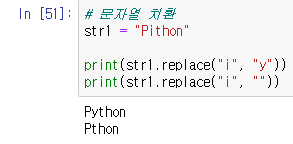
3/23 강의



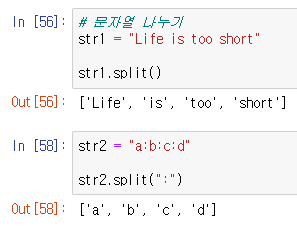
대소문자 변환 (upper, lower) : 컴퓨터는 대문자와 소문자를 완전히 다른 것으로 인지함



공백 제거 (strip) : 문자열 양쪽 끝에 남아있는 공백을 제거함 (trim) 공백도 문자 데이터의 하나임

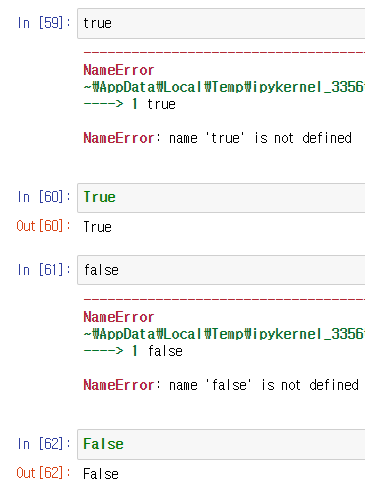


문자열 치환 (replace) : 특정 문자를 다른 문자로 변환 (감지되는 전부 다), 공백으로 바꾸면 사실상 삭제가 가능

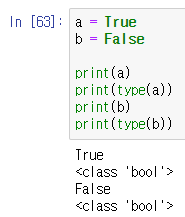


문자열 나누기 (split) : 문자열을 나눌수 있음 함수 내부를 공백으로 두면 띄어쓰기 기준으로 나누고 특정 문자를 넣어서 특정 문자 기준으로 나눌수도 있음

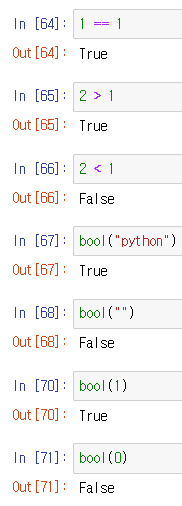
불 자료형 : 참과 거짓을 나타내는 자료형



True, False는 첫글자 대문자 지킬 것



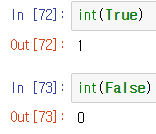
bool 타입



== : 같은 데이터인지 판단

부등호로 대소 구분 가능

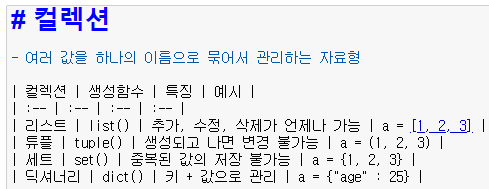
Bool() : 데이터가 값이 있으면 (텍스트, 숫자 구분 없이) True, 값이 없거나 0이면 False

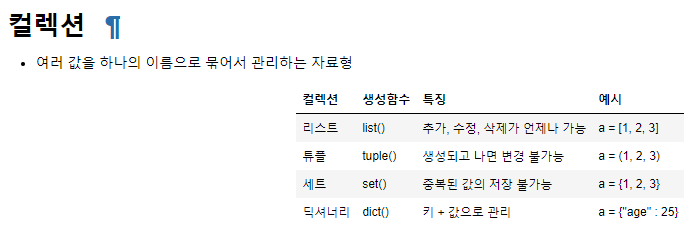


Int 함수 : 정수형으로 변환. True는 1, False는 0으로 변환됨



컬렉션





마크다운에서 | (\ + shift) 로 표 생성 가능

첫번째 줄은 컬럼 헤더 설정

:-- 왼쪽정렬

--: 오른쪽정렬

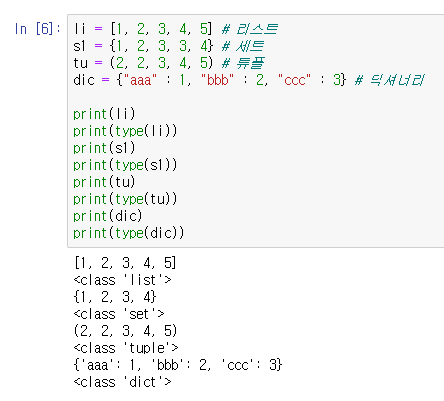
:--: 양쪽정렬

리스트 (list) : 데이터 세트 함수중 가장 많이 씀

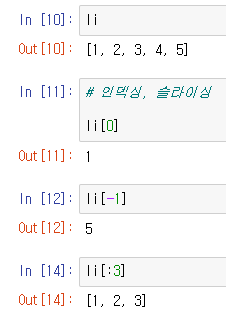
튜플 (tuple) : 데이터 처리에 리소스가 적게 드므로 데이터 전달이나 수정되면 안될 때 사용

세트 (set) : 중복값을 넣어도 중복값들이 제거됨 데이터 정리 가능

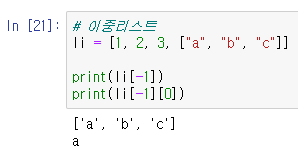
딕셔너리 (dict) : 데이터의 1:1 관리에 사용 키-값이 다른 키-값과 중복되지 않음



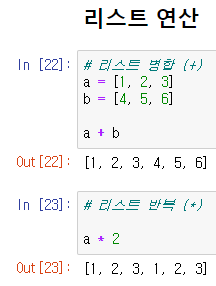
리스트



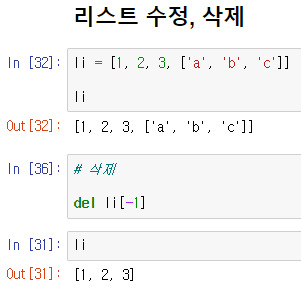
리스트도 데이터 단위로 인덱싱 슬라이싱이 가능함



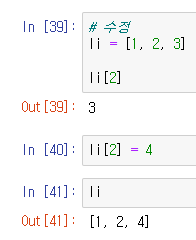
다중리스트 : 리스트 자체도 리스트 데이터 단위가 될 수 있음 이 리스트도 단독으로 뽑으면 리스트에 대한 연산 가능



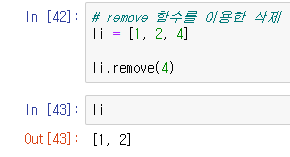
문자열처럼 리스트는 리스트 병합과 반복 연산이 가능함

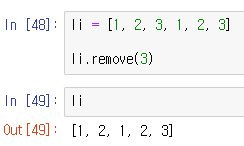


리스트 내부 특정 데이터를 삭제 가능 (인덱스 값으로)

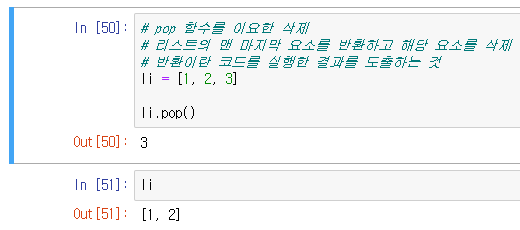


수정도 인덱스값 기반으로 가능

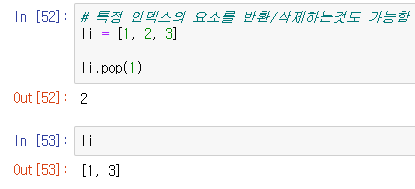




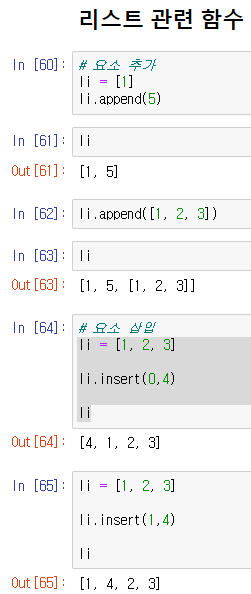
Remove 함수는 리스트에서 첫 번째로 나오는 특정 요소를 삭제하는 함수



Pop 함수 : 리스트 내 마지막 요소를 지우고 난 후 해당 값을 반환함

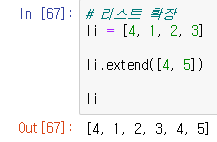


인덱스 값을 이용하여 특정 위치를 반환 및 삭제하는 것도 가능하다



Append : 리스트의 마지막에 요소를 추가함 (리스트 자체도 요소로서 들어감)

Insert(a, b) : 인덱스 값 기반으로 a 번째 위치에 4가 들어오도록 삽입

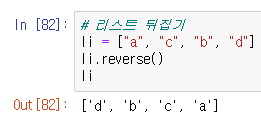


Extend(리스트) : 리스트를 뒤에다가 합침 (+랑 거의 같음. +가 함수 대상이 바뀌는건 아님… extend = 리스트 = 리스트+리스트2)

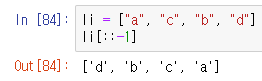


리스트 오름차순/내림차순 정렬 sort 함수

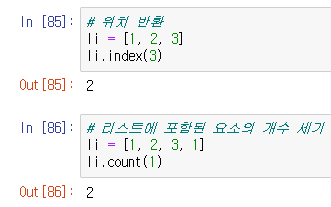
같은 타입의 대소 or 순서를 구분 가능한 요소끼리만 가능함



reverse 함수 : 리스트의 순서를 거꾸로 뒤집음

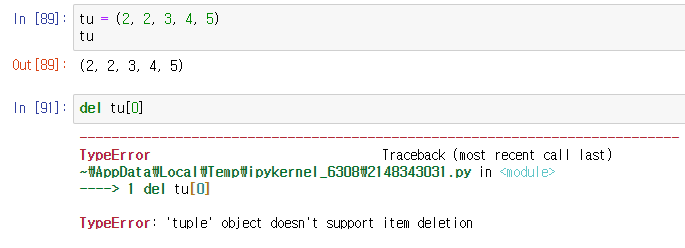


슬라이싱으로도 뒤집을수 있으나 함수 대상이 실제로 바뀌지는 않음

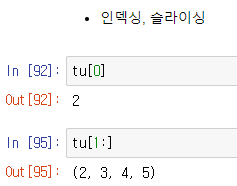


문자열에 사용되는 index, count 함수도 그대로 사용 가능

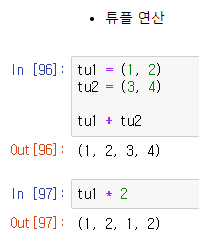
튜플



튜플은 데이터의 수정이 불가능함

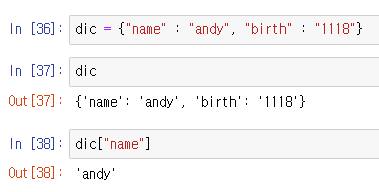


인덱싱, 슬라이싱은 튜플을 조작하는 것이 아닌 데이터를 추출하는 것이므로 적용 가능함



튜플을 건드는 것이 아닌 연산은 실행 가능함 (append 같은건 안됨)

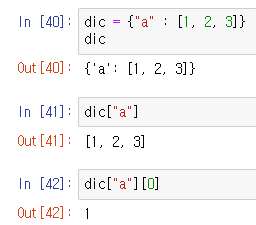
딕셔너리



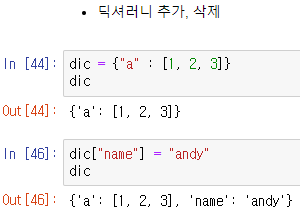
딕셔너리는 각 값에 이름표를 붙인 것

리스트에서 인덱스 값도 수동으로 지정을 한것으로 이해하면 편함 (name = 인덱스 0)

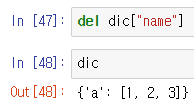
Key 값은 immutable만 가능! (문자열, 튜플 등)



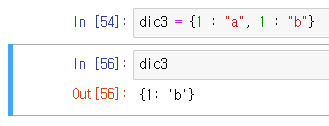
리스트같은것도 들어감



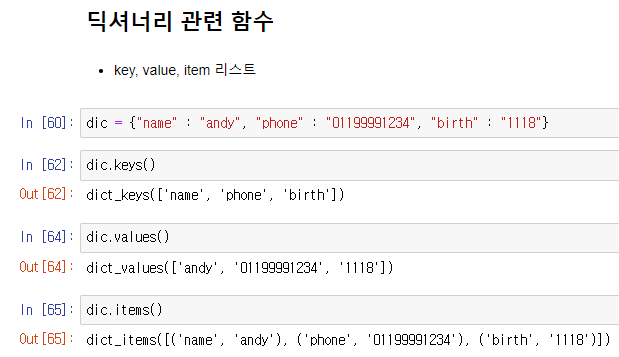
딕셔너리에 주소와 값을 지정하여 추가 가능



삭제를 할때도 주소만 써주면 값도 세트로 삭제됨



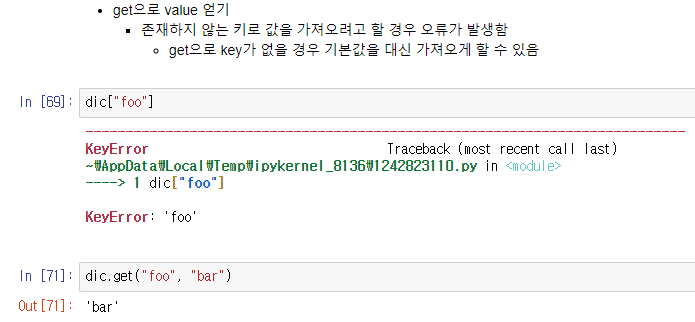
딕셔너리는 동일한 key (주소)가 존재할 수 없음 동일한 주소를 세팅하면 나중에 입력한것으로 덮어씌워짐



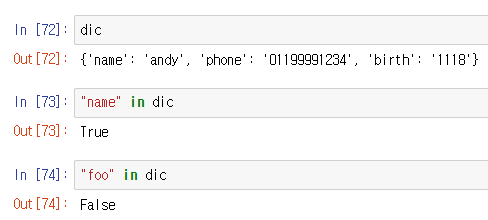
Keys 함수 : 딕셔너리의 key 리스트를 모두 출력

Values 함수 : 딕셔너리의 value 리스트를 모두 출력

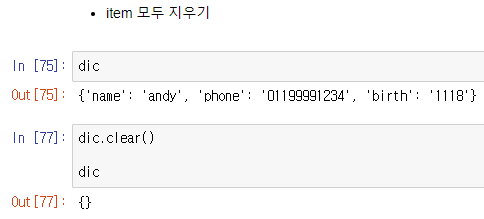
Items 함수 : 딕셔너리의 key – value 매칭을 튜플 형식으로 출력 딕셔너리가 딕셔너리 형태로 출력되면 안될 때 사용함



Get 함수 : 지정한 키에서 값을 찾아 오되 해당 키가 없으면 지정한 값을 반환 (엑셀 IFERROR 같은거)



In 함수 : 딕셔너리에 키 값이 있는지 여부를 True / False로 반환함



Clear 함수 : 딕셔너리 내의 키-값을 몽땅 지움