



# 파이썬기초

1주차 - 3교시  
파이썬 개발환경 설치





## 학습내용

- 파이썬 통합개발 환경 설치
- 아나콘다(Anaconda) 환경에서 코딩해보기



## 학습목표

- 파이썬 통합개발 환경에 대해 설명할 수 있고 개발 환경을 구축할 수 있다.
- 파이썬 프로그래밍 코딩 규칙에 맞게 코드를 작성할 수 있다.

01



# 파이썬 통합개발 환경 설치



## 1 | 파이썬 통합 개발 환경

**PyCharm**

**Jupyter Notebook**

**이클립스(Eclipse)**

**비주얼 스튜디오(Visual Studio)**

**텍스트 에디터(메모장, Sublime Text 등)**

## 2 | 파이썬 전용 아나콘다(Anaconda) 환경

- 수학, 과학 분야에서 필요한 중요한 라이브러리들이 함께 설치되어 있음

“ **아나콘다(Anaconda)환경이  
많이 사용되고 있는 추세** ”

## 3 | 파이썬 전용 아나콘다(Anaconda) 환경 구축

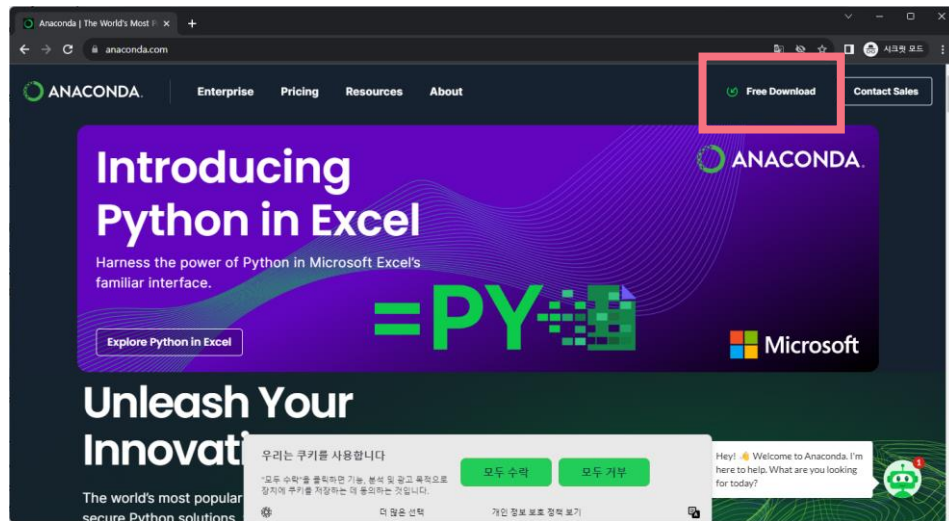
### 01 아나콘다(Anaconda) 설치

- 파이썬과 파이썬의 패키지 관리, 라이브러리 관리,  
다양한 수학, 과학 계산 등을 쉽고 편리하게 할 수 있는  
**파이썬 배포판**

## 3 | 파이썬 전용 아나콘다(Anaconda) 환경 구축

### 01 아나콘다(Anaconda) 설치

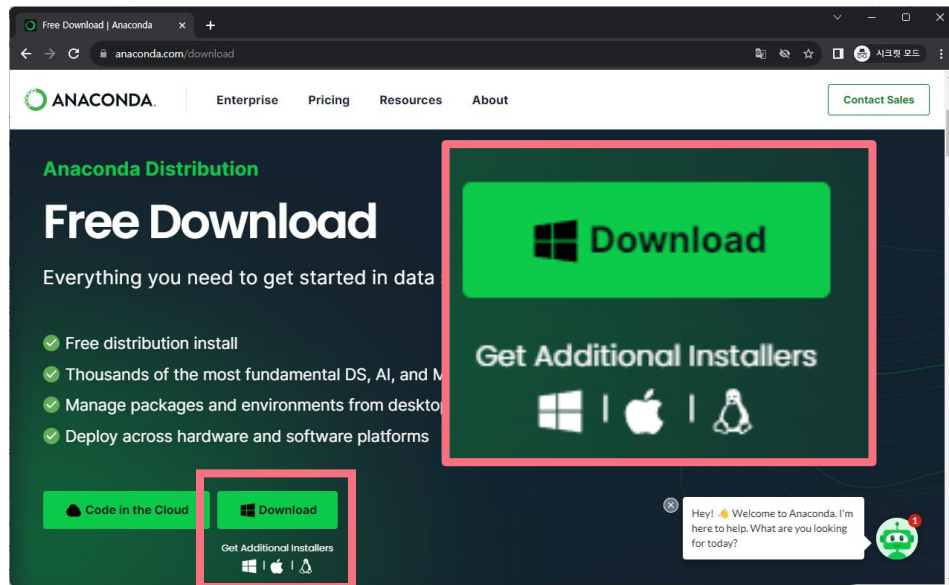
> <https://www.anaconda.com>



## 3 | 파이썬 전용 아나콘다(Anaconda) 환경 구축

### 01 아나콘다(Anaconda) 설치

➤ Download 클릭





## 3 | 파이썬 전용 아나콘다(Anaconda) 환경 구축

### 01 아나콘다(Anaconda) 설치

#### > 설치 시 주의사항

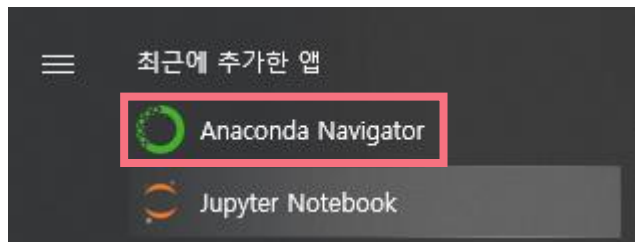
Windows PC의 경우, 설치 경로에 **한글이 없어야 함**

**예** C:\Users\WW\파이썬\WWDocuments → X  
C:\Users\WW\python\WWDocuments → O

## 3 | 파이썬 전용 아나콘다(Anaconda) 환경 구축

### 01 아나콘다(Anaconda) 설치

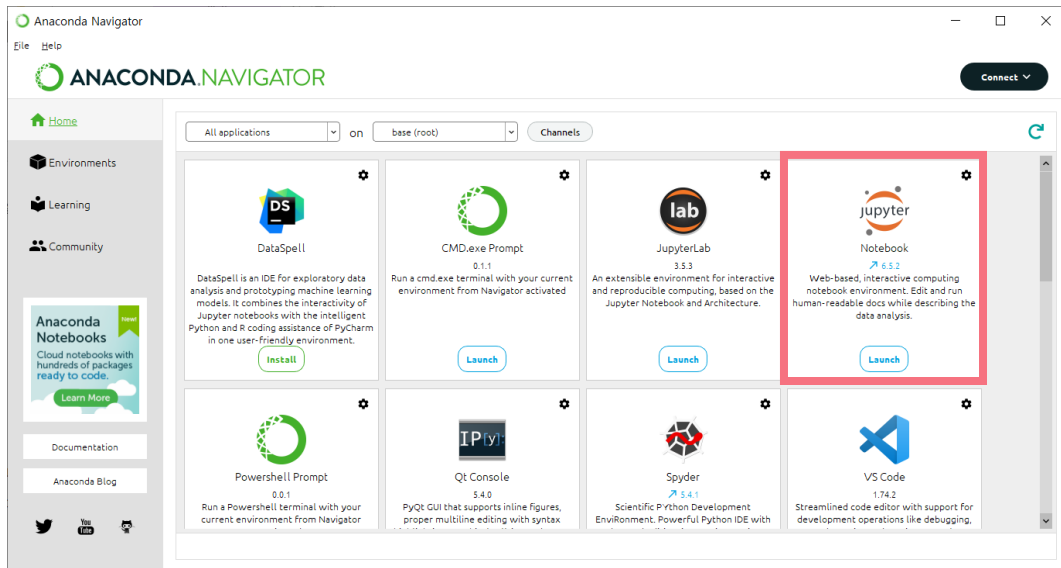
#### > Anaconda Navigator 실행



## 3 | 파이썬 전용 아나콘다(Anaconda) 환경 구축

### 01 아나콘다(Anaconda) 설치

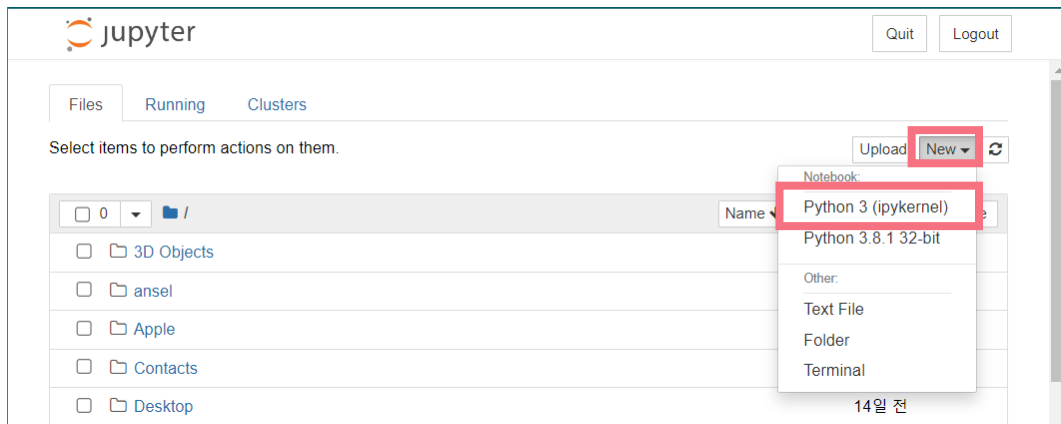
#### > Jupyter Notebook 실행



## 3 | 파이썬 전용 아나콘다(Anaconda) 환경 구축

### 01 아나콘다(Anaconda) 설치

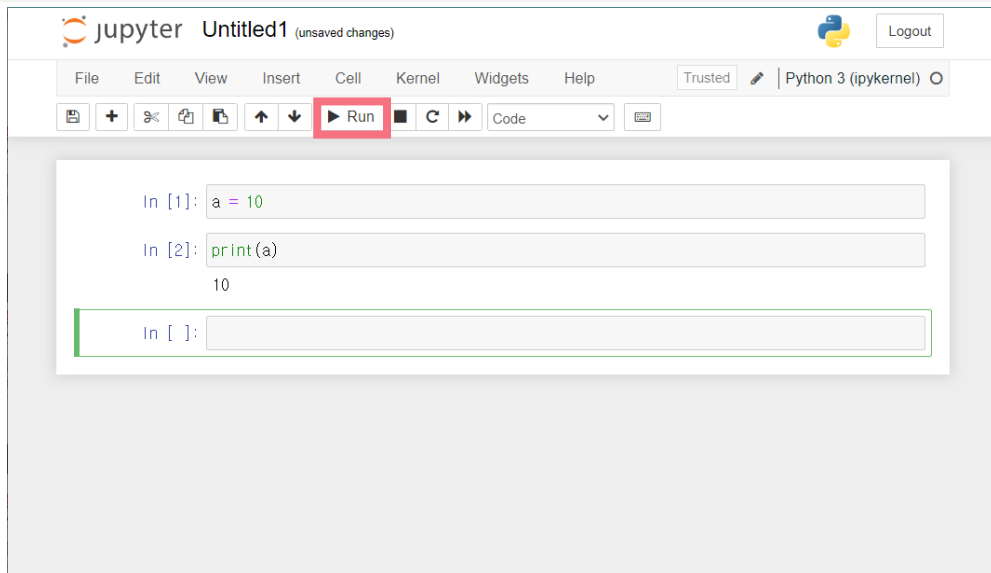
➤ Jupyter Notebook 우측 상단 New  
→ Python3 클릭



## 3 | 파이썬 전용 아나콘다(Anaconda) 환경 구축

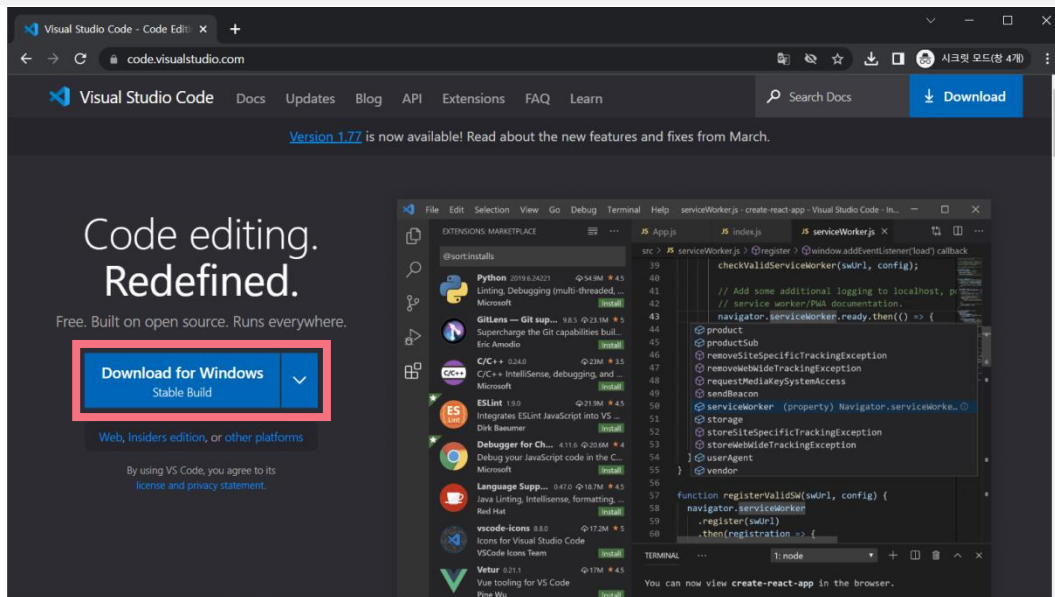
### 01 아나콘다(Anaconda) 설치

➤ 코드 입력 후 Run 버튼을 사용하여 코드 실행



## 4 | 비주얼 스튜디오(Visual Studio) 설치

➤ <https://code.visualstudio.com>  
[Downloads for Windows] 클릭하여 설치 파일  
다운로드



## 5 | PyCharm 설치

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



Version: 2021.2.2  
Build: 212.5284.44  
15 September 2021

[System requirements](#)  
[Installation Instructions](#)  
[Other versions](#)  
[Third-party software](#)

### Download PyCharm

[Windows](#) [macOS](#) [Linux](#)

#### Professional

For both Scientific and Web Python development. With HTML, JS, and SQL support.

Download

Free trial

#### Community

For pure Python development

Download

Free, built on open-source



Get the Toolbox App to download PyCharm and its future updates with ease

02



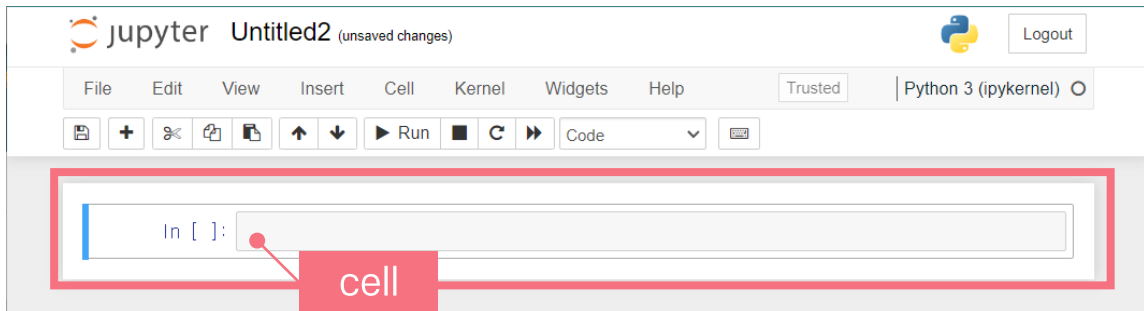
# 아나콘다환경에서 코딩해보기





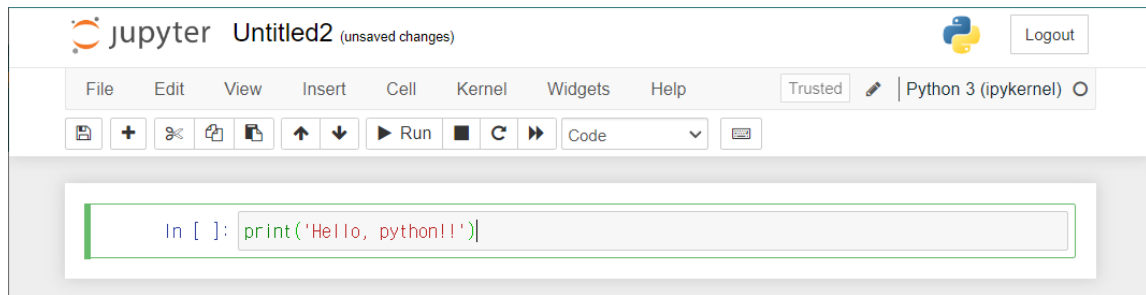
### 1 | 새로운 Notebook 생성

> cell 안에 코드를 작성하고 실행



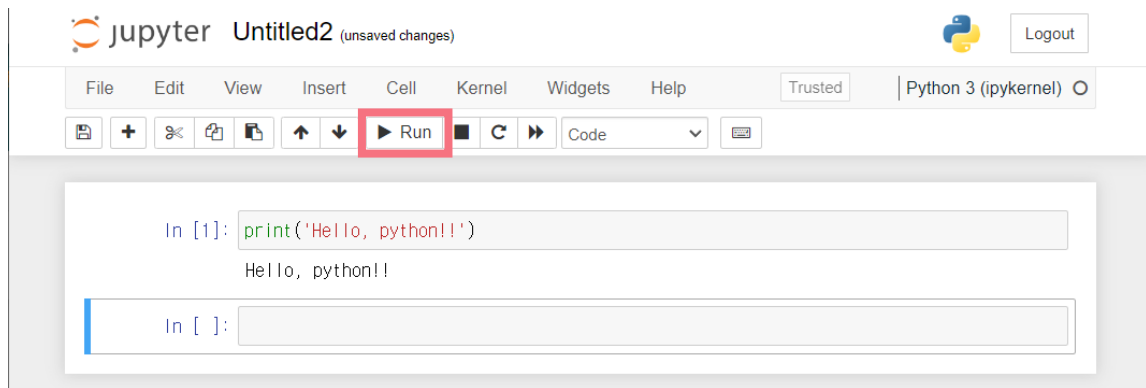
### 2 | 코드 작성 및 실행

> In [ ]: 오른쪽에 `print('Hello, python!!')`를 입력



### 2 | 코드 작성 및 실행

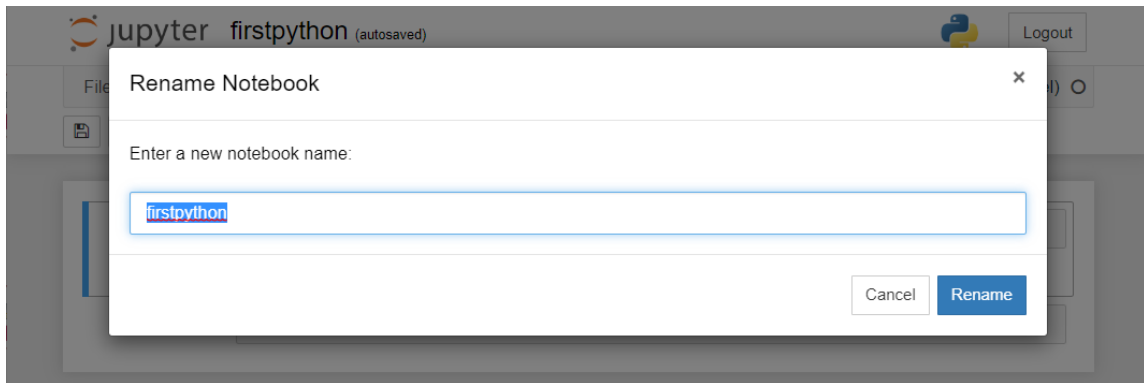
> Run 버튼을 클릭해서 실행



- 실행 결과가 cell 아래 출력되고 새로운 Cell 이 생성됨
- **In [1 ]:** → 첫 번째 실행된 코드임을 의미, 코드가 실행 되면서 숫자는 자동으로 증가되어 출력됨

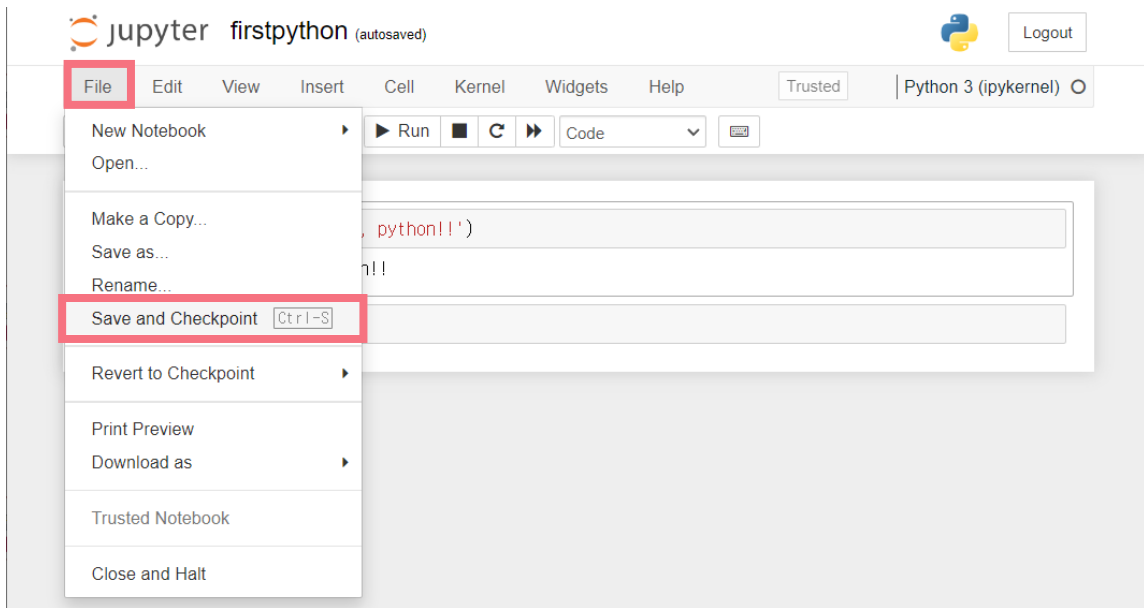
### 3 | notebook 제목변경

> Jupyter 로고 옆의 Untitled를 클릭



### 3 | notebook 제목변경

> 메뉴의 File > Save and Checkpoint를 클릭해서 저장



### 4 | 파이썬 출력문 작성하기



`print(인자)`

- 괄호 안에 작성한 **인자**를 출력함
- 문자열, 숫자, 연산결과 등을 출력

### 4 | 파이썬 출력문 작성하기

#### 01 코드 작성해 보기

> print("파이썬 코드를 작성합니다.")  
print("파이썬을 이용해서 코딩을 합니다.")  
print("프로그램을 만들기위해 파이썬 언어를 사용합니다.")

```
In [7]: print("파이썬 코드를 작성합니다.")  
        print("파이썬을 이용해서 코딩을 합니다.")  
        print("프로그램을 만들기위해 파이썬 언어를 사용합니다.")
```

파이썬 코드를 작성합니다.  
파이썬을 이용해서 코딩을 합니다.  
프로그램을 만들기위해 파이썬 언어를 사용합니다.

### 5 | 주석 이용하기

#### 01 주석이란?

##### 주석

프로그램 **코드의 설명**이나 프로그래머 **개인적인 메모**와 같은 것으로 프로그램 코드로 인식하지 않는 문자열

→ 파이썬에서는 # 을 쓰고 그 뒤에 주석을 작성



### 5 | 주석 이용하기

#### 02 코드 작성해 보기

> # 이름을 출력하는 코드입니다.  
print("손흥민")

```
In [8]: # 이름을 출력하는 코드입니다.  
print("손흥민")
```

손흥민

### 5 | 주석 이용하기

#### 03 주석을 잘못 사용한 경우

> 이름을 출력하는 코드입니다.  
`print("손흥민")`

```
In [9]: 이름을 출력하는 코드입니다.  
        print("손흥민")
```

```
Cell In[9], line 1  
    이름을 출력하는 코드입니다.  
    ^
```

**SyntaxError:** invalid syntax



# Q1

고급언어를 기계어로 변경해주는 번역기 시스템 중  
한 줄씩 해석해서 실행하는 시스템을 무엇인가?

- ① 컴파일러
- ② 어셈블러
- ③ 인터프리터
- ④ 운영체제



## Q1

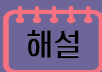
고급언어를 기계어로 변경해주는 번역기 시스템 중 한 줄씩 해석해서 실행하는 시스템을 무엇인가?

- ① 컴파일러
- ② 어셈블러
- ③ 인터프리터
- ④ 운영체제



정답

3번



해설

프로그램이 실행하면서 한 줄씩 해석하면서 실행하는 방식의 번역기는 인터프리터입니다. 반대로 한번에 모든 코드를 기계어로 변경해주는 번역기를 컴파일러라고 합니다.



# Q2

변수의 타입이 컴파일 시에 결정되는 것이 아닌 실행 시점에  
변수의 타입이 결정되는 언어를 무엇이라 하는지 쓰시오.

- ① 동적타입언어
- ② 정적타입언어
- ③ 객체지향언어
- ④ 절차지향언어



## Q2

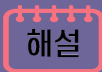
변수의 타입이 컴파일 시에 결정되는 것이 아닌 실행 시점에 변수의 타입이 결정되는 언어를 무엇이라 하는지 쓰시오.

- 1 동적타입언어
- 2 정적타입언어
- 3 객체지향언어
- 4 절차지향언어



정답

1번



해설

변수의 타입이 컴파일 시에 결정되는 것이 아닌 실행 시점에 변수의 타입이 결정되는 언어를 동적타입언어라고 하고 컴파일 시에 결정되는 언어를 정적타입언어라고 합니다.



# Q3

파이썬 언어의 특징이 아닌 것을 고르시오.

- ① 문법이 쉽고 간결함
- ② 플랫폼에 종속적인 언어
- ③ 다양한 분야의 라이브러리를 지원
- ④ 다른 언어에 비해 상대적으로 느림



## Q3

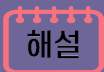
파이썬 언어의 특징이 아닌 것을 고르시오.

- ① 문법이 쉽고 간결함
- ② 플랫폼에 종속적인 언어
- ③ 다양한 분야의 라이브러리를 지원
- ④ 다른 언어에 비해 상대적으로 느림



정답

2번



해설

파이썬 언어는 플랫폼에 종속적이지 않은 플랫폼에 독립적인 언어입니다.





## 프로그래밍언어

- 프로그래머가 **프로그래밍언어**를 이용해서 컴퓨터 **프로그램**을 작성하는 것을 **프로그래밍**이라 함
- **고급언어**를 **기계어**로 변경하는 **언어 번역기**에 따라 **컴파일 언어**와 **인터프리터 언어**로 구분할 수 있음
- 프로그래밍 언어 특성에 따라 **객체지향 언어**와 **절차지향언어**, **정적타입언어**와 **동적타입언어**로 구분할 수 있음



## 파이썬

- 파이썬은 인터프리터 방식으로 실행되는 언어로 **객체지향**과 **동적타이핑**을 지원하는 언어
- 초보자도 쉽게 배울 수 있는 언어이고 **다양한 라이브러리**를 제공하고 있어 다양한 분야의 프로그램을 만드는데 사용됨
- 파이썬의 최신 버전은 2.x 와 3.x 버전을 제공하고 있고 최근 제공되는 라이브러리들이 3.x 버전에 잘 호환되고 있어 최근 버전을 활용하여 개발함



## 파이썬

- 파이썬 프로그래밍을 위한 통합개발환경은 **Pycharm, VSCode, Jupyter Notebook** 등이 있어 환경에 맞게 선택해서 사용 가능함



## 파이썬 전용 아나콘다(Anaconda) 환경

- 파이썬과 파이썬의 패키지 관리, 라이브러리 관리, 다양한 수학, 과학 계산 등을 쉽고 편리하게 할 수 있는 파이썬 배포판으로 수학, 과학 분야에서 필요한 중요한 라이브러리들이 함께 설치되어 있음



# 파이썬기초

NEXT - 2주차 1교시  
변수의 이해

