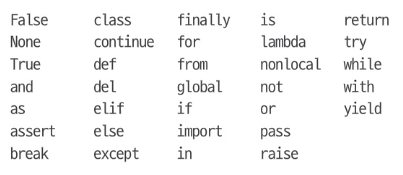
변수 Variables

: 값을 저장할 때 사용하는 식별자

변수의 자료형을 확인하는 방법 type()

* 개념상 특정 값을 저장하는 그릇
* 정수형 변수 int
* 문자열 변수 str
* 리스트 변수 list
* 부울형 변수 bool
* 변수는 가급적 하나의 자료형만을 사용할 것 (오류를 줄이기 위해)
* 변수명은 문자나 숫자로만 시작할 수 있고 대소문자 구분; 한글 가능
* 파이썬의 예약어로 사용할 수 없는 변수명



* 변수와 객체
  + var1 = 10 이라고 했을 때, var1은 변수, 10은 객체가 된다
* 변수와 자료형
  + 부울형 bool : 참, 거짓을 판단하는 표현식에 사용하는 자료형
    - True와 False 값을 가짐
    - 관계 연산자, 논리 연산자를 사용하는 표현식이 bool 값 반환
    - e.g.) p ⋀ q 에서 둘 중 하나라도 거짓이면, 이 명제는 거짓이다
      1. under18 = False
      2. male = True
      3. under18 and male
      4. False  #False와 True에 대한 논리 연산자 and의 연산 결과
  + 튜플 Tuple: ()안에 서로 다른 자료형의 값을 콤마로 구분해 하나 이상 저장할 수 있는 컬렉션 자료형
    - 개별 항목은 0부터 시작하는 인덱스를 이용해 접근할 수 있음
    - 한 번 저장된 항목은 변경 불가능
    - 튜플형 변수 선언
      1. student = (“홍길동”, 20)
      2. student[0]
      3. ‘홍길동’ 출력
      4. student[1]
      5. ‘20’ 출력
      6. student[2] → 에러 출력 (유효한 인덱스 범위가 0~1이기 때문)
      7. student[1] = 21 → 에러 출력 (한 번 지정된 값의 변경 불가능)
  + 리스트 List: []안에 서로 다른 자료형의 값을 콤마로 구분해 하나 이상 저장할 수 있는 컬렉션 자료형
    - 0부터 시작하는 인덱스를 이용해 접근할 수 있음
    - 한 번 저장된 항목이라도 변경할 수 있음
  + 셋 Set: {} 안에 서로 다른 자료형의 값을 콤마로 구분해 하나 이상 저장할 수 있는 컬렉션 자료형
    - 순서의 개념이 존재하지 않아 인덱스를 사용할 수 없음
    - 데이터 항목의 중복을 허용하지 않음
    - set형 변수 student 선언
      1. student = {“홍길동”, “이순신”, “강감찬”, “홍길동”}
      2. student |= {“을지문덕”, “이순신”} #집합의 개념을 가지고 있는 자료구조로 합집합 연잔사 | 제공
      3. student
      4. {‘강감찬’, ‘홍길동’, ‘이순신’, ‘을지문덕’} 출력
  + 사전 Dictionary: {} 안에 키:값 형식의 항목을 콤마로 구분해 하나 이상 저장할 수 있는 컬렉션 자료형
    - 키를 이용해 값을 읽어 올 수 있음
    - 항목 추가 .add()에 동일키가 없을 땐 새롭게 추가되고, 있을 땐 기존의 저장된 항목이 변경됨
    - dict[] 안에 key값을 입력하면 value 값이 출력됨
  + None: none 객체를 이용해 null 객체 상태를 표현함 (객체가 존재하지 않는 상태를 나타낸 것)
    - 파이썬은 최초 변수를 선언할 때 초기화를 하지 않으면 에러가 발생함
    - 초기화값을 제공하지 않을 경우엔 None 객체 저장 필요
    - none형 변수 선언
      1. obj
      2. NameError (아무 객체가 지정되지 않았기에)
      3. obj = None #에러방지
    - 연산자 사용
      1. obj = None
      2. obj is None #권장
      3. True 출력
      4. obj == None #잘 사용하지 않음
      5. True

변수의 생성 및 제거

* 변수는 기본적으로 한 번에 하나씩 생성하지만, 그 이상의 변수를 한 번에 생성할 수도 있음
* 변수를 생성하고 초기화하는 다양한 방법
  + 튜플 이용하기
    1. x = y = 10
    2. x
    3. 10
    4. y
    5. 10
    6. x, y = (10, 20)
    7. (x, y) = 10, 20
    8. x, y = 10, 20
    9. (x, y) = (10, 20)
* 이렇게 하면 x = 10, y = 20 으로 출력됨
* 리스트 이용

1. x, y = [10, 20]
2. [x, y] = 10, 20
3. [x, y] = [10, 20]

* 이렇게 하면 x = 10, y = 20 으로 출력됨
* 파이썬에서는 변수 생성과 초기화의 특징을 이용해 쉽게 변수의 값 교체 가능
* 변수의 제거
  + garbage collector를 이용해 객체가 사용한 메모리 공간이 자동관리 됨
  + 개발자가 메모리 관리를 직접할 필요 없음
  + 사용하지 않는 변수는 del() 함수 사용 → () 안에 변수 입력