----------------------------------------파이썬-------------------------------------------------------------

- Input 함수는 문자열 타입으로 리턴, 그렇기 때문에 숫자를 받고 싶다면 int()를 사용해서 형변환

- 타입을 알 수 있는 type()함수

- 선언되지 않은 변수 또는 삭제된 변수 읽으면 예외 발생 (name 'a' is not defined)

- 16진수(0x, hex()), 8진수(0o, oct()), 2진수(0b, bin())

- \n, \t, \”, \’, \\ 문자 이스케이프

- = (대입연산자), + - \* / \*\* // % (산술연산자), += (복합 대입 연산자)

- 문자열 연결 ( +(조심! 문자열만 +로 연결 가능, 다른 타입 연결하려면 str()변환 필요), \*)

- invalid literal for int() with base 10 -> int로 형변환할 때 주로 사용, 10진수가 아님, float() 대체

- float() -> 실수 변환, round(숫자, [반올림 자리수]) -> 실수 반올림 함수

- bool() -> 부울 변환 함수, list() -> 리스트 변환 함수, tuple -> 튜플 변환 함수, dict() -> 사전 변환 함수

- if 조건: , else: , elif 조건:, while 조건:, for 제어변수 in 컬렉션:, break, continue

- for문 range(시작, 끝 + 1, 증가값) (즉 마지막 값에 쓰는 건 포함되지 않음)

- False, None, 0, “”, [], () -> 거짓값

- 논리연산자 and, or, not

- def 함수이름(인수목록):

- 함수명(인수목록) -> 함수호출

- def 함수명(\*인수명): -> \*인수명 -> 가변인수, 튜플로 받음, 일반 인수 뒤에만 올 수 있음, 하나만 사용 가능, 인수의 수가 고정되지 않음

- def 함수명(인수명 = 인수의 기본값) -> 인수명 = 인수의 기본값 ->인수의 기본값, 함수 호출시 인수가 지정되지 않았을 때 사용할 값, 함수 정의시 인수에 값을 대입, 인수 목록의 마지막 부분에 배정, 중간에 배정시 구분 불가

- 일반적으로 함수 호출시 인수 배치 순서대로 매치하지만 키워드 인수는 인수의 명칭으로 매칭하는 방법이다

- def 함수명(\*\*인수명): -> \*\*인수명 -> 키워드 가변 인수, 키워드 인수를 가변 개수로 전달할 때 사용하는 방법, 사전으로 받는다

- 일반 변수, 가변 변수, 키워드 가변 변수 모두 사용하려면 일반 변수, 가변 변수, 키워드 가변 변수 순서로 배치, 즉 def 함수명(일반변수, \*가변변수, \*\*키워드가변변수):

- 지역 변수, 함수 내에서 사용된 변수, 함수 내에서만 사용 가능, 함수 밖에서 사용 불가, 지역변수를 밖에서 사용하려고 하면 뜨는 오류(name 'temp' is not defined)

- 지역 변수, 이름 같아도 다 다른 변수이다

- 전역 변수, 어디서든 접근 가능한 탑 레벨에서 사용된 변수

- 함수 안에서 전역 변수 쓰려면, global 붙이기

- “”” “”” -> docstring help(함수명) 호출시 출력될 문자열

- 문자열[정수] -> 0에서부터 인덱싱(왼쪽부터), 문자열[-정수] -> 끝에서부터 인덱싱

- 슬라이싱, 문자열[시작:끝:스텝], 스텝이 음수면 뒤에서부터 진행 뒤에서부터 읽는다, 끝 앞에까지!, 끝은 포함X

- s = "0123456789" / print(s[2:5]) print(s[3:]) / 234 3456789

- 문자열[:끝이 없으면 끝까지]

- 문자열.find(str) : str 문자열 찾아 인덱스 반환, 없으면 -1

- 문자열.rfind(str) : 뒤에서부터 str 문자열 찾아 인덱스 반환, 없으면 -1

- 문자열.index(str) : find()와 동일, 없으면 예외 발생

- 문자열.count(str) : str 문자열이 몇 번 등장하는지 리턴

- len(문자열) : 문자열 길이 리턴

- 단어 in 문자열 : bool 타입으로 문자열 안에 단어가 존재하는지 알려줌 (들어있어?)

- 단어 not in 문자열 : bool 타입으로 문자열 안에 단어가 존재하지 않는지 알려줌 (안 들어있어?)

- 문자열.startswitch(str) : bool 타입으로 문자열이 str로 시작하는지 알려줌 (str로 시작해?)

- 문자열.endswitch(str) : bool 타입으로 문자열이 str로 끝나는지 알려줌 (str로 끝나?)

- is로 시작하는 것들은 이게 맞나 bool로 리턴하는 함수가 대부분

- 문자열.lower() : 소문자로 바꿈, 문자열.upper() : 대문자로 바꿈

- 문자열.swapcase() : 대문자는 소문자로, 소문자는 대문자로 변환

- 문자열.capitalize() : 첫 글자는 대문자 나머지는 모두 소문자로 변환

- 문자열.title() : 모든 단어의 첫 글자를 대문자로 나머지는 소문자로 변환

- 문자열.strip() : 좌우에 있는 공백을 제거

- 문자열.lstrip() : 왼쪽에 있는 공백을 제거

- 문자열.rtrip() : 오른쪽에 있는 공백을 제거

- 문자열.split(구분자) : 구분자를 기준으로 단어를 분리하여 리스트로 리턴, 디폴트는 공백

- 문자열.splitlines() : 개행문자를 기준으로 분리, 개행문자만 있는 경우 비어있는 문자열로 처리

- 결합문자열.join(문자열) : 글자들을 결합문자열로 연결하여 하나의 문자열로 리턴

- s =".\_." / print(s.join("대한민국")) / 대.\_.한.\_.민.\_.국