

WEEK02. 자바 복습

자료구조 실습

조교 소개

- ▶ 민진우
- ▶ 컴퓨터공학과 대학원
- ▶ 생물정보학 연구실(A1304)
- ▶ 이메일 : mzinoo@hallym.ac.kr

과제 제출

- ▶ 과제를 제출할때 생성한 자바 프로젝트 폴더를 압축하여 제출합니다.

.class 파일, 혹은 .java 파일만 제출하면 안됩니다.

아래 이미지와 같이 자바 프로젝트 생성시 지정한

폴더 전체를 압축하여 제출하세요.

또, 폴더 이름을 "학번_이름" 으로 수정한 후 압축해서

- ▶ **제출하세요.**의 이름은 본인의 학번_이름 으로 제출해주세요. 해당 제출 양식을 반드시 지켜주세요!

- ▶ 예시) 20171234_갱플.zip



객체; 문제01

- ▶ 해당 코드를 작성 및 실행하여 결과값을 확인 해볼것.

```
class Fruit {  
    int apple = 5;    // 객체 참조 변수  
    int straw = 10;  
    int grapes = 15;  
}  
  
class Buy {  
    public static void main(String[] args) {  
        int quantity1, quantity2;  
        Fruit f1 = new Fruit();  
        Fruit f2 = f1;  
        quantity1 = f1.apple + f1.straw + f1.grapes;  
        quantity2 = f2.apple + f2.straw + f2.grapes;  
  
        System.out.println("객체 f1의 초기 과일 개수 "+quantity1+"개");  
        System.out.println("객체 f2의 초기 과일 개수 "+quantity2+"개");  
  
        f1.apple = 10;  
        f2.straw = 20;  
        f1.grapes = 30;  
        quantity1 = f1.apple + f1.straw + f1.grapes;  
        quantity2 = f2.apple + f2.straw + f2.grapes;  
  
        System.out.println("객체 f1의 값 변동 후 개수 "+quantity1+"개");  
        System.out.println("객체 f2의 값 변동 후 개수 "+quantity2+"개");  
    }  
}
```

클래스

객체 생성

메소드

- ▶ 메소드 선언부에 지정된 반환 타입과 실제 반환하는 데이터 타입이 일치해야 한다.

```
public static void add(int i) {  
}  
  
public static int add(int i, int j) {  
    return i+j;  
}  
  
public static double divide(int a, double b) {  
    return a/b;  
}  
  
public static String text(String str) {  
    return str + "result";  
}  
  
public static boolean bool(Boolean i) {  
    return true;  
}
```

메소드; 문제02

- ▶ 해당 코드를 직접 작성한 후 실행하여 결과값을 확인 해볼것.

클래스

```
class ParaPassing1 {  
    public void change(int i, int j[]) {  
        i = 20;  
        j[3] = 400;  
    }  
  
    public void display(int i, int j[]) {  
        System.out.println("객체 속성 변수 i의 값 : " + i);  
        System.out.print("배열의 값 : ");  
        for(int index = 0; index < j.length; index++)  
            System.out.print(j[index] + " ");  
        System.out.println("");  
    }  
}  
  
class ParaPassingDemo1 {  
    public static void main(String args[]) {  
        ParaPassing1 cb = new ParaPassing1();  
        int i = 10;  
        int j[] = { 1, 2, 3, 4 };  
        System.out.println("첫번째 display() 메소드 호출");  
        cb.display(i, j);  
        cb.change(i, j);  
        System.out.println("=====");  
        System.out.println("값을 변환한 다음 두번째 display() 호출");  
        cb.display(i, j); // 기본 자료형 값은 변하지 않는다. 참조 자료형은 변함.  
    }  
}
```

메소드

메소드

객체 생성

배열

- ▶ 자바에서 배열은 객체이므로 반드시 new 연산자를 사용하여 생성.

```
int[] value = new int[10];
```

이런식으로 배열을
선언과 동시에 초기화 할 수 있다.

```
class tmp {  
    public static void main(String[] args) {  
        int[] value = {4, 2, 1, 7, 8, 9, 5};  
        int a=0;  
  
        System.out.println("배열 정렬 전");  
        while(a != value.length) {  
            System.out.print(value[a]+" ");  
            a++;  
        }  
  
        System.out.println("\n배열 정렬 후");  
        bubble(value);  
    }  
}
```

배열; 문제03

▶ Call by Value

```
public class tmp02 {  
    public static void main(String[] args) {  
        int a = 5;  
  
        add(a);  
        System.out.println("Main : " + a);  
    }  
  
    public static void add(int a) {  
        a = a+10;  
        System.out.println("Method : " + a);  
    }  
}
```

▶ Call by Reference

```
public class tmp02 {  
    public static void main(String[] args) {  
        int[] a = {10, 20, 30, 40, 50};  
  
        add(a);  
        for(int i=0; i<5; i++)  
            System.out.println("Main : a[" + i + "] = " + a[i]);  
    }  
  
    public static void add(int[] a) {  
        a[2] = a[2]+100;  
        for(int i=0; i<5; i++)  
            System.out.println("Method : a[" + i + "] = " + a[i]);  
    }  
}
```

- ▶ 위 두 코드를 각각 작성하고 실행하여 결과를 비교해보세요.
기본 자료형과 참조 자료형을 전달할 때의 차이점을 설명하세요.
참조 자료형 코드 작성 후 주석 처리해서 작성.
- ▶ 스캠에서 학습자료실의 '자바 복습' 게시글의 피피티 자료, 앞선 문제 참고.

배열; 문제04

- ▶ 생성된 배열의 원소를 정렬하는 메서드를 생성.
- ▶ 해당 문제에서는 '버블 정렬'을 이용하여 원소를 오름차순으로 정렬.
- ▶ 첫 원소부터 인접한 원소와 비교하여 오름차순 순서가 맞지 않으면 서로 교환한다.
 - ▶ a[0]과 a[1]을 비교했을때, a[0]이 더 크면 오름차순 정렬에 맞지 않기 때문에 두 원소를 서로 교환한다.
- ▶ 위 코드에서 이중 for문 안에 빈칸을 작성.
해당 정렬 방법 외 다른 방법으로 정렬해도 무방함.

```
class tmp {  
    public static void main(String[] args) {  
  
        int[] value = {4, 2, 1, 7, 8, 9, 5};  
        int a=0;  
  
        System.out.println("배열 정렬 전");  
        while(a != value.length) {  
            System.out.print(value[a]+" ");  
            a++;  
        }  
  
        System.out.println("\n배열 정렬 후");  
        bubble(value);  
    }  
  
    public static void bubble(int[] value) {  
        for(int j=value.length-1; j>=0; j--) {  
            for(int i=0; i<j; i++) {  
  
            }  
        }  
        int a=0;  
        while(a != value.length) {  
            System.out.print(value[a]+" ");  
            a++;  
        }  
    }  
}
```

문제 04

과제

- ▶ 1번, 2번 문제는 교수님께서 올려주신 코드입니다. 한번 직접 작성하여 객체와 메소드의 사용 방법을 이해하면 됩니다. 코드를 작성하여 함께 제출하세요.
- ▶ 3번 문제는 1, 2번 문제의 코드와 교수님께서 올려주신 학습자료실의 ppt자료를 공부하여 이해한 내용을 바탕으로 문제를 해결하시면 됩니다. 내용은 주석으로 작성하여 제출하시면 됩니다. 참조 자료형 코드 하단부에 작성해 주세요.
- ▶ 4번 문제는 배열의 원소를 정렬하는 문제입니다. 굳이 버블정렬로 할 필요 없습니다. 본인이 생각해볼 수 있는 정렬 방법으로 정렬해도 됩니다. 다만 완전한 오름차순 정렬이 되어야 합니다.