|  |
| --- |
|  |
| **4시간만에 끝내는 파이썬 기초** |
|  |

목차

[1 개요 3](#_Toc515957861)

[1.1 프로젝트 주제 3](#_Toc515957862)

[1.2 프로젝트 추진 배경 목표 3](#_Toc515957863)

[2 왜 파이썬인가? 4](#_Toc515957864)

[3 파이썬의 문법 5](#_Toc515957865)

[3.1 입력과 출력 5](#_Toc515957866)

[3.2 변수와 변수명 6](#_Toc515957867)

# 개요

## 프로젝트 주제

|  |
| --- |
| 개인 역량 강화 프로젝트 |

표 1- 1 프로젝트 주제

## 프로젝트 추진 배경 목표

|  |
| --- |
| 시큐어가드테크놀러지 입사 전 기초적인 python 문법을 익혀 역량을 강화해보자. |

표 1- 2 프로젝트 추진 배경 및 목표

# 왜 파이썬인가?

|  |
| --- |
| 문법을 쉽게 배우고 사용할 수 있고 가독성이 뛰어남. |
| 파이썬은 다른 언어로 만든 프로그램을 파이썬 프로그램에 포함시킬 수 있음. |
| 개발 속도가 빠름. |
| 교차 플랫폼 언어이기에 윈도우, 리눅스 등 다른 플랫폼에서도 동일하게 구동 가능. |
| 무료, 오픈소스이기에 자유롭게 파이썬을 이용 가능 |
| 파이썬으로 GUI(Graphical User Interface)를 개발 가능. |

표 2- 1 파이썬의 장점

|  |
| --- |
| 인터프리터 언어이기에 JAVA나 C++같은 컴파일 언어보다 느림. |
| 동시다발적 멀티스레드를 처리하거나 CPU에 집중된 많은 스레드를 처리하는 데 부적합. |
| 2.X 버전과3.X 버전이 호환되지 않음. |

표 2- 2 파이썬의 단점

# 파이썬의 문법

## 입력과 출력

**print() :** 입력한 자료를 출력한다.



그림 3- 1 print 사용의 간단한 예

print(‘python’ ‘is’ ‘funny’)와 print(‘python’+‘is’+‘funny’)의 결과값은 같게 나온다.



그림 3- 2 print에서 +와 따옴표 연산의 동일함을 보여주는 예

콤마(,)는 문자열의 띄어쓰기 역할을 한다. print(‘python’, ‘is’, ‘funny’)의 결과는 이렇다.



그림 3- 3 print에서 콤마(,)의 역할

한 줄에 결과값을 출력할 수도 있다. for문을 이용해 예제를 만들어보았다.



그림 3- 4 print에서 인수 end를 이용해 한 줄로 연결

(위 3-4 코드는 파이썬 2.7 버전의 경우 print(i, end=’ ‘)가 아닌 print i, 를사용한다.)

**input() :** 사용자에게 입력을 받는다.



그림 3- 5 input() 사용의 간단한 예

사용자에게 입력을 받을 때 안내문구 또는 질문을 출력하고 싶다면 input() 괄호 안에 내용을 입력해주면 내용을 출력한 후 입력을 받는다.



그림 3- 6 내용 출력 후 input()으로 5 입력

## 변수와 변수명

**변수 :** 객체를 가리키는 것이라고도 말할 수 있다. 객체란 자료형과 같은 것을 의미한다. 변수의 상징적인 이름을 데이터의 실제 위치로 치환해야 한다. 변수 값, 형, 위치는 일반적으로 고정된 채 유지되는 반면 위치에 저장되어 있는 데이터는 프로그램 실행 도중 변경될 수 있다

|  |
| --- |
| 영문자(대문자, 소문자), 숫자, 언더바(\_)를 사용할 수 있다. |
| 첫 자리에는 숫자를 사용할 수 없다. |
| 파이썬 키워드는 변수명으로 사용할 수 없다. |

표 3- 1 변수명의 규칙

파이썬에서 변수를 선언할 때 C, Java 등과 다르게 변수의 자료형과 함께 쓸 필요가 없다. 파이썬은 변수에 저장된 값을 스스로 판단하여 자료형을 알아낸다.

변수가 가리키는 메모리 주소를 id()함수로 알 수 있다. a 변수의 메모리 주소를 알아보자.



그림 3- 7 변수 a의 메모리 주소 출력