



DAS 사전교육

- Python 기초 -



Jupyter Notebook



- ▶ 실행방법
- ▶ 노트북 만들기
- ▶ 주석 마크다운 ➔ 참고: <https://www.markdownguide.org/cheat-sheet#downloads>
- ▶ 단축키
- ▶ 테마 변경 `pip install jupyterthemes`
 - ▶ `jt -l` #가능한 테마 리스트
 - ▶ `jt -t <테마 이름> -T # -T 를 안붙이면 툴바가 보이지 않습니다.`
 - ▶ `jt- r #default css로 복귀`
- ▶ 주피터 확장탭 `pip install jupyter_contrib_nbextensions && jupyter contrib nbextension install --user`
 - ▶ hinterland 자동완성
 - ▶ Snippets 코드예시
 - ▶ Table of contents 목차만들기
 - ▶ Execute Time 코드실행시간확인
 - ▶ 참고: <https://jupyter-contrib-nbextensions.readthedocs.io/en/latest/install.html>



기본 개념1



- ▶ 1. 변수
- ▶ 2. 연산자
- ▶ 3. 표준 입출력
- ▶ 4. 선택문
- ▶ 5. 반복문과 기타 제어문
- ▶ 6. 함수



1. 변수



- ▶ 변수 variable : 쉽게 변하는 수
- ▶ 특징
 - ▶ 일시적으로 자료를 저장하는 공간
 - ▶ 변수에 저장된 값은 변할 수 있다
 - ▶ 숫자, 문자열 등 모든 자료형을 저장할 수 있다
 - ▶ 다른 변수의 값도 저장할 수 있다
 - ▶ 사용되기 전에 반드시 할당되어 있어야 한다



1. 변수



▶ 변수를 사용하는 이유

- ▶ 프로그램에서 데이터를 변수에 저장해놓으면 필요할 때마다 꺼내 사용할 수 있다

▶ 변수 생성 및 규칙

- ▶ 규칙1: 영문자, 숫자, 밑줄 문자(_)로 구성

첫 글자는 반드시 영문자/밑줄 문자(_)로 시작

밑줄 문자(_) 제외한 특수문자는 사용할 수 없다

- ▶ 규칙2: 공백이 들어가면 안된다
- ▶ 규칙3: 대문자와 소문자를 구별한다
- ▶ 규칙4: 예약어는 변수명으로 사용할 수 없다



1. 변수



▶ 좋은 변수명 원칙

- ▶ 짧게 줄이는 것보다는 의미 전달이 잘 되게 하는 것이 좋다
- ▶ 변수명은 간단명료한 것이 좋다
- ▶ 변수명은 전체 프로그램에서 하나의 일관된 규칙으로 작성하는 것이 좋다

주로 소문자와 밑줄 문자가 섞여있는 형식을 사용



1. 변수



▶ 다양한 타입의 변수

자료형	의미	예시
int	integer, 정수	n = 100
float	float, 부동소수점	n = 95.5
str	string, 문자열	n = 'Kim' n = "Kim"
bool	Boolean, 불	n = True n = False



2. 연산자



▶ 산술 연산자

산술 연산자	기능
+	더하기
-	빼기
*	곱하기
/	나누기
**	제곱
//	나눗셈의 몫
%	나눗셈의 나머지



2. 연산자



▶ 대입(할당, assignment) 연산자

- ▶ 변수에 값을 할당하기 위해 사용
- ▶ 기본적으로 등호(=) 사용, 산술 연산자와 함께 사용하면 간결하게 할당 가능

대입 연산자	기능
=	왼쪽 변수에 오른쪽 값을 할당
+=	왼쪽 변수에 오른쪽 값을 더하고 결과를 왼쪽 변수에 할당
-=	왼쪽 변수에 오른쪽 값을 빼고 결과를 왼쪽 변수에 할당
*=	왼쪽 변수에 오른쪽 값을 곱하고 결과를 왼쪽 변수에 할당
/=	왼쪽 변수에 오른쪽 값을 나눈 몫의 결과를 왼쪽 변수에 할당
%=	왼쪽 변수에 오른쪽 값을 나눈 나머지의 결과를 왼쪽 변수에 할당
**=	왼쪽 변수에 오른쪽 값만큼 제곱을 하고 결과를 왼쪽 변수에 할당
//=	왼쪽 변수에 오른쪽 값을 나눈 몫의 결과를 왼쪽 변수에 할당



2. 연산자



▶ 비교 연산자

비교 연산자	기능
<	작다
<=	작거나 같다
>	크다
>=	크거나 같다
==	같다
!=	같지 않다



2. 연산자



▶ 논리 연산자

논리 연산자	기능
and	둘 다 참일 때만 참
or	둘 중 하나만 참이여도 참
not	논리 상태를 반전



2. 연산자



- ▶ 문자열 연산자
 - ▶ 문자열 더하기 ➔ 연결
 - ▶ 문자열 곱하기 ➔ 반복



3. 표준 입출력



- ▶ 표준 입력 → `input()` 함수
 - ▶ 함수 내에 안내할 문자열을 포함시켜 사용
- ▶ 표준 출력 → `print()` 함수
 - ▶ 콤마(,)를 이용한 출력
 - ▶ % 형식지정자를 이용한 출력
 - ▶ `format()` 함수를 이용한 출력
 - ▶ f-string 포매팅을 이용한 출력