교육 결과 보고서

[Online] Python Programming

1. Python 특징

1.1 가독성

간결하고 가독성이 좋음

코드 블록을 들여쓰기로 구분

1.2 풍부한 라이브러리

매우 광범위한 라이브러리가 기본으로 포함되어 있음.

외부 라이브러리가 풍부하여 확장이 쉬움.

1.3 접착성

쉽게 라이브러리르 추가 할 수 있음.

파이썬에서 C 로 구분된 부분을 사용할 수도 있으며, C 에서 파이썬을 사용할 수 도 있음.

1.4 무료 소프트웨어

파이썬 소프트웨어 재단에서 무료와 다름없는 라이선스 적용 (Python Software Foundation License)

1.5 유니코드

파이썬에서의 문자열은 모두 유니코드로 한글, 한자 등을 표현하기 위한 특별한 노력이 필요 없음.

1.6 동적 타이핑

런타임 시에 타입을 체크 하는 동적 타이핑을 지원

메모리 관리 자동으로 함

2. 자료 형 및 연산자

2.1 수치 및 연산자

수치 : int, float, complex

연산자 : +, -, \*, /, //, %, \*\*, =

2.2 문자

\* 표현 : ‘string’, “string”,

“””줄바꿈

적용 문자열”””

\* +, \* 연산자 사용 가능 (py + thon => python, py \* 3 => pypypy)

\* 인덱싱 & 슬라이싱 가능 (python[0] => p, python[1:4] => yth)

\* 기본적으로 유니코드, 유니코드 이외의 인코딩이 있는 문자열은 bytes 로 표헌.

2.3 리스트

\* 갑들의 나열, 인덱싱, 슬라이싱 가능

colors = [‘red’, ‘green’, ‘gold’]

2.4 튜플

\* 리스트와 유사

\* 읽기 전용

2.5 딕셔너리

\* 키와 값의 쌍으로 이루어져 있음

\* 새로운 값은 새로운 키와 값 할당으로 추가.

3. 함수

3.1 선언

def <함수명>(인수1, 인수2, …. 인수N):

<구문>

return <반환값>

4. 생성자 와 소멸자

클래스와 인스턴스 개념 이용

4.1 생성자

생성 시 초기화 작업을 수행

인스턴스 객체가 생성될 때 자동으로 호출

\_\_init\_\_()

4.2 소멸자

소멸 시 종료 작업 수행.

인스턴스 객체의 참조 카운터가 0 이 될 때 호출

\_\_del\_\_()

5. 모듈 임포트 방법

\* from <모듈> import <어트리뷰트> : from simpleset import union => union()

\* from <모듈> import \*

\* import <모듈> as <별칭> : <모듈> 이름을 <별칭> 으로 변경하여 임포트

6. 예외 처리

try:

< 예외 발생 가능성 있는 문장>

except <예외 종류>:

< 예외 처리 문장 >

except <예외1, 예외2>:

< 예외 처리 문장 >

else:

< 예외가 발생하지 않은 경우 수행할 문장 >

finally:

< 예외 발생 유무에 관계 없이 try 블록 이후 수행할 문장 >