

2020-1

객체프로그래밍 실습

송인서

공과대학 IT미디어공학과 4학년(17)
songinseo0910@duksung.ac.kr
010-9610-9779

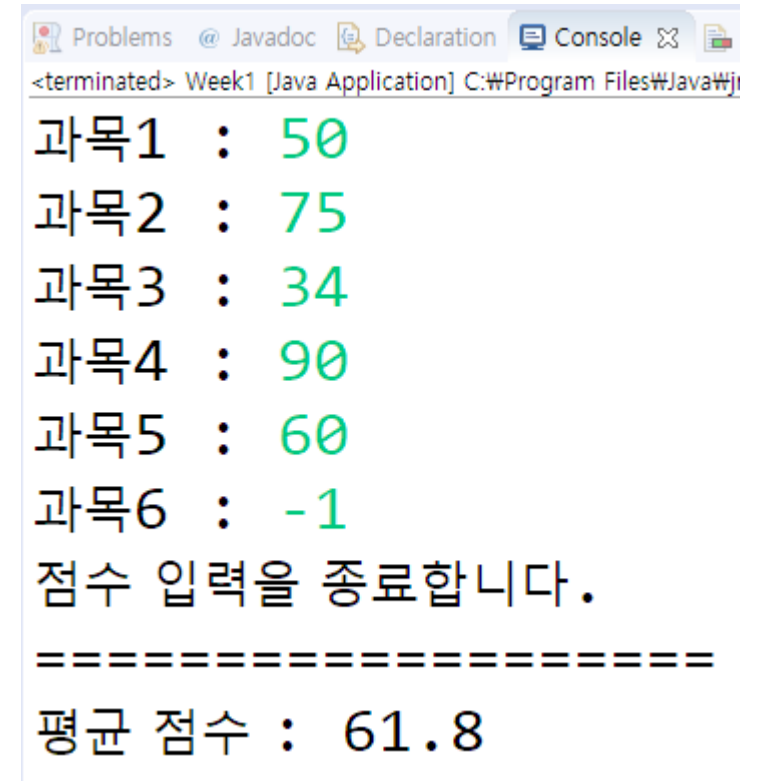
QnA 및 공지용  slack

2020-1-dswu-it-java.slack.com



예제 6 - 평균점수 구하기

1. 점수는 0~100의 정수
2. 입력은 한 줄에 하나씩 받음
3. 범위를 벗어나는 숫자가 들어오면 입력 종료
4. 입력이 종료되면 평균을 계산해 화면에 출력



```
Problems @ Javadoc Declaration Console
<terminated> Week1 [Java Application] C:\Program Files\Java\j
과목1 : 50
과목2 : 75
과목3 : 34
과목4 : 90
과목5 : 60
과목6 : -1
점수 입력을 종료합니다.
=====
평균 점수 : 61.8
```

예제 6 - 힌트

```
int count = 0;
int sum = 0;

while(true) {
    System.out.print("과목" + (count+1) + " : ");
    int score = sc.nextInt();

    if ( 입력된 값 score가 점수가 아니라면, (범위판정 필요) ) {
        System.out.println("점수 입력을 종료합니다.");
        break;
    }
    else {
        score가 유효한 값이라면 변수 count를 1 증가시키고,
        입력된 score를 sum에 더해야 함
    }
}

System.out.println("=====");
System.out.println("평균 점수 : " + 평균 점수 구하기 );
```

예제 6 - 답

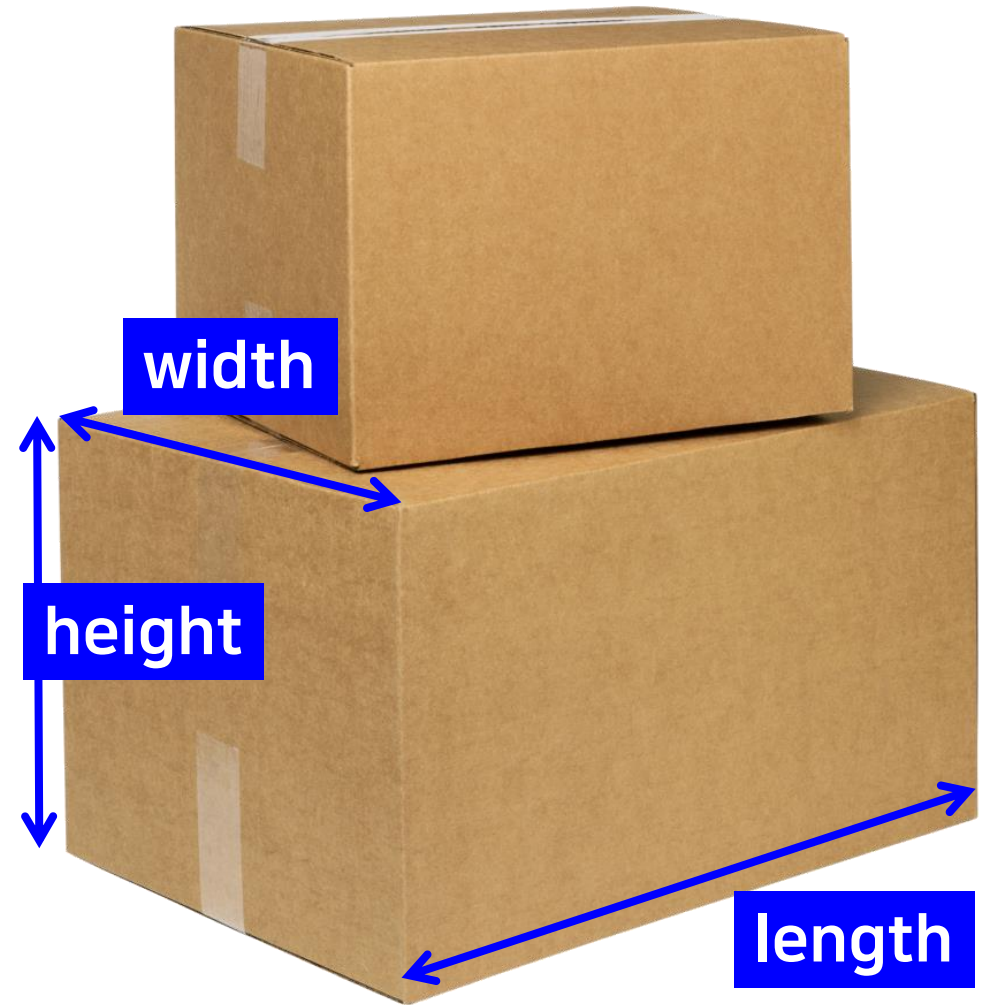
```
int count = 0;
int sum = 0;

while(true) {
    System.out.print("과목" + (count+1) + " : ");
    int score = sc.nextInt();

    if ( (score > 100) || (score < 0) ) {
        System.out.println("점수 입력을 종료합니다.");
        break;
    }
    else {
        sum += score;
        count++;
    }
}
System.out.println("=====");
System.out.println("평균 점수 : " + (float)sum/count );
```

예제 - Box

- 클래스 Box가 가져야 할 필드
 - 상자의 크기(가로, 세로, 높이)
 - 모든 상자의 수
- 클래스 Box가 가져야 할 메소드
 - 가로, 세로, 높이를 10으로 초기화하는 생성자
 - 가로, 세로, 높이를 인자로 받는 생성자
 - 상자의 부피를 구해 반환하는 메소드
 - 모든 상자의 수를 반환하는 메소드(힌트: static)

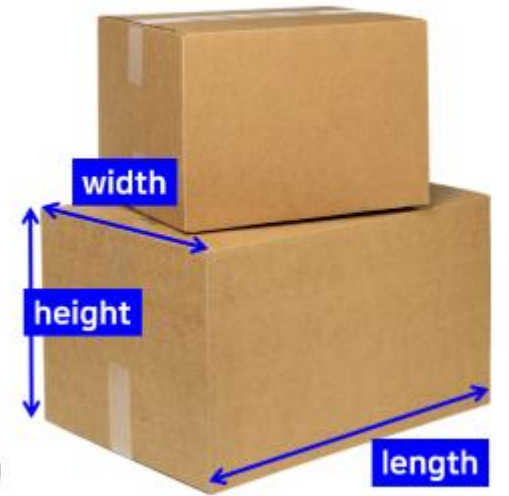


예제 - Box

```
public class Box {  
  
    static int numberOfBox;  
    int width;  
    int height;  
    int lenght;  
  
    Box(){  
        width = height = lenght = 10;  
        numberOfBox++;  
    }  
  
    Box(int width, int height, int lenght){  
        this.width = width;  
        this.height = height;  
        this.lenght = lenght;  
        numberOfBox++;  
    }  
  
    int getVolume() { return width*height*lenght; }  
    static int getNumeberOfBox() { return numberOfBox; }  
}
```

예제 - Box

- 클래스 Box가 가져야 할 필드
 - 상자의 크기(가로, 세로, 높이)
 - 모든 상자의 수
- 클래스 Box가 가져야 할 메소드
 - 가로, 세로, 높이를 10으로 초기화하는 생성자
 - 가로, 세로, 높이를 인자로 받는 생성자
 - 상자의 부피를 구해 반환하는 메소드
 - 모든 상자의 수를 반환하는 메소드(힌트: static)



this 와 this()

- 지난 시간에 배웠던 this – 현재 인스턴스를 가리키는 레퍼런스
- 오늘 배울 this() – 현재 클래스의 생성자를 의미, 즉 메소드
 - 생성자 오버로딩에 사용
 - 생성자 내부에서만 사용 가능
 - 반드시 생성자 블록의 첫 줄에 사용

예제 - Box에서 this() 사용해보기

```
class Box {
```

```
    static int numberOfBox;  
    int width;  
    int height;  
    int lenght;
```

```
Box(){  
    width = height = lenght = 10;  
    numberOfBox++;  
}  
  
Box(int width, int height, int lenght){  
    this.width = width;  
    this.height = height;  
    this.lenght = lenght;  
    numberOfBox++;  
}
```



```
Box(){  
    this(10, 10, 10);  
}  
  
Box(int width, int height, int lenght){  
    this.width = width;  
    this.height = height;  
    this.lenght = lenght;  
    numberOfBox++;  
}
```

```
int getVolume() { return width*height*lenght; }  
static int getNumeberOfBox() { return numberOfBox; }  
}
```


(참고) 디버거 사용법

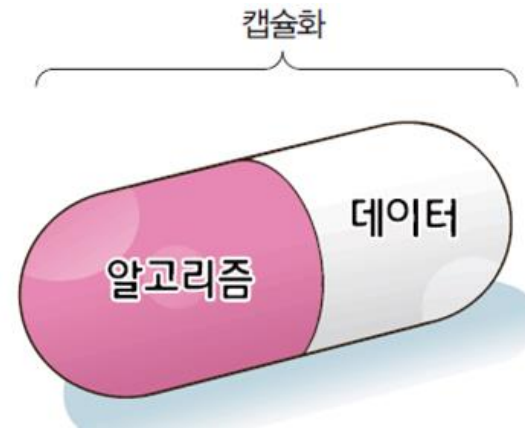
- <https://opentutorials.org/module/4294/26662>

입문자일수록 개념을 적게, 도구는 많이 사용해야 합니다. 중급으로 나아갈수록 지식의 양이 기하급수적으로 늘어납니다. 이때 스스로 모르는 것을 찾아내기 위해서는 도구가 필요합니다. 정말 유용한 도구인 디버거를 소개합니다.

캡슐화(Encapsulation)



사용자는 자판기 안에 돈이 얼마나 들어있는지,
음료수 재고는 어떻게 관리하는지,
혹은 자판기가 어떻게 동작하는지 알 필요 없이
그냥 자판기를 이용만 하면 됨

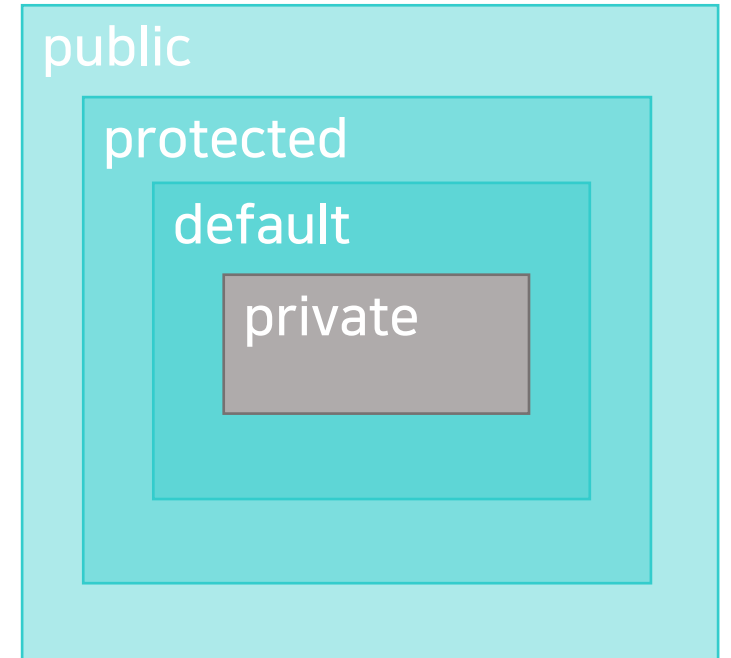


접근지정자

교재 167p

[표 7.1] 멤버 접근 지정자에 대한 액세스 제어

	해당클래스	해당패키지	해당클래스의 서브클래스	기타 (다른 패키지)
public	O	O	O	O
protected	O	O	O	X
*default	O	O	X	X
private	O	X	X	X



Enum(열거형)

교재 134p 6.4 Enum 타입

- 레퍼런스 타입 객체
- 상수들의 집합을 관리할 때 사용
 - 요일, 영어 달 이름, 계절, 메뉴, 태양계 행성 등...
- 상수 키워드에 어떤 값이나 논리를 부여-> 상수를 클래스처럼 사용할 수 있게 해줌
 - Java의 enum은 기능이 매우 풍부한 편
 - 적절하게 잘 쓰면 코드의 가독성도 올라가고 유지보수도 쉬워짐

예제 - Planet

- 교재 137~138p 참조하여 파일을 작성해봅시다.(파일명 Planet.java로 생성)

	//mass	//radius	
<i>MERCURY</i>	(3.303e+23,	2.4397e6),
<i>VENUS</i>	(4.869e+24,	6.0518e6),
<i>EARTH</i>	(5.976e+24,	6.37814e6),
<i>MARS</i>	(6.421e+23,	3.397e6),
<i>JUPITER</i>	(1.9e+27,	7.1492e7),
<i>SATURN</i>	(5.688e+26,	6.0268e7),
<i>URANUS</i>	(8.686e+25,	2.5559e7),
<i>NEPTUNE</i>	(1.024e+26,	2.4746e7);

public static final double *G* = 6.67300e-11;

```

double surfaceWeight(double otherMass) {
    return otherMass * surfaceGravity();
}

public static void main(String[] args) {
    if (args.length != 1) {
        System.err.println("Usage: java Planet <earth_weight>");
        System.exit(-1);
    }
    double earthWeight = Double.parseDouble(args[1]);
    double mass = earthWeight/EARTH.surfaceGravity();
    for (Planet p : Planet.values())
        System.out.printf("Your weight on %s is %f\n", p,
            p.surfaceWeight(mass));
    }
}

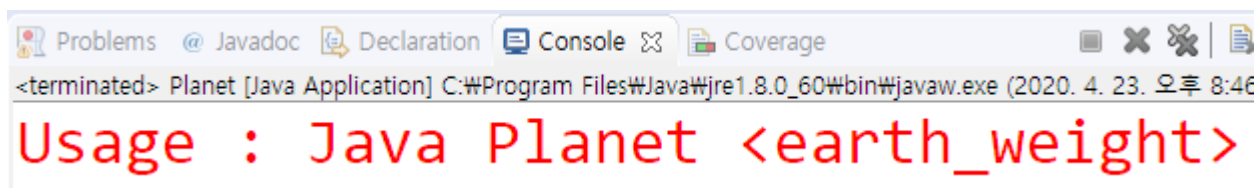
```

args[1]을 args[0]으로 수정

만약에 175kg을 인수로 가지는 명령문으로 Planet.class를 실행하며 다음과 같은 결과를 얻을 것이다.

예제 - Planet

- 평소처럼 실행해봤지만 에러가 났음



```
if(args.length != 1) {  
    System.err.println("Usage : Java Planet <earth_weight>");  
    System.exit(-1);  
}
```