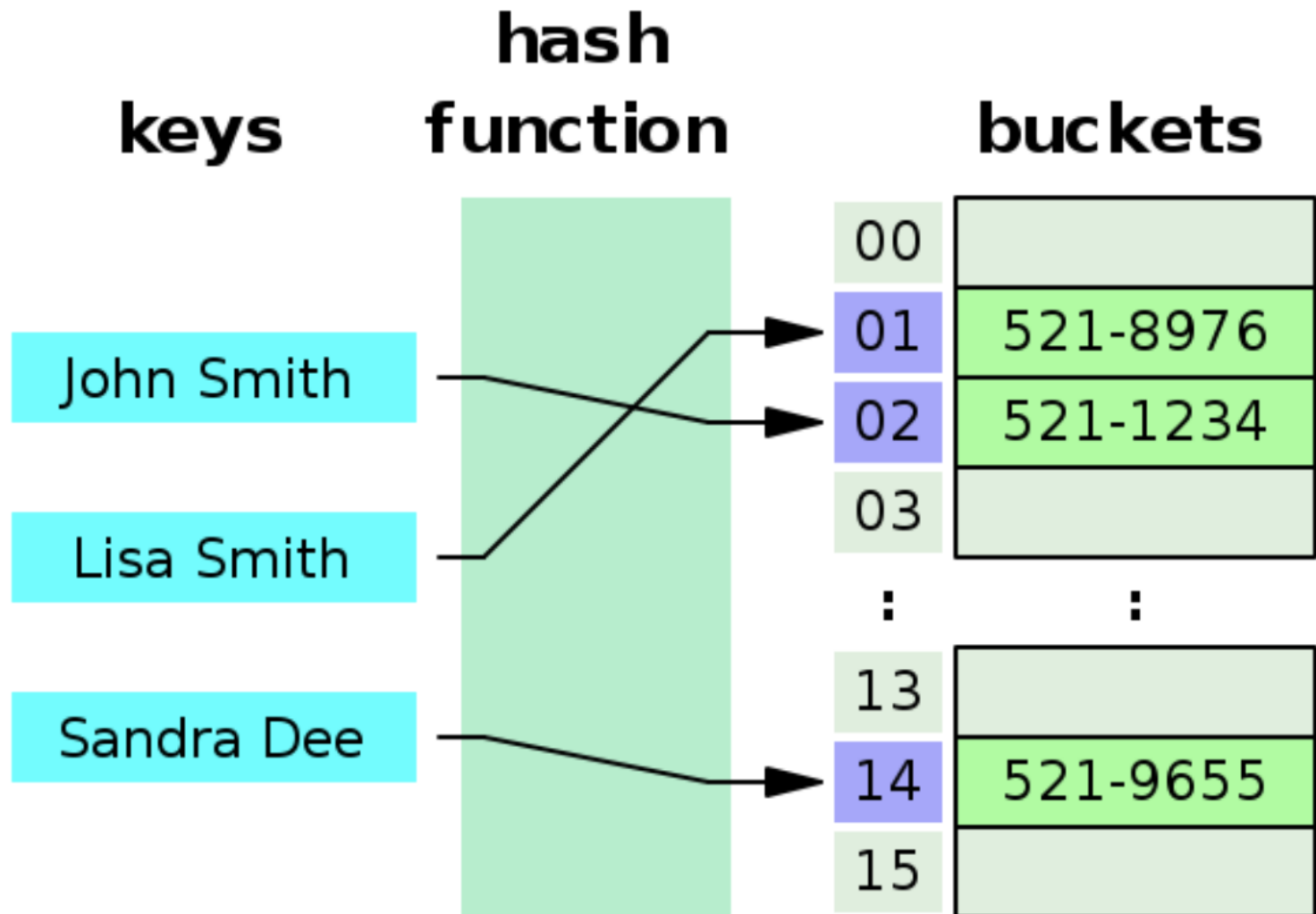


- 해시맵이란?
- 해싱과 해시함수
- 충돌(Collision)
- 충돌 해결
- Hashtable vs HashMap (in java)
- Enumeration vs Iterator

해시맵이란?

- Key-Value 방식의 자료구조
- $O(1)$ 의 시간복잡도 -> 빠름



해싱과 해시함수

해싱

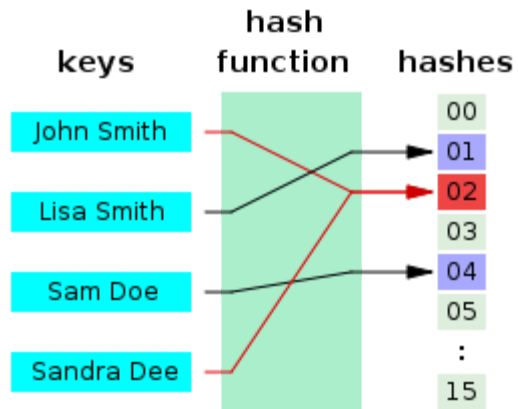
- 가변적 길이의 데이터를 고정적 길이의 데이터로 만듦
- 데이터의 효율적 관리가 목적

해시함수

- 해싱할 수 있도록 하는 함수

충돌(Collision)

- 해쉬함수는 기본적으로 One-to-Many 매핑
- 각자 다른 Key가 동일한 해시값을 가질 수 있음

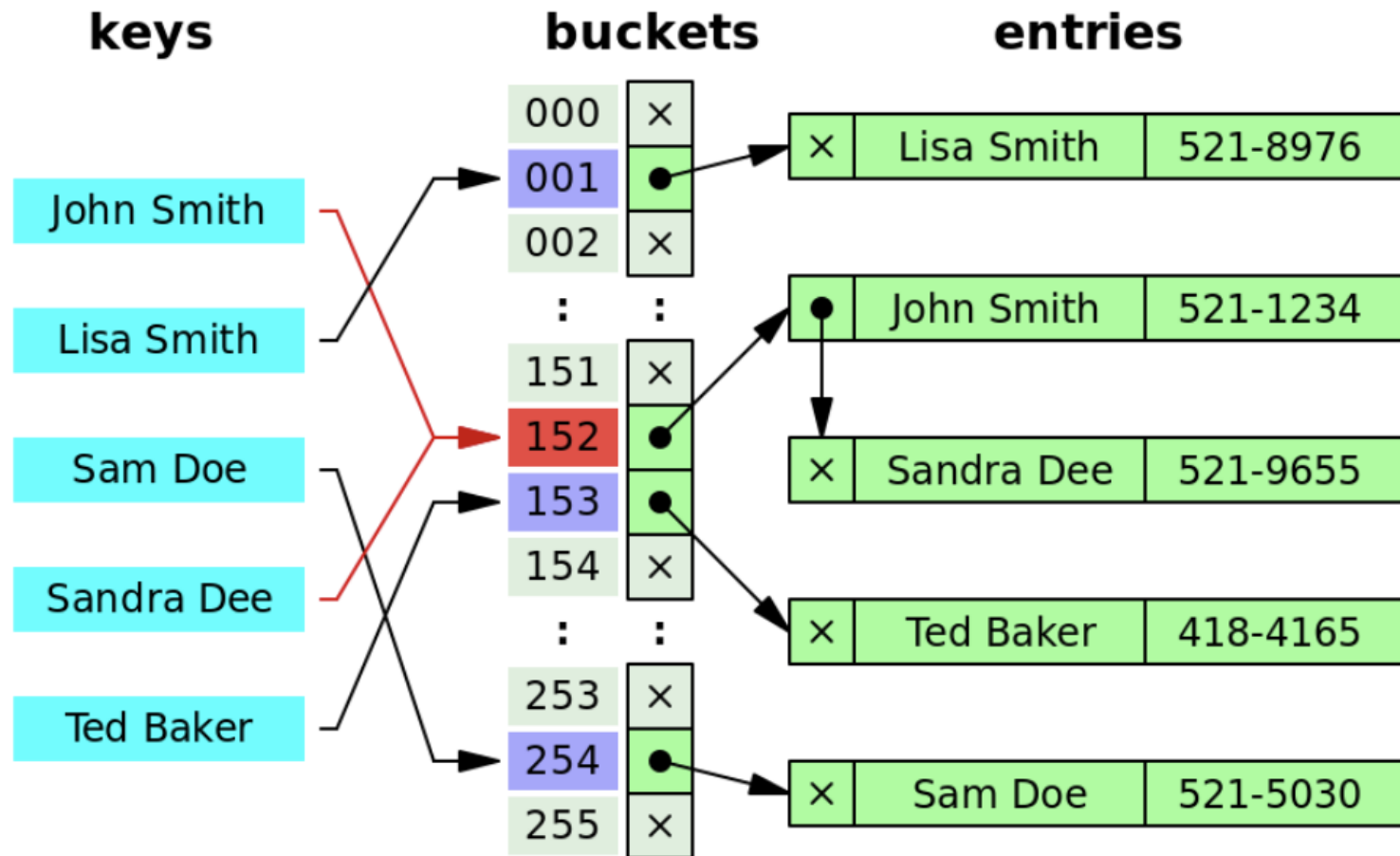


충돌 해결

1. Seperate Chaining
2. Open Addressing

1) Separate Chaining

- Linked List를 이용

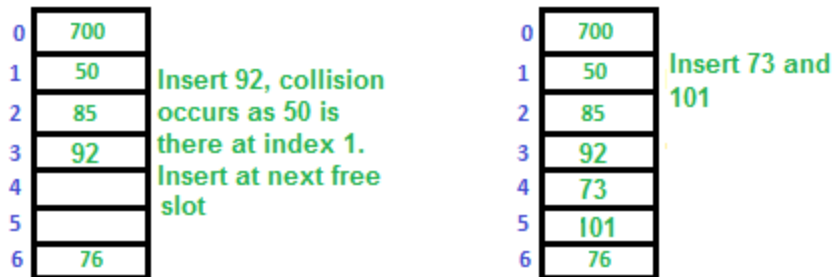
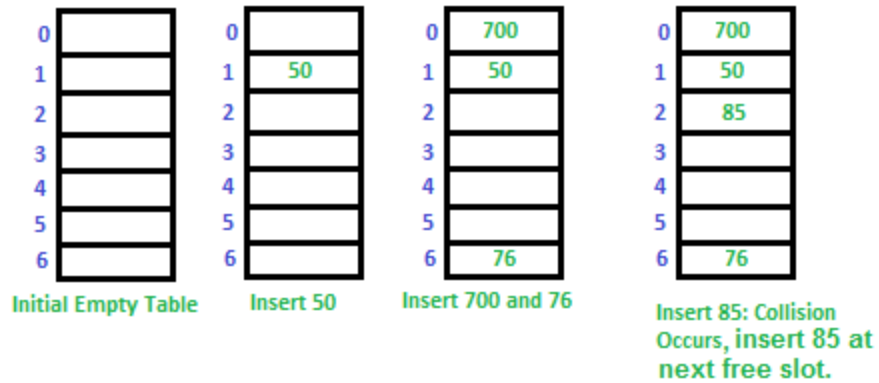


- JDK가 사용하고 있는 충돌처리방식
- Java 8에서는 충돌 Key-Value 쌍 개수가 8개 이상이 되면 Linked List를 **Tree**로 변경한다.
- Red-Black Tree로 구현되어 있다고 한다.
- 해당 값이 6개가 되면 다시 Linked List로 변경한다.
- 차이가 8-7이 아닌 8-6인 것은 잦은 변경으로 인한 성능저하를 방지하기 위함.
- 유연하다는 장점이 있으나 메모리 문제를 야기함

2) Open Addressing

- a) Linear Probing
- b) Quadratic Probing
- c) Double hashing

a) Linear Probing (선형 탐사)



해시함수 -> 키 값을 8로 나눈 나머지

0	1	2	3	4	5	6	7
		10	18	26			

18을 삭제

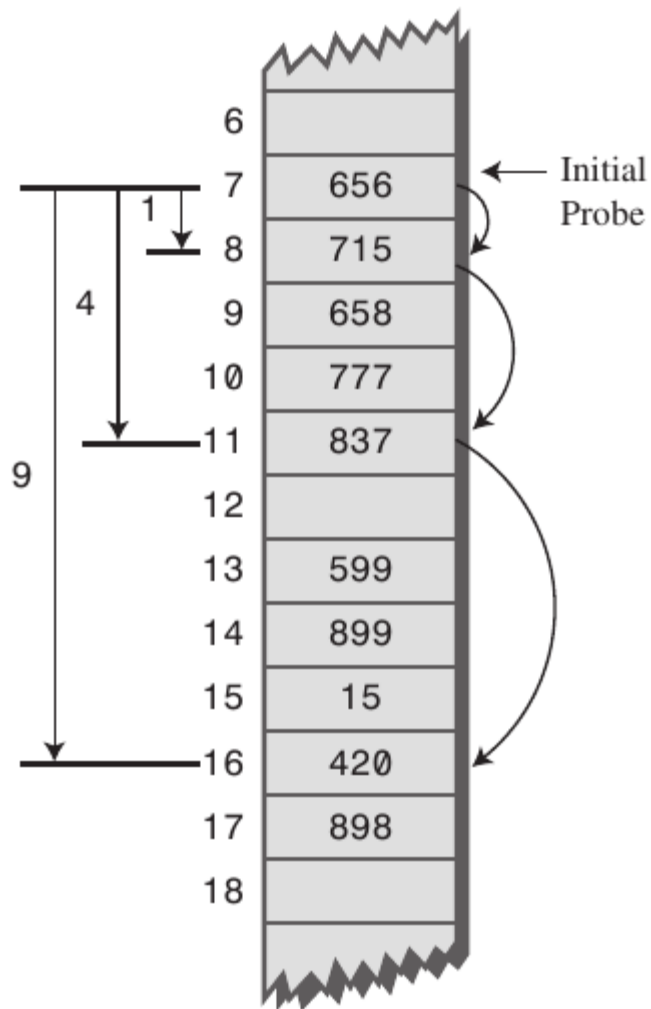
0	1	2	3	4	5	6	7
		10		26			

0	1	2	3	4	5	6	7
		10	DEL	26			

-> DEL 이라는 Dummy node를 삽입한다. Dummy node는 실제 값은 가지지 않지만 검색 시 연결해주는 역할을 한다.

-> 특정 해시값 주변이 모두 채워져 있는 **Primary Clustering**의 경우, 검색에 취약하다.

b) Quadratic Probing (제곱 탐사)



-> 동일한 초기 해시값(Initial Probe)을 갖는 **Secondary Clustering**의 경우, 검색에 취약하다.

c) Double Hashing

2개의 해시함수

1. 최초의 해시값을 얻는 해시함수

2. 해시충돌이 일어났을 때 탐사 이동폭을 얻기 위한 함수

-> 최초 해시값이 같더라도 탐사 이동폭이 달라지고, 탐사 이동폭이 같더라도 최초 해시값이 달라짐.

-> Primary Clustering, Secondary Clustering을 모두 완화할 수 있음.

- 메모리 문제가 발생하진 않으나 해시충돌이 발생할 수 있음.

Hashtable vs HashMap (in java)

1. Synchronization
2. Return Value

1) Synchronization

- HashMap -> 동기화 지원 X
- Hashtable -> 동기화 지원 O
- Hashtable은 동기화 처리에 드는 비용 때문에 HashMap에 비해 느림
-> HashMap을 사용하되 동기화가 필요한 시점에서는 Java 5부터 제공하는 ConcurrentHashMap을 사용하는 것을 권장함

2) Return Value

- HashMap -> Fail-Fast-Iterator
- Hashtable -> Enumeration

Enumeration vs Iterator

- Collection에 저장된 요소를 접근하는데 사용되는 인터페이스
- Enumeration is an older interface.
- Iterator is a newer interface.

Iterator

- 메소드 네이밍 간략화
- remove() 메소드 추가
- ConcurrentModificationException을 통한 무결성 보장 (Fail-Fast-Iterator)

-> According to Java API Docs, Iterator is always preferred over the Enumeration.

Reference

해쉬 테이블의 이해와 구현 (Hashtable)

[JAVA] HashMap과 Hashtable 차이

HashMap의 해시충돌(hash collision)과 자바의 대응책

해싱, 해시함수, 해시테이블

Differences Between Enumeration Vs Iterator In Java

How Hashtable and HashMap key-value are stored in the memory?