**20회차(1)**

package test;

import java.util.\*;

import java.util.Scanner;

public class Test01 {

public static void main(String[] args) {

/\* 슈퍼마켓 상품 등록 프로그램 가정

\* 아래와 같은 형태가 되도록 map을 이용하여 구현

\* 상품이름을 key로 하고, 가격을 값으로 가지는 map을 생각

\* 상품이름 : 참이슬

\* 상품가격 : 1200

\* 상품등록이 완료되었습니다...현재 상품 1개가 등록되어 있습니다.

\* 상품이름 : 허니버터칩

\* 상품가격 : 1500

\* 상품등록이 완료되었습니다...현재 상품 2개가 등록되어 있습니다.

\* 상품이름 : 참이슬

\* 상품가격 : 1500

\* 이미 등록된 상품입니다. 가격을 변경하겠습니까?(Y/N)

\* (Y) 가격 변경이 완료되었습니다.

\* (N) 등록이 취소되었습니다.

\* 상품이름 : 종료

\* 실행을 종료합니다.

\*/

// int 타입의 HashMap 객체를 생성할 때는 -> Integer 클래스 사용

Scanner sc = new Scanner(System.in);

HashMap<String, Integer> map = new HashMap<String, Integer>();

System.out.println("슈퍼마켓 상품 등록 프로그램");

while (true) {

System.out.print("상품이름 : ");

String name = sc.next();

// 종료 체크

if (name.equals("종료")) {

System.out.println("실행을 종료합니다.");

break;

}

System.out.print("상품가격 : ");

int price = sc.nextInt();

// 해당 상품이 존재하는지 확인

// 상품이름이 key가 되기 때문에 containsKey() 메소드

if (map.containsKey(name)) {

// 키가 존재하면? 상품이 존재하는 것

// 가격을 업데이트

System.out.println("이미 등록된 상품입니다. 가격을 변경하겠습니까?(Y/N)");

String me = sc.next();

// 대소문자를 구분하지 않기 위해서 하나로 통일

// 문자열 메소드에서 toUpperCase(), toLowerCase() 중에 하나를 사용

me = me.toUpperCase();

if (me.equals("Y")) {

map.replace(name, price);

System.out.println("(Y) 가격 변경이 완료되었습니다.");

} else {

System.out.println("(N) 등록이 취소되었습니다.");

}

} else {

// 키가 존재하지 않으면? put() 메소드를 사용해서 등록처리

map.put(name, price);

System.out.printf("상품등록이 완료되었습니다...현재 상품 %d개가 등록되어 있습니다.\n",map.size());

}

}

}

}

**20회차(2)**

package test;

import java.util.\*;

import java.util.Scanner;

public class Test02 {

public static void main(String[] args) {

// 이름과 전화번호를 저장하고 관리하는 프로그램을 작성

// map을 이용해서 구현

// 전화번호는 숫자가 아닌 문자열로 취급(010-xxxx-xxxx)

// 데이터 추가, 삭제, 수정, 보기

// 1. 추가 2. 삭제 3. 수정 4. 보기

// 이름과 전화번호는 중복될 수 없습니다

HashMap<String, String> map = new HashMap<String, String>();

Scanner sc = new Scanner(System.in);

System.out.println("이름/전화번호 관리 프로그램");

System.out.println("1. 추가 2. 삭제 3. 수정 4. 보기");

while (true) {

String name;

String phone;

int menu;

System.out.print("원하시는 메뉴를 입력하세요 : ");

menu = sc.nextInt();

if (menu == 1) {

// 1. 추가 -> 중복을 고려해서 작성

System.out.print("이름 입력 : ");

name = sc.next();

System.out.print("전화번호 : ");

phone = sc.next();

// 중복 처리

if (map.containsKey(name)) {

System.out.println("이미 등록된 정보가 있습니다.");

} else if (map.containsKey(phone)) {

System.out.println("이미 등록된 정보가 있습니다.");

} else {

map.put(name, phone);

System.out.println("정보가 추가되었습니다");

}

} else if (menu == 2) {

// 2. 삭제 -> 삭제하려는 이름이 존재하는 경우에만 삭제

System.out.print("삭제하고자 하는 이름 입력 : ");

name = sc.next();

if (map.containsKey(name)) {

map.remove(name);

System.out.println("정보가 삭제되었습니다");

} else {

System.out.println("삭제하려는 정보가 존재하지 않습니다");

}

} else if (menu == 3) {

// 3. 수정 -> 수정하려는 이름이 존재하는 경우에만 번호 수정

System.out.print("수정하고자 하는 이름 입력 : ");

name = sc.next();

System.out.print("수정된 전화번호 입력 : ");

phone = sc.next();

if (map.containsKey(name)) {

map.replace(name, phone);

System.out.println("정보가 수정되었습니다");

} else {

System.out.println("수정하려는 정보가 존재하지 않습니다");

}

} else if (menu == 4) {

// 4. 보기 -> entrySet() 사용

System.out.println("보기");

for (Map.Entry<String, String> item : map.entrySet()) {

System.out.printf("이름 : %s, 전화번호 : %s\n", item.getKey(), item.getValue());

}

**20회차(3)**

package test;

public class Mobile {

private String name;

private String telecom;

private int price;

Mobile(){};

Mobile(String name, String telecom, int price) {

this.name = name;

// 이렇게 작성하는 것도 가능하다

// this.set\_name(name);

this.telecom = telecom;

this.set\_price(price);

}

public String get\_name() {

return this.name;

}

public String get\_telecom() {

return this.telecom;

}

public int get\_price() {

return this.price;

}

private void set\_name(String name) {

this.name = name;

}

public void set\_telecom(String telecom) {

this.telecom = telecom;

}

public void set\_price(int price) {

if (price < 400000) {

System.out.println("휴대폰 최소가격은 400,000");

this.price = 400000; // 지역변수!

return;

}

this.price = price;

}

void info() {

System.out.printf("%s\t%s\t%d", this.name, this.telecom, this.price);

}

}

}

}

}

}

package test;

import java.util.\*;

public class Test03 {

public static void main(String[] args) {

// 이전 수업에서 mobile 클래스를 정의했었다

// 이제는 mobile 객체를 map을 통해서 관리

// 객체를 생성한 다음에 아래의 내용을 구현

// name telecom price 객체번호

// [1] 갤럭시8 SKT 300000 1

// [2] 아이폰13 LG 510000 2

// [3] G6 KT 330000 3

// 객체번호를 키로 하고, mobile 객체를 값으로 하는 map을 구현

// HashMap의 지네릭스는 다음과 같은 형태이다

HashMap<Integer, Mobile> map = new HashMap<Integer, Mobile>(); // -> 객체관리

Scanner sc = new Scanner(System.in);

map.put(1, new Mobile("갤럭시8", "SKT", 400000));

map.put(2, new Mobile("아이폰13", "LG", 510000));

map.put(3, new Mobile("G6", "KT", 430000));

// 출력

System.out.println("name\ttelecom\tprice\t객체번호");

for (Map.Entry<Integer, Mobile> item : map.entrySet()) {

// map의 value는 Mobile 객체이므로

// value를 가져와서 Mobile의 멤버메소드를 호출

item.getValue().info();

System.out.println("\t" + item.getKey());

}

}

}