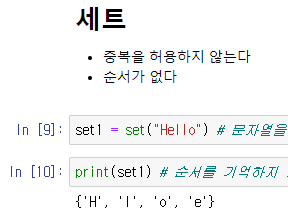
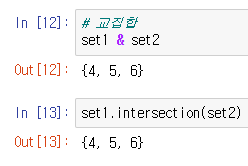
3/28 강의

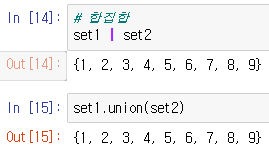


세트는 순서가 없고 중복을 허용하지 않음

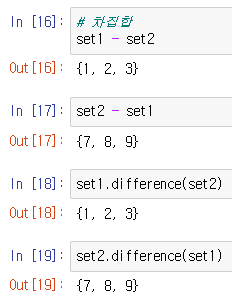
집합 개념을 사용할수 있음



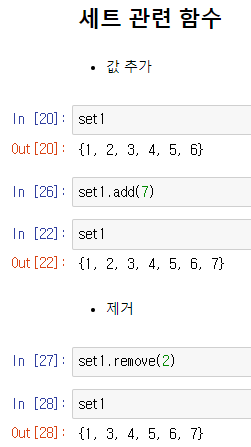
세트의 교집합 & or intersection



세트의 합집합 | or union

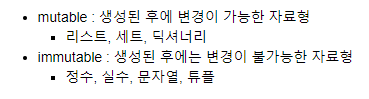


세트의 차집합 – or difference

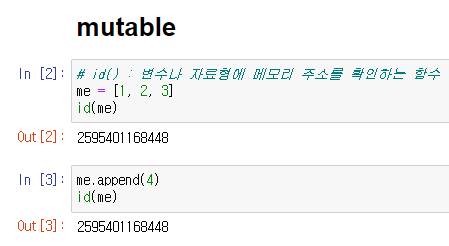


세트에서 값을 추가하거나 제거 가능

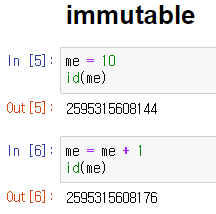
세트는 리스트에서 중복값을 제거할 때 많이 쓰임 (리스트 -> 세트 -> 리스트 변환)



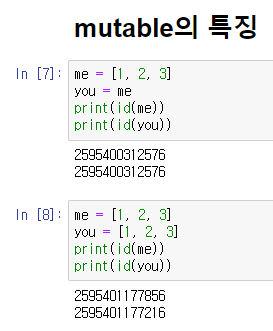
Mutable과 immutable



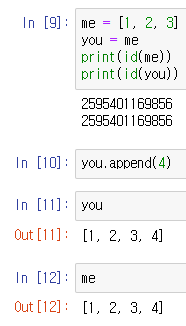
Mutable은 변형이 일어나도 메모리 주소가 변하지 않는다 (변경이 가능하므로 해당 메모리 내에서 변형이 일어남)



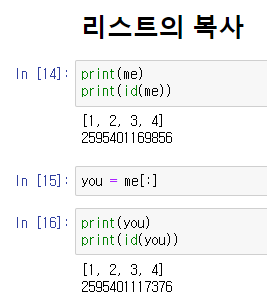
Immutable은 값이 변하지 않음. 그러므로 해당 변수를 바꾸었을 경우 새로운 id로 배정되어서 메모리에 저장됨



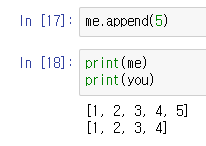
파이썬은 메모리를 효율적으로 사용하기 위해서 같은 데이터를 지정할 경우 다른 변수라도 id가 같아짐 (별도로 똑같이 설정하면 id가 달라질 수 있음)



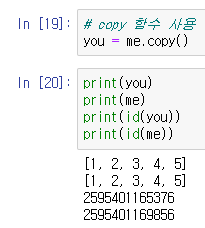
같은 id에 배정된 mutable들은 하나만 변경이 일어나면 다른 mutable도 모두 변경됨



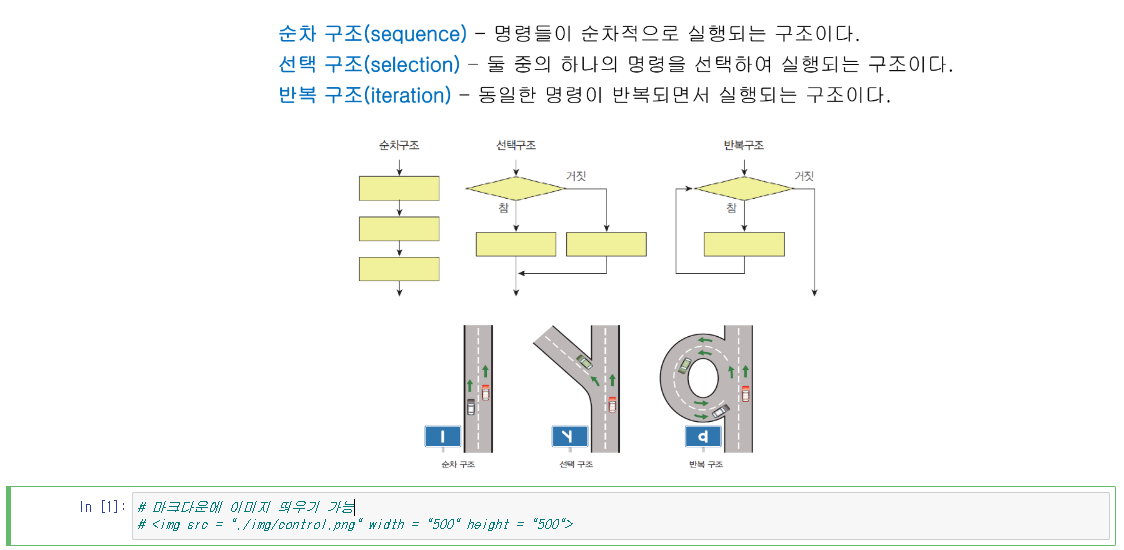
리스트의 경우 슬라이싱을 처음부터 끝까지 하는 것으로 다른 id에 리스트를 복사시킬수 있다



Id가 다르므로 한쪽의 리스트가 변형되어도 다른 리스트는 변형되지 않는다



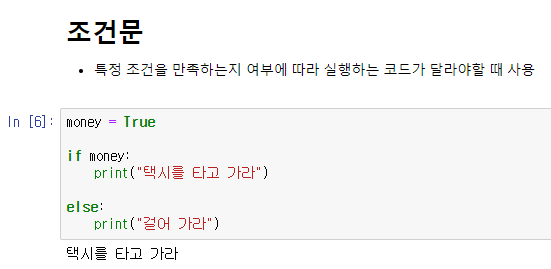
Copy 함수로도 복사 가능



마크다운에는 이미지 첨부 가능

절대 경로 : ?드라이브 포함한 모든 경로를 다 입력

상대 경로 : 기본 작업 폴더까지 들어온 상태에서 나머지 주소 입력 ex.) “./img/control.png”



조건문

If (조건코드):

(들여쓰기 = Tab) 코드1

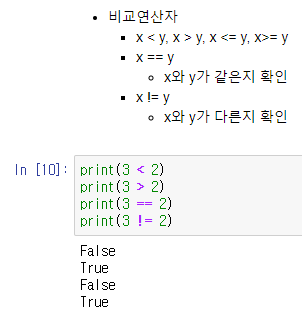
Else:

(들여쓰기 = Tab) 코드2

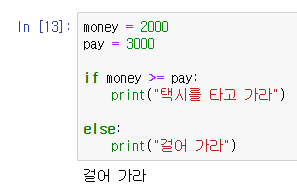
조건문의 True, False 여부에 따라 실행하는 코드가 달라짐 (코드1 : True, 코드2 : False)

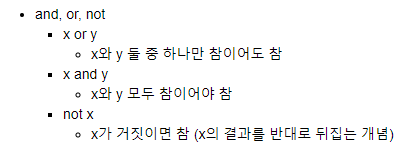
If / else 뒤 들여쓰기에 있는 코드들 = 코드블럭, 조건문에서 실행할 코드 단위를 설정하는 것으로 매우 중요함, 들여쓰기가 없는 줄로 넘어가게 되면 if / else의 코드블럭이 끝난 것으로 간주

콜론(:) 뒤의 Tab이 코드블럭을 정의함

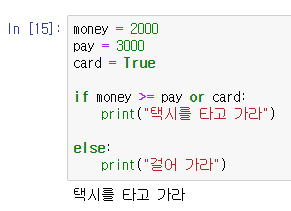


조건식 : 비교연산자… !=은 다를경우 True 반환

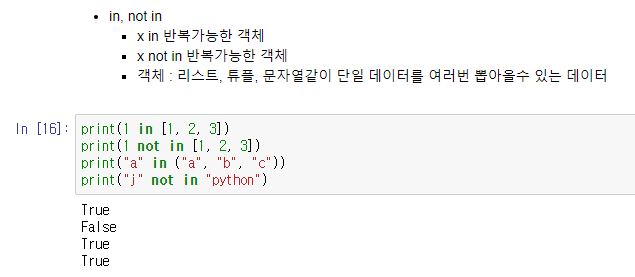




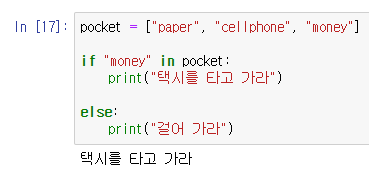
And, or, not



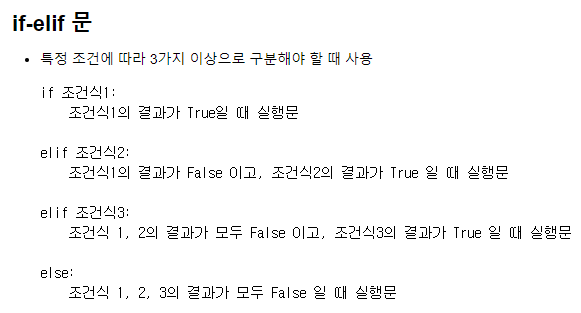
or 사용



In, not in 조건문



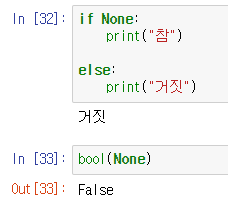
사용 예시



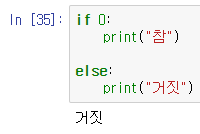
Elif를 이용하면 다중 if 사용 가능함



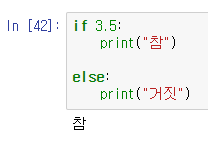
Elif와 다중 if 사용



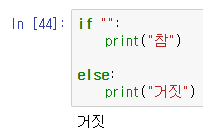
None : 값이 없음을 나타내는 데이터 형태 (거짓 취급)



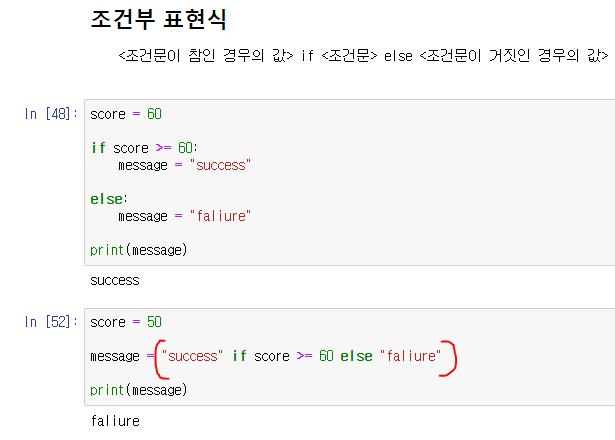
0도 거짓 취급 (False의 int형, 1은 참)



데이터가 있는 값도 참 (문자열을 넣어도 데이터가 있는 것이므로 참)



빈 문자열은 거짓



조건부 표현식으로 if를 줄일수도 있음

이 식은 단일 값에 대한 것이므로 조건에 따라 지정하는 변수가 다를 경우에는 조건부 표현식을 사용할 수 없음 (참일 경우 a 변수에 부여, 거짓일 경우 b 변수에 부여 등) 일반 if문을 사용해야됨