[조요한 교수님]

퀴즈 20분 - 10분전에 열림

리뷰 20분

실습 1시간 20분

파이썬은 인터프리터

C언어는 컴파일러

수행방법 python file\_name.py

Interactie shell : 한 줄마다 해석해줌

Jupyter notebook : kernel

Expression:표현식

Operands : 피연산자

Operators : 연산자

Evaluataed : 계산되는 것

2\*\*3 = 8

7//2 = 3 : 정수 부분

5%2 = 1 : 나머지 부분

4==5

Python 가능한 변수명

True\_

\_\_my\_score\_\_

\_

totalPrice2

x,y,z = 5, 10, 15

x, y = y, x+z > x=10, y=5+15=20, z= 15 : 변수 x를 업데이트한 후 뒤를 계산하는게 아니라 y, x+z가 먼저 계산된 후에 할당됨

1. assignment statement(=) 오른쪽에 있는 표현식을 계산한다 : 계산, 함수(할당)

2. 계산값을 담고있는 object를 메모리에 올려놓음

3. python의 할당 : 메모리(5라는 object)의 주소를 변수에게 할당(reference)

4. 만약 이미 x가 이미 존재한다면, 메모리의 다른 주소(5라는 object)를 reference 하고 있는 상황 3이라는 메모리의 다른 주소로 reference하도록 할당이 일어남

Function call

1. frame이라는 데이터 structure을 만들어 냄(로컬 변수, 인자, return 주소)
2. 전역 scope에는 add, var1, var2
3. add라는 frame에는 로컬변수, 인자
4. namespace(dictionary) 다양한 변수들의 이름과 -를 매핑해놓은 테이블, 각각의 이름과 무엇을 가리키는지 매핑
5. 함수 호출되면 메모리가 할당됨. 변수들에게 할당된 값들은 object의 형태로 메모리에 올라옴. 이러한 메모리 공간이 필요해서 메모리가 할당됨 num1은 5를 objec를 reference한

Python string – conversion specifier

1. %s : 스트링 형태로 출력 %name이라는 변수를 스트링 형태로
2. %d : 변수의 값을 정수로 바꾸어 출력
3. %f : floating형태로 기본적으로 6자까지 출력(반올림)
4. %.2f : 소수점 뒤에 두 개까지 출력
5. %x : 16진수로 바꾸어 출력
6. %c : 숫자에 해당하는 ASCII코드에 해당하는 값을 출력 A=65
7. %% : %를 출력하고 싶을 때

데이터 사이언스 – ABC

A : AI모델 알고리즘

B: 빅데이터

C: 컴퓨팅

+ D : 도메인, 애플리케이션(데이터의 소스)

Drawing useful conclusion : 애플리케이션에 유용

컴퓨터 사이언스

: 컴퓨터를 효율적으로 사용할 줄 알아야 함

알고리즘과 프로그램

: 컴퓨터가 따라갈 수 있는 논리적인 스텝(알고리즘)

: 프로그램(instruction set) – 컴퓨터가 이해할 수 있는 언어로

1. Abstraction : 관련 있는 정보(분야)만 알면 됨

How does computer run a Python program?

어떻게 하드웨어가 알아듣고 그거에 맞춰 반응해서 돌아가는지?

Ex) 차 작동) Car : 핸들, 엑셀 : interface

차의 종류, 내연 기관 : implementation

Abstraction – Machine Learning

Pytorch, tensorflow, : abstraction이 잘 됨

MATLAB : 잘 안되어 있음

Abstraction – Mobile Phone

하드웨어 – operating system(운영체제) : android – application : 인스타

Abstraction – More General Computer

Application – 로스트 아크, 줌, 등

운영 체제 – 윈도우-----------------------------

Hardware – 와이파이, 블루투스, 유선, cpu, 스피커

: 마우스를 바꿨다고 ETL을 못하지 않음

Abstraction – Thank you operating system!

-하드웨어에 직접 접근하는 특수한 프로그램

- application이 직접 접근할 수 없고 운영체제를 통해 요청을 보낼 수 있음

Abstraction – Thank you Interpreter

Application – python interpreter : 바로 운영체제로 가는게 아니라

Operating system

하드웨어

파이썬 – 파이썬 인터프리터 – 운영체제

파이썬 인터프리터

.py extension

실행할 때 인터프리터가 실행

Each statement로 확인하고 싶을 때 shell을 활용 (매 줄마다 인터프리터가 실행됨)

Jupyter notebook : 여러 줄, 한 줄 상관없이 그룹으로 실행할 수 있음

Summary

Abstaction – interface and implementation

Interface 위주

결국 점점 low-level system programming!

하드 웨어를 감춰주는 Operating system and 이를 또 활용하는 application

Python intepreter