

## 22.01.10 OPR\_D반 황정은

### ● List

#### ○ List

- list [ ] 안의 요소들(elements)은 동일한 형태로 작성
- indexing = index를 통해 특정 위치의 값을 가져오는 것을 의미
- 변수명 지정시 데이터 타입을 지정하는 경우도 있다.
  - ex) strlist = list('I love you')

#### ○ List 연산

- 연산은 요소 단위로 가능
- 문자열과 숫자의 직접적인 연산은 불가능
  - **요소의 형태가 같아야 연산이 가능하다.**
- 연산 함수
  - Slicing
    - ex) data = [1, 2, 3, 4, 5]
      - data[:2] = [1,2]
      - data[2:] = [3,4,5]
    - ex) data3 = [1,2,3,['a','b','c'],4,5]
      - data3[3][:2] = ['a','b']
  - .append(\_) = list 마지막에 요소 추가
  - .sort(\_) = list 요소 정렬
  - .reverse(\_) = list 의 순서를 반대로 정렬
  - .index(\_) = list 요소의 index 위치 찾기
  - .insert(index,object) = 특정 위치에 index를 넣고 싶은 경우
  - .remove(\_) = list 내 **특정 요소**를 삭제, 입력값을 찾아 한번만 삭제
  - .pop(index) = list 의 **index 위치의 요소**를 삭제
  - del list변수명 [index] = **index 위치의 요소**를 삭제
  - .count(\_) = 특정 요소의 개수를 센다.
  - .extend(\_) = 확장 함수(list 합침)
    - **합쳐지는 리스트의 값은 계산된 list값이 된다.**
    - ex) a =[ 1,2,3] b = [4,5,6]  
a.extend(b)  
a = [1, 2, 3, 4, 5, 6]

### ● Tuple

#### ○ Tuple indexing

- list와 차이점 : 요소(element)의 수정이 불가능

#### ○ Tuple slicing

- list slicing과 유사

ex) t1 = (1,2,'a','b')      t1[1:] = (2, 'a', 'b')

#### ○ Tuple 연산자

- list 연산과 유사

ex) t1 = (1,2,'a','b')      t2 = ('c','d')

- t1+t2 = (1, 2, 'a', 'b', 'c', 'd')
- t2\*3 = 'c', 'd', 'c', 'd', 'c', 'd'
- len(t1) = 4