

Lab 2분반

2022-05-09

담당교수: 박경신 교수님

과목: 자바프로그래밍1

소속: 상경대학 경영학과

학번: 32160707

이름: 김성현

1. Lab3.java소스코드

```
//Java22_Lab3_2분반_32160707_김성현

import java.util.Scanner;

/* 본인의 메소드와 클래스 작성*/

//getUserInput 클래스: BMI/BFP 선택메서드와 중단 선택메서드를 포함.
class UserInput {

    static Scanner scan = new Scanner(System.in);

    // 계산법 선택 1번, 2번, 3번, 4번을 반환하는 메서드:
    public static int getChoiceKey() {

        System.out.print("***** 연산메뉴 *****BMI측정을 원하시면 1번키를,
BFP측정을 원하시면 2번키를 입력하세요.: \n");

        System.out.print("1. BMI\n");
        System.out.print("2. BFP \n");
        System.out.print("3. BMR \n");
        System.out.print("4. CI\n");

        String num = scan.nextLine();
        if (num.contentEquals("1"))
            return 1;
        else if (num.contentEquals("2"))
            return 2;
        else if (num.contentEquals("3"))
            return 3;
        else if (num.contentEquals("4"))
            return 4;
        else
```

```

        System.out.println("다시 수를 입력해주세요");
    }
    return 0;
}

// 중단 메서드
public static boolean getExitKey() {
    System.out.print("Press q-key to exit or enter-key to continue: ");
    String s = scan.nextLine();
    if (s.contentEquals("q"))
        return true;
    else
        return false;
}

```

```

public class Lab3 {

```

```

    public static void main(String[] args) {

```

//1. foreach문을 이용해서 모든 data를 사용해서 모든 모드(BMI, BFP, BMR, CI)를 계산

```

//static void calculate(Mode mode, PersonData person)

```

```

class calculateAll { //1번은 구현X

```

```

    static void calculate(Mode mode, PersonData person) {

```

```

        //이중 for문? mode(0부터 3까지 4개의 계산)에 따라 각

```

```

        PersonData

```

//foreach구문을 제대로 이해못한거 같다.

```
for (Mode i : mode.values()){  
    System.out.println(i);  
    for (PersonData j : PersonData.data){  
        System.out.println(j);  
    }  
}  
}  
  
}
```

```
calculateAll all = new calculateAll();
```

```
System.out.println("1번은 구현하지 못했습니다.");
```

//2. 유저입력값에 따라 계산

//계산법 초기값을 설정. 원래 초기값이 false라 의미없는 코드인가

```
boolean exitChoice = false;
```

/* 계산방법 선택 메서드와 작업중지 선택메서드를 사용하기 위해 인스

턴스 생성 */

```
UserInput info = new UserInput();
```

/*작업중지 선택기능을 위해 전체를 do-while로 감쌌다. */

```
do {
```

```
    //계산방법 선택 메서드. Lab2에서는 이를 boolean값으로
```

줬으나, 계산방법이 많아져 int로 주게됨.

```
    int choice = info.getChoiceKey();
```

```
    //1일 경우 BMI 계산
```

```
    if (choice == 1){
```

```
        BMICalculator bmi1 = new BMICalculator();
```

```
        bmi1.getUserInput();
```

```
        System.out.println(bmi1.toString());
```

```

    } else if (choice == 2) {
        BFPCalculator bfp1 = new BFPCalculator();
        bfp1.getUserInput();
        //성별이 남자일 경우, setter로 값을 변경
        bfp1.setGender(Gender.MALE);
        System.out.println(bfp1.toString());

    } else if (choice == 3) {
        BMRCalculator bfp1 = new BMRCalculator();
        bfp1.getUserInput();
        //성별이 남자일 경우, setter로 값을 변경
        bfp1.setGender(Gender.MALE);
        System.out.println(bfp1.toString());

    } else if (choice == 4) {
        CalorieIntakeCalculator bfp1 = new
CalorieIntakeCalculator();

        bfp1.getUserInput();
        //성별이 남자일 경우, setter로 값을 변경
        bfp1.setGender(Gender.MALE);
        System.out.println(bfp1.toString());
    } else
        break;

    //main함수 내 연산을 계속 반복할 것인지에 대한 continue
구문.

    //조건식처럼 false일 경우 작업을 반복하고, 만약 값이 true라
면 반복구문을 빠져나와 시스템을 종료.

    exitChoice = info.getExitKey();

```

```

        if (exitChoice == true)
            continue;
    } while (exitChoice == false);

    System.out.println("시스템을 종료합니다.");
}
}
}

```

1-2. HealthCalculator.java소스코드

//추상클래스 작성-변수와 메서드중 중복되는 부분 고려.

```

public abstract class HealthCalculator {
    //변수가 필요한가?
    protected static double weight;
    protected static double height;
    protected static Gender gender;
    //생성자를 따로 작성해야하나?
    //어떤 계산방법을 사용할 것인가.
    public abstract double calculate();
    //사용자 정보 받아오기.
    public abstract void getUserInput();
}

```

1-3. BMICalculator.java소스코드

```

import java.util.Scanner;

public class BMICalculator extends HealthCalculator {
    // getter/setter
    public double getWeight() {

```

```

        return BMICalculator.weight;
    }

    public void setWeight(double weight) {
        if (weight <= 0 || weight > 150)
            return; // 무게(kg) 값의 범위 지정
        BMICalculator.weight = weight;
    }

    public double getHeight() {
        return BMICalculator.height;
    }

    public void setHeight(double height) {
        if (height <= 0 || height > 200)
            return; // 키(cm) 값의 범위 지정
        BMICalculator.height = height;
    }

    // public String toString() 메서드: BMI출력
    @Override
    public String toString() {
        return "당신의 BMI지수는 " + calculate() + " 입니다.";
    }

    @Override
    public double calculate() {
        return weight / (height * 0.01) / (height * 0.01);
    }
}

```

```

@Override
public void getUserInput() {
    Scanner scanner = new Scanner(System.in);
    System.out.println("몸무게를 입력해주세요. ");
    weight = scanner.nextDouble();
    System.out.println("키를 입력해주세요.");
    height = scanner.nextDouble();

}

}

```

1-4. BFPCalculator.java소스코드

```

import java.util.Scanner;

public class BFPCalculator extends HealthCalculator {
    //상속받은 변수 외에 필요한 변수 선언
    double neck;
    double waist;
    double hip;

    // BFPCalculator클래스의 멤버필드는 매개변수가 있는 constructor로 선언함.

    public void BFPCalculator(Gender gender, double height, double neck, double waist, double hip) {

        this.gender = gender;
    }
}

```



```

        this.height = height;
        this.neck = neck;
        this.waist = waist;
        this.hip = hip;
    }

    // getter/setter
    public Gender getGender() {
        return gender;
    }

```

```

    public void setGender(Gender gender) {
        //클래스 멤버인 boolean형의 Gender값을 조건으로 변경하도록 했

```

다.

```

        if (Gender == false)
            return;
        this.gender = gender;
    }

```

```

    public double getHeight() {
        return height;
    }

```

```

    public void setHeight(double height) {
        this.height = height;
    }

```

```

    public double getNeck() {
        return neck;
    }

```

```

public void setNeck(double neck){
    this.neck = neck;
}

public double getWaist(){
    return waist;
}

public void setWaist(double waist){
    this.waist = waist;
}

public double getHip(){
    return hip;
}

public void setHip(double hip){
    this.hip = hip;
}

//public String toString() 메서드: BFP 출력
@Override
public String toString(){
    return "당신 성별은" + gender + "이고, 이에 따른 BFP지수는" +
calculate() + " 입니다.";
}

/* 본인의 메소드와 클래스 작성*/
// 성별을 가를 수 있는 클래스 메서드 작성

```

```
// 입력한 성별에 따라 논리값을 반환한다.
// 멤버필드 입력 전, 성별을 먼저 입력받는다.
// Gender메서드를 호출해 클래스변수인 boolean Gender에 값을 저장한다.
```

```
static Scanner scan = new Scanner(System.in);

public static boolean Gender() {
    System.out.println("성별이 남자면 1을, 여자면 2를 입력해주세요.
");

    String gen = scan.nextLine();
    if (gen.contentEquals("1"))
        return true;
    else
        return false;
}

static boolean Gender = Gender();

//.scan이 널값이라는 예러가 났었는데, gender변수에 입력받지 않은 스캐너
값을 먼저 대입하려고 하니, null에러가 뜬 것. 코드 순서를 바꿔줌
```

```
@Override
// calculateBFP 메서드, 멤버필드로 계산
public double calculate() {
    if (Gender == true) {
        return 495 / (1.29579 - .35004 * Math.log10(waist - neck) +
0.22100 * Math.log10(height)) - 450;
    } else {
        return 495 / (1.29579 - .35004 * Math.log10(waist + hip -
neck) + 0.22100 * Math.log10(height)) - 450;
    }
}
}
```

```

@Override

// public void getUserInputBFP(): 사용자 입력값을 멤버필드로 쓸 수 있는 메서드
// 입력한 성별에 따라 논리값을 조건으로 입력을 다르게 했다.

public void getUserInput() {

    Scanner scanner = new Scanner(System.in);

    System.out.println("키(cm)를 입력해주세요.");
    height = scanner.nextDouble();

    System.out.println("목(cm)를 입력해주세요.");
    neck = scanner.nextDouble();

    System.out.println("허리(cm)를 입력해주세요. ");
    waist = scanner.nextDouble();

    if (Gender != true) {

        System.out.println("엉덩이(cm)를 입력해주세요.");
        hip = scanner.nextDouble();

    }

}

}

```

1-5. BMRCalculator.java소스코드

```

import java.util.Scanner;

public class BMRCalculator extends HealthCalculator {

    protected static int age;

    public BMRCalculator() { //매개변수를 따로 받지 않아도 되는구나!

```

```
this.gender = gender;  
this.height = height;  
this.weight = weight;  
this.age = age;  
}
```

// getter/setter

```
public static int getAge(){  
    return age;  
}
```

```
public static void setAge(int age){  
    BMRCalculator.age = age;  
}
```

```
public Gender getGender(){  
    return gender;  
}
```

```
public void setGender(Gender gender){  
    //클래스 멤버인 boolean형의 Gender값을 조건으로 변경하도록 했다.  
    if (Gender == false)  
        return;  
    this.gender = gender;  
}  
  
public double getWeight(){  
    return BMRCalculator.weight;  
}
```

```

public void setWeight(double weight){
    if (weight <= 0 || weight > 150)
        return; // 무게(kg) 값의 범위 지정
    BMRCalculator.weight = weight;
}

```

```

public double getHeight(){
    return BMRCalculator.height;
}

```

```

public void setHeight(double height){
    if (height <= 0 || height > 200)
        return; // 키(cm) 값의 범위 지정
    BMRCalculator.height = height;
}

```

//성별 선택에 대한 클래스. (이건 BMR, BFP 중복코드니까 따로 클래스로 두면 간결해질 것이다.)

```

static Scanner scan = new Scanner(System.in);

public static boolean Gender(){
    System.out.println("성별이 남자면 1을, 여자면 2를 입력해주세요. ");
    String gen = scan.nextLine();
    if (gen.contentEquals("1"))
        return true;
    else
        return false;
}

```

// public String toString() 메서드: BMR출력

```

static boolean Gender = Gender();

@Override

public String toString(){

    return "당신 성별은" + gender + "이고, 이에 따른 BMR지수는" + calculate()
+ " 입니다.";

}

```

```

@Override

public double calculate(){

    if (Gender == true){

        return 66.4730 + (13.7516 * weight) + (5.0033 * height) - (6.7550 *
age);

    } else {

        return 655.0955 + (9.5634 * weight) + (1.8496 * height) - (4.6756 *
age);

    }

}

```

```

@Override

public void getUserInput(){

    Scanner scanner = new Scanner(System.in);

    System.out.println("몸무게를 입력해주세요. ");

    weight = scanner.nextDouble();

    System.out.println("나이를 입력해주세요.");

    age = scanner.nextInt();

}

```

```
}
```

1-6. CalorieIntakeCalculator.java소스코드

```
import java.util.Scanner;

public class CalorieIntakeCalculator extends BMRCalculator {
    protected static Activity activity;

    public CalorieIntakeCalculator() { //매개변수를 따로 받지 않아도 되는구나!
        this.gender = gender;
        this.height = height;
        this.weight = weight;
        this.age = age;
    }

    // getter/setter

    public static int getAge() {
        return age;
    }

    public static void setAge(int age) {
        BMRCalculator.age = age;
    }

    public Gender getGender() {
        return gender;
    }
}
```



```

public void setGender(Gender gender){
    //클래스 멤버인 boolean형의 Gender값을 조건으로 변경하도록 했다.
    if (Gender == false)
        return;
    this.gender = gender;
}

public double getWeight(){
    return BMRCalculator.weight;
}

public void setWeight(double weight){
    if (weight <= 0 || weight > 150)
        return; // 무게(kg) 값의 범위 지정
    BMRCalculator.weight = weight;
}

public double getHeight(){
    return BMRCalculator.height;
}

public void setHeight(double height){
    if (height <= 0 || height > 200)
        return; // 키(cm) 값의 범위 지정
    BMRCalculator.height = height;
}

@Override
public String toString(){

```

```

        return "당신 성별은" + gender + "이고, 이에 따른 BMR지수는" + calculate()
+ " 입니다.\n" +
        "*****CalorieIntakeCalculator*****"+
        "SEDENTARY: " + calculate() * 1.2 + "\n" +
        "LIGHTLY_ACTIVE: " + calculate() * 1.375 + "\n" +
        "MODERATELY_ACTIVE: " + calculate() * 1.55 + "\n" +
        "VERY_ACTIVE: " + calculate() * 1.725 + "\n" +
        "EXTRA_ACTIVE: " + calculate() * 1.9;

```

//enum을 선언하고도 쓰지 않아서, 코드가 길어진다.

```

@Override
public double calculate() {

    if (Gender == true){
        return 66.4730 + (13.7516 * weight) + (5.0033 * height) - (6.7550 *
age);
    } else {
        return 655.0955 + (9.5634 * weight) + (1.8496 * height) - (4.6756 *
age);
    }
}

```

```

@Override
public void getUserInput() {
    Scanner scanner = new Scanner(System.in);
    System.out.println("몸무게를 입력해주세요. ");
    weight = scanner.nextDouble();
    System.out.println("나이를 입력해주세요.");
    age = scanner.nextInt();
}

```

```
}
```

```
}
```

1-7. Mode.java소스코드

```
public enum Mode {  
    BMI, BFP, BMR, CI  
}  
//키값(1~4번을 넣어서 쓸 수 있을까?)
```

1-8. Mode.java소스코드

```
public enum Gender {  
    MALE, FEMALE;  
  
}
```

1-9. Activity.java소스코드

```
public enum Activity {  
    SEDENTARY, LIGHTLY_ACTIVE, MODERATELY_ACTIVE, VERY_ACTIVE,  
    EXTRA_ACTIVE  
}  
//enum요소들을 각각 계산식을 넣어서 정의하면 출력부분에서 식이 길어지지 않을텐  
데. 구현하지 못함.
```

1-10. PersonData.java소스코드

```

public record PersonData(
    int age,
    Gender gender,
    double weight,
    double height,
    double neck,
    double waist,
    double hip,
    Activity activity
){
    static PersonData[] data = {
        new PersonData(23, Gender.FEMALE, 54.25, 167.75, 36.2,
85.2, 94.5, Activity.EXTRA_ACTIVE),
        new PersonData(74, Gender.MALE, 94.1, 177.8, 40.8, 108.5,
107.1, Activity.LIGHTLY_ACTIVE),
        new PersonData(34, Gender.FEMALE, 63.49, 169.07, 36.0,
83.4, 89.6, Activity.VERY_ACTIVE),
        new PersonData(54, Gender.MALE, 94.1, 177.8, 40.8, 108.5,
107.1, Activity.MODERATELY_ACTIVE),
        new PersonData(35, Gender.MALE, 57.03, 166.37, 34.0,
75.0, 89.2, Activity.EXTRA_ACTIVE),
        new PersonData(35, Gender.MALE, 80.38, 180.34, 38.4,
90.3, 98.7, Activity.MODERATELY_ACTIVE),
        new PersonData(56, Gender.FEMALE, 59.93, 161.61, 38.7,
90.3, 99.9, Activity.MODERATELY_ACTIVE),
        new PersonData(47, Gender.MALE, 102.83, 182.25, 41.5,
108.8, 114.4, Activity.SEDENTARY),
        new PersonData(69, Gender.FEMALE, 63.72, 165.72, 35.8,
79.4, 89.0, Activity.LIGHTLY_ACTIVE),
    }
}

```

```

        new PersonData(55, Gender.MALE, 90.03, 188.59, 38.3,
96.7, 106.6, Activity.LIGHTLY_ACTIVE)
    };
}

```

2. 결과창

1번 레코드 정보를 모든 모드에 따라 계산결과 출력X

2번 키값을 입력하면 각 모드에 따라 사용자 입력과 계산결과가 나오도록 구현.

```

***** 연산메뉴 *****BMI측정을 원하시면 1번키를, BFP측정을 원하시면 2번키를 입력하세요.:
1. BMI
2. BFP
3. BMR
4. CI
1
몸무게를 입력해주세요.
70
키를 입력해주세요.
170
당신의 BMI지수는 24.221453287197235 입니다.
Press q-key to exit or enter-key to continue:
***** 연산메뉴 *****BMI측정을 원하시면 1번키를, BFP측정을 원하시면 2번키를 입력하세요.:
1. BMI
2. BFP
3. BMR
4. CI
2
성별이 남자면 1을, 여자면 2를 입력해주세요.
1
키(cm)를 입력해주세요.
170
목(cm)를 입력해주세요.
30
허리(cm)를 입력해주세요.
33
당신 성별은MALE이고, 이에 따른 BFP지수는-144.76619998904687 입니다.
Press q-key to exit or enter-key to continue:
Control  찾기 실행 디버그 TODO 문제점 터미널 SonarLint 서비스
25ms에서 성공적으로 완료되었습니다 (1분 전) 3 hrs 34 mins

```

```

2
성별이 남자면 1을, 여자면 2를 입력해주세요.
1
키(cm)를 입력해주세요.
170
목(cm)를 입력해주세요.
30
허리(cm)를 입력해주세요.
33
당신 성별은MALE이고, 이에 따른 BFP지수는-144.76619998904687 입니다.
Press q-key to exit or enter-key to continue:
***** 연산메뉴 *****BMI측정을 원하시면 1번키를, BFP측정을 원하시면 2번키를 입력하세요.:
1. BMI
2. BFP
3. BMR
4. CI
3
성별이 남자면 1을, 여자면 2를 입력해주세요.
1
몸무게를 입력해주세요.
70
나이를 입력해주세요.
20
당신 성별은MALE이고, 이에 따른 BMR지수는1744.5460000000003 입니다.
Press q-key to exit or enter-key to continue: q
시스템을 종료합니다.

종료 코드 0(으)로 완료된 프로세스
Control  찾기 실행 디버그 TODO 문제점 터미널 SonarLint 서비스

```