자료구조 실습 보고서

[제08주] 컬렉션 & 해시테이블

2018/05/07

201402414 장수훈

```java
package H08;

import java.util.*;

public class TestFrequency {
    public static void main(String[] args) {
        new TestFrequency();
    }

    public TestFrequency() {

        String[] countries = {"KO","DE","ES","FR","DE","ES","DE"};
        List list = Arrays.asList(countries);
        System.out.println("frequency(list, [DE]) : "+frequency(list, "DE"));
        System.out.println("frequency(list, [KO]) : "+frequency(list, "KO"));
        System.out.println("frequency(list, [ES]) : "+frequency(list, "ES"));
        System.out.println("frequency(list, [FR]) : "+frequency(list, "FR"));
    }

    int frequency(List list, Object object) {
        int count = 0;
        Iterator it = list.iterator();
        while(it.hasNext()) {
            if(it.next().equals(object)) count++;
        }
        return count;
    }
}

```

<TestFrequency 클래스>

[6~8] new TestFrequency() 으로 TestFrequency 를 호출하였다.

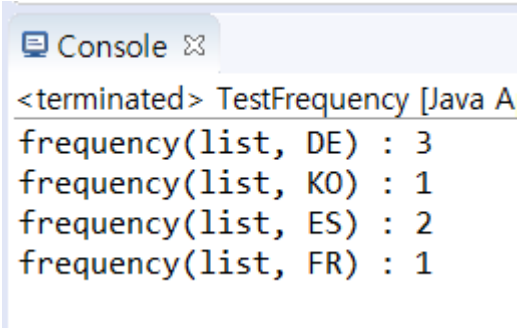
[10~18] 실제 입출력을 담당하는 함수로써 java.util.* 을 import 해서 리스트 라이브러리를 사용하고, 반복자를 사용하여 리스트 내부에 같은 원소의 개수를 출력하였다.

내부 변수는 과제 pdf 에서 주어진대로 지정하였다.

[20~27] 반복자로 list 의 내부를 탐색하면서 같은 애용이 몇 개 들어있는지 확인하는 함수

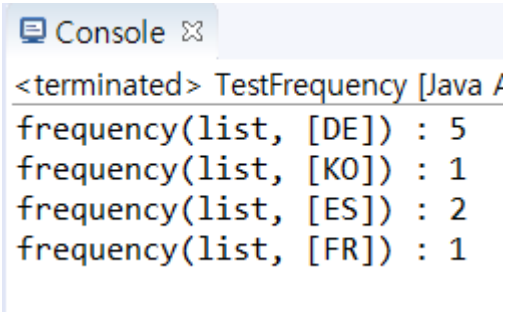
<결과화면 출력>

```
 {"KO","DE","ES","FR","DE","ES","DE"};
```



```
{"KO","DE","ES","FR","DE","ES","DE","DE","DE"};
```

"DE"를 2개 추가해보았다.



<HashTable>

1. loadFactor 값이 무엇을 의미하는가

   - ```
     if(used>loadFactor*entries.length) rehash();
     ```

   - 설정한 배열의 length를 늘릴지 말지 조건을 거는 변수

2. hashCollision이 발생하는 이유가 무엇인가.

   - 해시 메소드가 다른 두개의 입력값에 대해 동일한 출력값을 내기 때문

3. hashCollision이 발생 할 때 해당 함수는 선형 조사로 문제를 해결하고 있는데, 해당 함수에서 적용한 선형 조사 방법이 무엇인가.

   - 충돌이 일어나면 테이블의 다음 빈 공간을 탐색하여 저장하고, 빈곳이 없으면 rehash함수를 통해 공간을 늘린후 데이터 저장.

4. 선형 조사 방법을 사용했을 때 단편화가 발생할 수 있는 이유가 무엇인가.

   - 해시 테이블의 크기가 제한되어있고 ,빈공간을 찾아야 하기 때문에 제한이 있다.

5. 해당 함수는 key를 해쉬해서 나온 값으로 entry를 직접 접근해서 데이터를 반환하는데, hashCollision이 발생했을 때와 hashCollsion이 발생하지 않았을 때 entry를 접근하는 방법의 차이를 설명

   - hashCollision이 발생 -> entry가 key를 rehash메소드를 거쳐 나온값을 참조

   - hashCollision이 발생x → entry가 key를 해시에서 나온값을 참조
     ```