적

읽기 쉬움

간단한 코드 작성

"아름다운게 추한 것보다 낫다."

"명시적인 것이 암시적인 것보다 낫다."

"단순함이 복잡함보다 낫다."

"복잡함이 난해한 것보다 낫다."

"가독성은 중요하다."

- 1) 파이썬(Python) 언어의 특징
 - ② 파이썬 언어의 특징
 - 1

대화 기능의 인터프리터 언어

```
In [2]: print(1) #컴퓨터와 개발자가 대화하듯

1
In [3]: print(10) #한 줄씩 주고받으며 프로그래밍

10
In [4]: print(3)

3
```

- 1) 파이썬(Python) 언어의 특징
 - 2 파이썬 언어의 특징
 - 2 동적인 데이터 타입 결정 지원

```
a = "안녕"
b = "하세요"

print (a+b)

안녕하세요
```

```
a = [1,2,3]
b = [4,5,6]

print(a + b)

[1, 2, 3, 4, 5, 6]
```

```
a = 1
b = 2

print(a + b)
```

- 1) 파이썬(Python) 언어의 특징
 - 2 파이썬 언어의 특징
 - 3 플랫폼 독립적 언어
 - 똑같은 코드를 윈도우, 맥, 리눅스에서 별도의 과정 없이 완벽하게 실행 가능
 - 4 간단한 프로그램을 빠르게 작성하고 테스트할 수 있는 언어
 - 실행 결과를 바로 확인 할 수 있는 장점
 - 실행의 효율성보다는 개발 기간 단축에 보다 초점
 - 5 간단하고 쉬운 문법

```
a = [1, "문자", 1+4j, [12,3]]
print (a)
[1, '문자', (1+4j), [12, 3]]
```

- 6 들여쓰기 필수
- 7 고수준의 내장 객체 자료형 제공

- 2 파이썬(Python) 언어의 활용처 및 장·단점
 - 1 파이썬 언어의 활용처



- ② 파이썬(Python) 언어의 활용처 및 장·단점
 - 2 파이썬의 장점
 - 1 직관적이고 문법이 쉬워 배우기 쉬움
 - 2 들여쓰기를 철저하게 지키도록 설계
 - 코드의 가독성 및 유지보수 향상
 - **3** 메모리를 개발자가 신경 쓸 필요가 없음
 - 자동으로 관리

- 2 파이썬(Python) 언어의 활용처 및 장·단점
 - 2 파이썬의 장점
 - 4 많은 수의 라이브러리 제공
 - 특히, 4차 산업 혁명과 관련된 주요 라이브러리 제공
 - 5 짧은 코드로 작성 가능하고 유지보수가 쉬움
 - 일급함수, 다중 상속 등을 지원하는 객체지향 언어로 짧은 코드로 작성 가능
 - 다른 언어에 비해 깔끔하게 작성 가능(들여쓰기, 콜론, 동적 자료형 등)하므로 유지보수에 용이

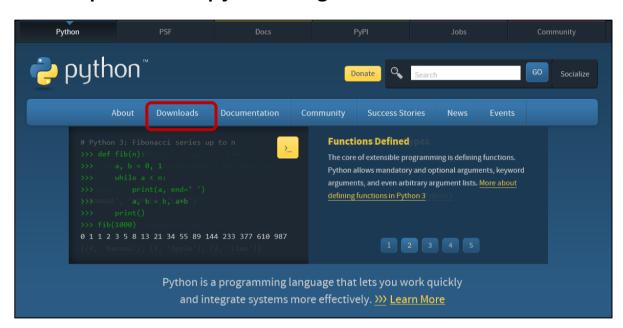
- ② 파이썬(Python) 언어의 활용처 및 장·단점
 - ③ 파이썬의 단점
 - 1 인터프리터 언어로 엔터프라이즈급 프로그래밍을 하기엔 속도가 느림
 - 고바일, 시스템 프로그래밍을 하기엔 적합하지 않음



- 1 파이썬 코딩 환경 구축하기
 - 1 파이썬 설치

파이썬 홈페이지에서 직접 파이썬 다운로드

https://www.python.org/



- 1 파이썬 코딩 환경 구축하기
 - 2 파이썬 통합 개발 환경



PyCharm



Jupyter Notebook



이클립스(Eclipse)



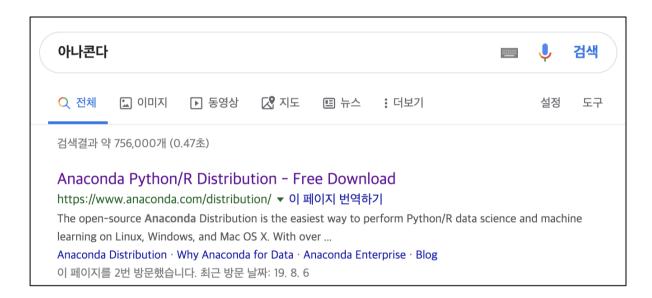
비주얼 스튜디오(Visual Studio)



텍스트 에디터(메모장, Sublime Text 등)

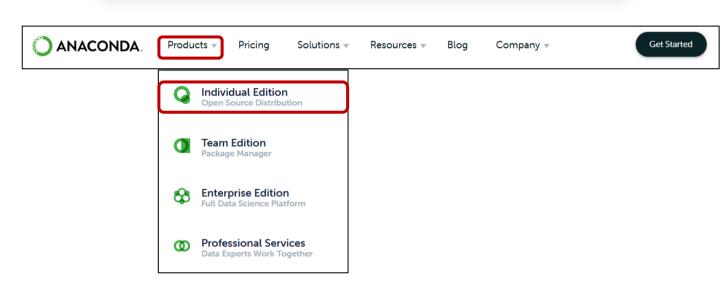
- 1 파이썬 코딩 환경 구축하기
 - ③ 파이썬 전용 아나콘다(Anaconda) 환경 사용
 - 수학, 과학 분야에서 필요한 중요한 라이브러리들(NumPy, Pandas…)을 함께 설치
 - 필요에 따라 환경을 여러 개 만들어 각자 다른 환경에서 프로그래밍할 수 있음
 - 최근 4차 산업혁명 관련해 IT분야에서 파이썬을 많이 활용
 - ➡ 아나콘다(Anaconda)가 거의 정석처럼 사용되고 있는 추세

- 1 파이썬 코딩 환경 구축하기
 - ③ 파이썬 전용 아나콘다(Anaconda) 환경 사용
 - 1 아나콘다 사이트 접속
 - https://www.anaconda.com/

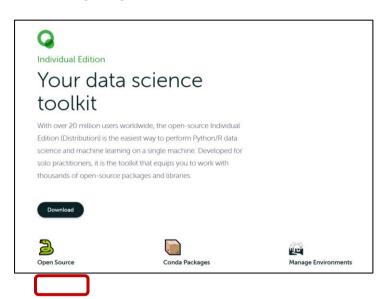


- 1 파이썬 코딩 환경 구축하기
 - ③ 파이썬 전용 아나콘다(Anaconda) 환경 사용
 - 2

우측 상단의 Products - Individual Editon 메뉴 클릭

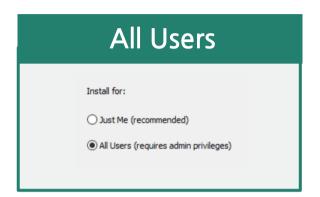


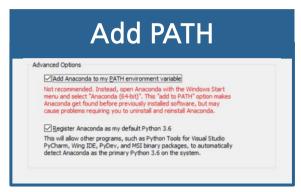
- 1 파이썬 코딩 환경 구축하기
 - ③ 파이썬 전용 아나콘다(Anaconda) 환경 사용
 - 3 다운로드 메뉴 클릭
 - 윈도우, 맥, 리눅스 버전 제공





- 1 파이썬 코딩 환경 구축하기
 - 4 추천 설치 옵션

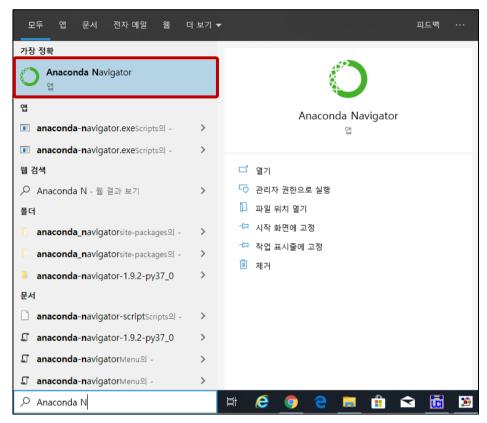


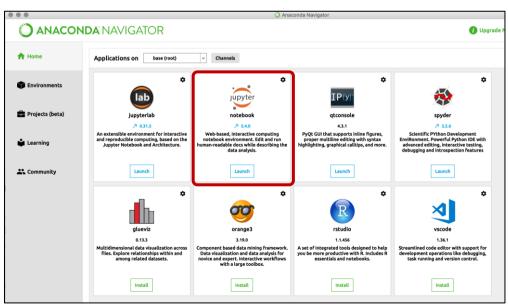




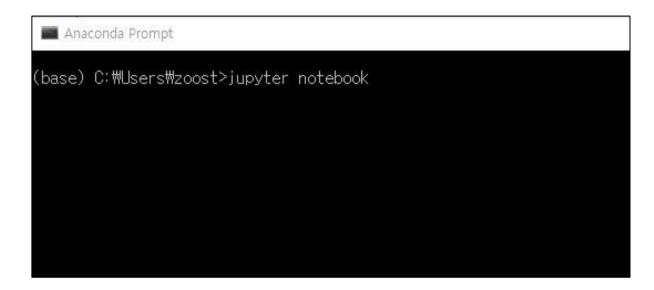
<mark>주의!</mark> 설치 경로에 한글이 있으면 제대로 설치되지 않을 수 있음

- ② 파이썬 실행하기(Jupyter Notebook)
 - 1) 아나콘다 네비게이터(Anaconda Navigator)

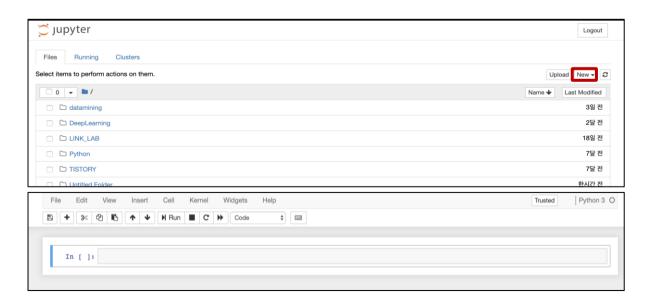




- 2 파이썬 실행하기(Jupyter Notebook)
 - 2 콘솔 창에서 실행



③ 우측 상단 New → Python 3 클릭



- 2 파이썬 실행하기(Jupyter Notebook)
 - 4 그 외 추천하는 코딩 환경 구성 방법



PyCharm 설치

- https://www.jetbrains.com/pycharm/ download/
- 무료 버전 다운로드 및 설치(Community 버전)



무료 코드 에디터 설치

- Sublime Text 3 설치
 - https://www.sublimetext.com/3
- 아톰, 브라켓, VS Code 등

Run! 프로그래밍



Mission 1 type() 함수를 활용하여 데이터 타입 확인

$$a = 1$$

$$c = 1.0$$

$$d = [1]$$

정답

print(type(a)), print(type(b))
print(type(c)), print(type(d))

Mission 2

Hello, World 출력

$$b = 'llo'$$

정답

print(a+b+c)

학습정리

1. 컴파일 언어와 스크립트 언어의 이해

컴파일 언어

- C언어, 자바(JAVA)와 같이 코드를 한번에 기계어로 번역하여 실행하는 언어
- 메모리가 많이 필요하지만 실행 시간이 빠른 장점이 있음

스크립트 언어

- 소스코드를 한 줄 한 줄 읽어 바로 실행하는 인터프리터 언어
- 플랫폼에 독립적이고 메모리가 적게 필요한 장점이 있음

2. 파이썬 언어의 이해와 특징

파이썬 언어의 이해

- 인공지능, IoT, 빅데이터 등 최신 IT 분야에 많이 활용되는 언어
- 대표적인 스크립트 언어 중 하나
- 컴파일 과정이 따로 없는 인터프리터 언어

파이썬 언어의 특징

- 대화형으로 한 줄씩 프로그래밍이 가능하기 때문에,
 메모리를 적게 차지하며 코드 작성이 간결하고 쉬운 장점이 있음
- 동적인 데이터 타입을 지워
- 리스트와 같은 고수준의 자료형을 제공함
- 플랫폼 독립적 언어로 어느 운영체제에서도 동일하게 수행됨
- 확장성이 높고 유지보수가 편리함

학습정리

3. 파이썬 코딩 환경 구축 및 실행하기

파이썬 코딩 환경 구축하기

- 홈페이지에서 직접 설치
- 아나콘다(Anaconda)라는 플랫폼을 설치해 필요한 라이브러리들을 함께 설치할 수 있음

파이썬 실행하기

- 단순히 텍스트 에디터에서 코드를 작성해 실행할 수 있음
- Jupyter Notebook을 활용
- 실시간으로 코드 결과를 확인해 보며 프로그래밍을 할 수 있음