

Hash

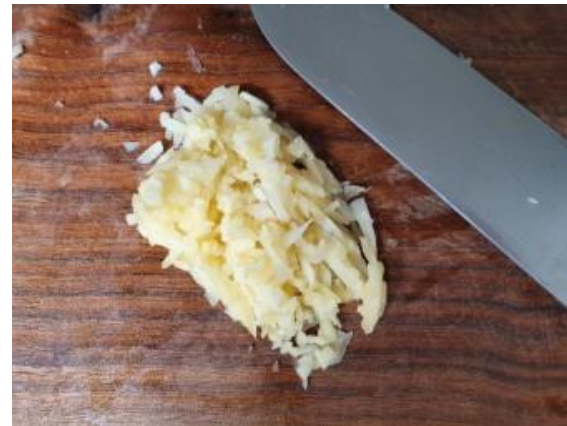
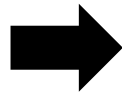
해시(Hash)?

hash

미국·영국[hæʃ]  영국식 

명사

- 1 해시(고기와 감자를 잘게 다져 섞어 요리하여 따뜻하게 차려 낸 것)

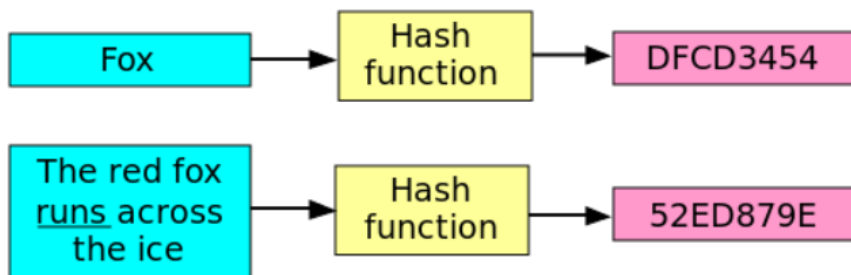


IT에서의 해시화 : 평문의 데이터를 감자 다지듯 잘게 도마질해서 원래 어떤 형태였는 지 복원할 수 없는 비가역적 데이터로 만드는 것.



비밀번호 저장, 블록체인 등에서 사용

- 입력값을 일정한 규칙에 의해 정수로 변환하는 함수



<https://www.convertstring.com/ko/Hash/SHA256s>

코드

```
print(hash("nice"))
print(hash("very nice"))
print(hash((12,3,5,6)))
print(hash(2000.1))
print(hash([1,2,3]))
```

```
3123681602639128896
-7310242060864729103
6207878451375355657
230584300921161680
TypeError: unhashable type: 'list'
```

■ hashable

- **hash()** 함수에 값을 넣어서 결과를 반환 받을 수 있는 객체
- 숫자, 문자, 튜플 같은 불변(immutable) 객체

■ unhashable

- 리스트, 딕셔너리, 셋

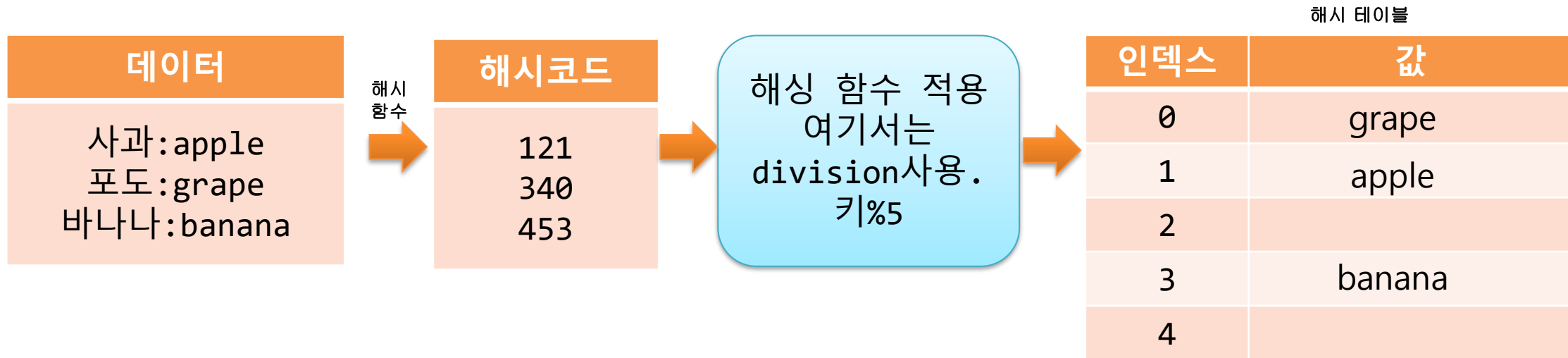


리스트, 딕셔너리, 셋이 **unhashable**인 이유

변경가능한 객체들은 해시를 만들고도 중간에 데이터가 변경되면 해시값도 매번 변경되어야 하기 때문임.

해시 테이블(Hash table)

- 파이썬의 딕셔너리는 해시 테이블로 구현됨.



해시테이블을 사용하면 빠른 속도로 데이터를 검색, 삭제할 수 있다.

연산의 (평균)시간복잡도가 $O(1)$



해싱 함수: division, mid-square, folding, ...



복잡도 : 알고리즘의 성능 평가 척도

시간 복잡도: 알고리즘 수행에 필요한 연산의 횟수.

공간 복잡도: 알고리즘 수행에 필요한 메모리 크기.

$$1+2+3+\dots+100 = 5050$$

$$100*(100+1)/2 = 5050$$

99번의 연산

3번의 연산

■ 빅오(Big-O) 표기법을 주로 사용.

	$\log n$	n	$n \log n$	n^2	n^3	2^n
$O(1)$						
$O(\log n)$	0	1	0	1	1	2
$O(n)$	1	2	2	4	8	4
$O(n \log n)$	2	4	8	16	64	16
$O(n^2)$	3	8	24	64	512	256
$O(n^3)$	4	16	64	256	4096	65536
$O(2^n)$	5	32	160	1024	32768	4294967296

참고: n 은 입력 데이터의 개수

파이썬 리스트와 딕셔너리의 시간 복잡도

딕셔너리	
연산	시간복잡도
<code>len(d)</code>	$O(1)$
<code>d[key]</code>	$O(1)$
<code>d[key]=value</code>	$O(1)$
<code>key in d</code>	$O(1)$

리스트	
연산	시간복잡도
<code>len(li)</code>	$O(1)$
<code>li[i]</code>	$O(1)$
<code>li.append(e)</code>	$O(1)$
<code>key in li</code>	$O(n)$