

스낵류를 입력하세요.

종류 : 빵

이름 : 케이크

맛 : 블루베리

개수 : 1

가격 : 15000

저장 완료되었습니다.

저장한 정보를 확인하시겠습니까?(y/n) : y

빵(케이크 - 블루베리) 1개 15000원

또는

스낵류를 입력하세요.

종류 : 빵

이름 : 케이크

맛 : 블루베리

개수 : 1

가격 : 15000

저장 완료되었습니다.

저장한 정보를 확인하시겠습니까?(y/n) : n

## 2. 구현 클래스 다이어그램 (Class Diagram)

Snack
- kind : String // 종류
- name : String // 이름
- flavor : String // 맛
- numOf : int // 개수
- price : int // 가격
+ Snack()
+ Snack(kind:String, name:String, flavor:String, numOf:int, price:int)
+ information() : String
+ setter() / getter()

SnackMenu
- sc : Scanner = new Scanner(System.in)
- scr : SnackController = new SnackController
+ menu() : void

SnackController
- s : Snack = new Snack()
+ SnackController()
+ saveData(kind:String, name:String, flavor:String, numOf:int, price:int) : String
+ confirmData() : String

Run
+ <u>main(args:String[]) : void</u>

### 3. 구현 클래스 설명

Package명	Class명	Method	설명
com.kh.practice. snack.model.vo	Snack	+ Snack()	기본 생성자
		+ Snack(kind:String, name:String, flavor:String, numOf:int, price:int)	매개변수 있는 생성자
		+ getXXX()	저장된 데이터를 불러오는 메소드
		+ setXXX()	데이터를 변수에 저장하는 메소드
		+ information() : String	담긴 데이터를 리턴하는 메소드
com.kh.practice. snack.controller	SnackController	+ SnackController()	기본 생성자
		+ saveData(kind:String, name:String, flavor:String, numOf:int, price:int) : String	데이터를 setter를 이용해 저장하고 저장완료되었다는 결과를 반환하는 메소드
		+ confirmData() : String	저장된 데이터를 반환하는 메소드
com.kh.practice. snack.view	SnackMenu	+ menu():void	사용자가 Snack객체에 데이터를 저장할 수 있도록 값을 받고 저장한 값을 출력하는 메소드
com.kh.practice. snack.run	Run	<u>+ main(args:String[]):void</u>	SnackMenu 객체를 생성 후 menu() 실행

### 4. class 구조

```

public class Run{

    public static void main(String args[]) {

        // menu() 호출

    }

}

```