

chapter5. 해시 테이블

- Hello Coding 그림으로 개념을 이해하는 알고리즘 -



TABLE OF CONTENTS



1. 해시 함수
2. 해시 테이블 사용하는 예
 - 중복된 항목 방지하기, 전화번호부, DNS 작업 등
3. 핵심 내용 정리

1. 해시 함수

해시 함수는 문자열(여기에서 문자열은 임의의 데이터, 즉 바이트 열을 뜻함)을 받아서 숫자를 반환하는 함수입니다.

기술 용어로 말하면 **해시 함수는 문자열에 대해 숫자를 할당**합니다.

〈해시 함수 요건〉

1. 일관성이 있어야 한다. 예를 들어 apple을 넣었을 때 4를 반환한다면 apple을 넣을 때 마다 항상 4가 반환되어야 함.
2. 다른 단어가 들어가면 다른 숫자가 나와야 한다. 서로 다른 단어에 대해 모두 서로 다른 숫자가 나와야 한다.

해시 함수와 배열을 합쳐서 해시 테이블이라고 부르며 자료구조 중 쓸모가 많다.

해시테이블은 **해시 맵, 맵, 딕셔너리, 연관 배열**이라는 이름으로도 알려져 있다.

chapter2에서 배웠 듯이 배열은 원소를 순간적으로 얻을 수 있습니다. 해시 테이블도 자료를 저장할 때 배열을 사용하며 마찬가지로 속도도 빠릅니다.

★★★파이썬에는 딕셔너리라는 해시 테이블이 있습니다.★★★

2. 해시 테이블을 사용하는 예

해시 테이블은 다음과 같은 일을 하고자 할 때 좋습니다.

- ▶ 어떤 것을 다른 것과 연관시키고자 할 때
- ▶ 무언가를 찾아보고자 할 때

<전화번호부>

```
phonebook = {"안지찬": "010-5454-3483",  
             "김채리": "010-1234-5678",  
             "이순신": "010-3333-5555"  
}
```

```
print(phonebook["안지찬"])
```

010-5454-3483

<DNS 확인작업>

```
DNS_resolution = {  
    "google.com": "74.125.239.133",  
    "facebook.com": "173.252.120.6",  
    "scribd.com": "23.235.47.175"  
}
```

2-1. 중복된 항목을 방지하기

```
voted = {"김사과":True}

def check_voter(name):
    if voted.get(name):
        print("돌려보내세요")
    else:
        voted[name] = True
        print("투표하게하세요")

check_voter("안지찬")
check_voter("김사과")
check_voter("안지찬")
```

```
투표하게하세요
돌려보내세요
돌려보내세요
```

투표소를 관리한다고 가정해봅시다.

모든 사람은 한번 씩 투표 가능합니다.

만약 리스트로 관리한다면 누군가가 투표하러 올 때마다 이미 투표했는지 확인하기 위해 긴 목록을 뒤져봐야 합니다.

하지만 이보다 훨씬 더 빠른 방법이 있습니다.

바로 해시 테이블입니다 !

리스트에 투표자를 저장한다면 리스트 전체를 단순탐색해야 해서 정말 느리겠지만 해시 테이블은 이름이 있는지 없는지 순간적으로 알려주기 때문에 중복확인 하는것도 해시테이블이 훨씬 빠릅니다.

3. 핵심 내용 정리

1. 해시테이블은 해시 함수와 배열을 결합해서 만듭니다.
2. 충돌을 줄이는 해시 함수가 있어야 합니다.
3. 해시 테이블은 정말 빠른 탐색, 삽입, 삭제 속도를 가집니다.
4. 해시 테이블은 어떤 항목과 다른 항목의 관계를 모형화하는데 좋습니다.
5. 사용률이 0.7보다 커지면 해시 테이블을 리사이징 할 때입니다.
6. 해시 테이블은 (웹 서버 등에서) 데이터를 캐싱하는 데도 사용됩니다.
7. 해시 테이블은 중복을 잡아내는 데도 뛰어납니다.