

[문제 1] 다음과 같은 조건을 만족하는 프로그램을 작성하시오

2차원 배열에 들어 있는 데이터들의 합계와 평균을 구한다.

1. 사용 데이터

```
int [][] array = {  
    {12, 41, 36, 56, 21},  
    {82, 10, 12, 61, 45},  
    {14, 16, 18, 78, 65},  
    {45, 26, 72, 23, 34}  
};
```

2. 구현 클래스

Package명	Class명	Method	설명
com.elite.test01	ArrayTest	<u>+main(args:String[]):void</u>	main 함수 안에서 모든 코드 작업 진행

* class 명과 method 명은 변경하지 않는다.

3. 실행 결과

출력 결과는 다음과 같다.

합계 : 767.0
평균 : 38.35

합계와 평균 값은 double로 처리한다.

[문제 2] 다음과 같은 조건을 만족하는 프로그램을 작성하시오

Human 클래스를 상속받은 Student 클래스를 이용하여 프로그램 한다.

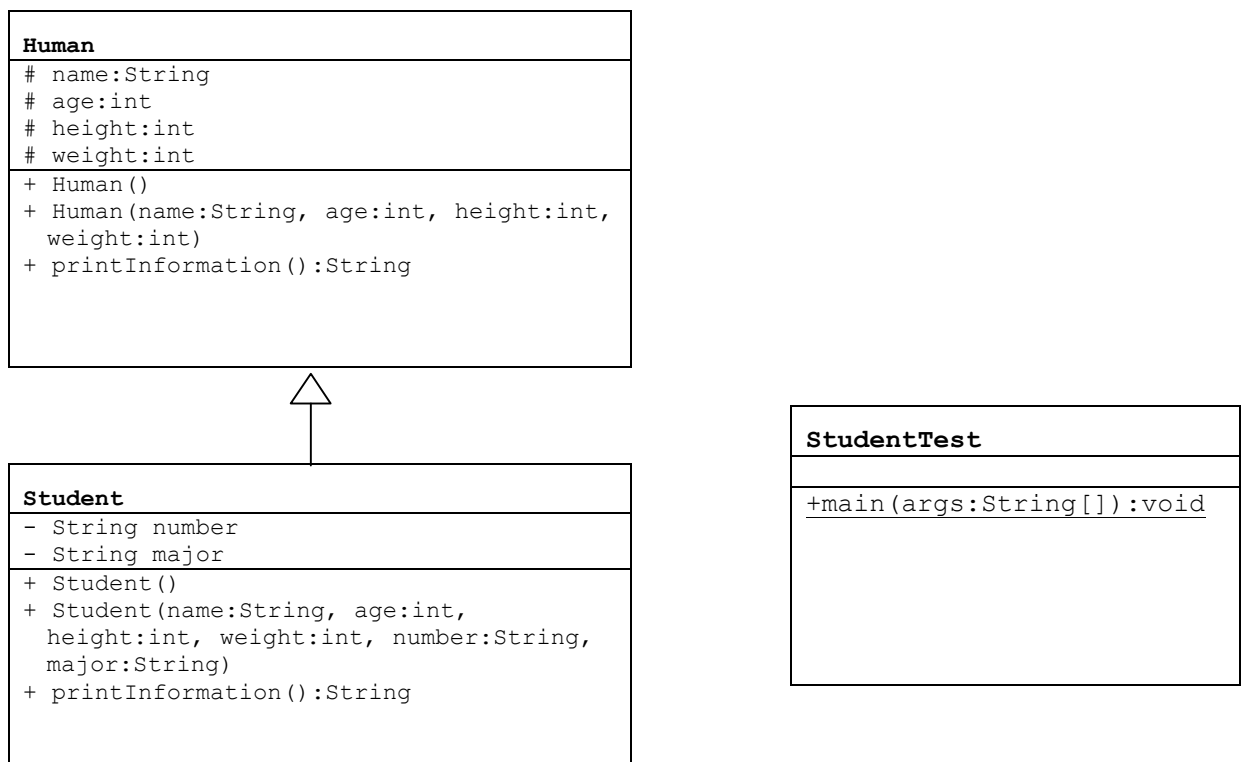
3개의 Student 객체를 생성하여 배열에 저장한 후, 각 객체의 모든 정보를 출력한다.

1. 사용 데이터

아래와 같이 3개의 Student Object를 생성하여 프로그램을 동작 시킨다.

name	나이	신장	몸무게	학번	전공
홍길동	15	171	81	201101	영문
한사람	13	183	72	201102	건축
임걱정	16	175	65	201103	무용

2. 구현 클래스 다이어그램



* 모든 클래스 변수의 getter, setter 함수는 직접 구현한다.

* “#” 는 protected

3. 구현 클래스

Package명	Class명	method	설명
com.elite.test02	Human	+ Human()	기본 생성자
		+ Human(name:String, age:int, height:int, weight:int)	4개의 클래스 변수를 받는 생성자
		+ printInformation():String	Human 정보를 리턴한다.
	Student	+ Student()	기본 생성자
		+ Student(name:String, age:int, height:int, weight:int, number:String, major:String)	6개의 클래스 변수를 받는 생성자
		+ printInformation():String	Student 정보를 리턴 한다.
	StudentTest	+ <u>main(String args[]): void</u>	main 함수 안에서 Student 타입의 배열을 선언하여 동작시킨다.

- * 클래스 명과 method 명은 변경하지 않는다.
- * 위에 선언한 클래스 변수와 클래스 함수만을 이용한다.
- * 클래스 변수의 getter, setter 함수는 모두 구현한다.

4. StudentTest 클래스 구조

Student 객체를 담을 수 있는 배열을 선언하고, 3개의 Student 객체를 생성하여 담는다.

```
public class StudentTest {
    public static void main(String args[]) {

        Student arrays [] = new Student[3];
        // Student 객체를 3개 생성하여 배열에 넣는다.
        // 배열에 있는 객체 정보를 모두 출력한다. - for 문을 이용 할 것

    }
}
```

5. 실행 결과

홍길동	15	171	81	201101	영문
한사람	13	183	72	201102	건축
임걱정	16	175	65	201103	무용

[문제 3] 다음과 같은 조건을 만족하는 프로그램을 작성하시오

주어진 String 데이터를 ","로 나누어 5개의 실수 데이터를 추출하고 합과 평균을 구한다.
단, String 문자열의 모든 실수 데이터를 배열로 만들어 계산한다.

1. 사용 데이터

String str = "1.22,4.12,5.93,8.71,9.34";

2. 구현 클래스

Package명	Class명	Method	설명
com.elite.test03	StringTest	<u>+main(args:String[]):void</u>	main 함수 안에서 모든 코드 작업 진행

* class 명과 method 명은 변경 하지 않는다.

3. Class 구조

```
public class StringTest{
    public static void main(String args[]) {
        String str = "1.22,4.12,5.93,8.71,9.34";
        double data []= new double[5];
        double sum = 0;

        // StringTokenizer를 이용하여 데이터를 꺼낸다.
        for(){
            // 배열에 실수 데이터를 넣는다.
        }

        for(){
            //배열 데이터의 합을 구한다.
        }

        // 결과 값을 출력한다.
    }
}
```

4. 실행 결과

* 합계와 평균은 모두 소수점 4자리에서 반올림하여 소수점 3자리까지만 표현한다.

```
합계 : 29.320
평균 : 5.864
```

[문제 4] 아래 조건에 맞는 프로그램을 작성하시오

2부터 5까지의 정수형 데이터만을 argument로 입력 받아, 1부터 입력 받은 데이터까지의 합을 출력한다.
단, 입력 받은 데이터가 2부터 5까지의 범위를 벗어나면 "입력 값에 오류가 있습니다."를 출력한다.

1. 사용 데이터

2부터 5까지의 정수를 입력 받는다.

2. 구현 클래스

Package명	클래스명	메소드	설명
com.elite.test04	ExTest	+main(args:String[]):void	main 함수 안에서 Calc 객체를 생성 하여 getSum()를 호출하고 발생한 Exception 처리를 한다.
	Calc	+getSum(data:int):double	1부터 data까지의 합을 계산한다. 입력 받은 숫자가 2부터 5까지의 범위를 벗어나면 InvalidException을 발생 시키기만 한다.
	InvalidException	+InvalidException()	기본 생성자

* 클래스 명과 메소드 명은 변경하지 않는다.

3. 구현 클래스 구조(Calc, InvalidException)

```
public class Calc{
    //getSum() 구현
}

public class InvalidException extends Exception{
    public InvalidException(){
        super();
    }
}
```

4. 실행 결과

5입력 시

결과 값: 15.0

6입력 시

입력 값에 오류가 있습니다.

[문제 5] 아래 조건에 맞는 프로그램을 작성하시오

HashMap에 1~100사이의 정수형 데이터 10개를 랜덤하게 만들어서 저장한다

HashMap에 데이터 입력 시, 순서대로 1부터 10까지의 Number를 Key로 이용하여 입력한다.

HashMap에 입력한 값을 화면에 출력하고 합계와 평균을 출력한다.

1. 사용 데이터

1~100까지의 정수형 데이터를 랜덤하게 발생하여 저장한다.

2. 구현 클래스

Package명	클래스명	메소드	설명
com.elite.test05	HashMapTest	<u>+main(args:String[]):void</u>	main 함수 안에서 모든 코드 작업 진행

* 클래스 명과 메소드 명은 변경하지 않는다.

3. 구현 클래스 구조

```
public class HashMapTest{
    public static void main(String args[]) {

        HashMap<Integer, Integer> map = new HashMap<Integer, Integer>();
        double sum=0.0;

        for() {
            // HashMap에 1~100까지의 정수를 랜덤하게 10개 저장한다.
        }

        for() {
            // HashMap 데이터를 화면에 출력한다.
        }

        for() {
            // HashMap 데이터의 합계를 계산한다.
        }

        // 합계와 평균을 화면에 출력한다.

    }
}
```

4. 실행 결과

합계와 평균은 소수점 3자리에서 반올림하여 소수점 2자리까지 표현한다.

```
18 52 73 45 64 70 32 74 27 27
합계 : 482.00
평균 : 48.20
```

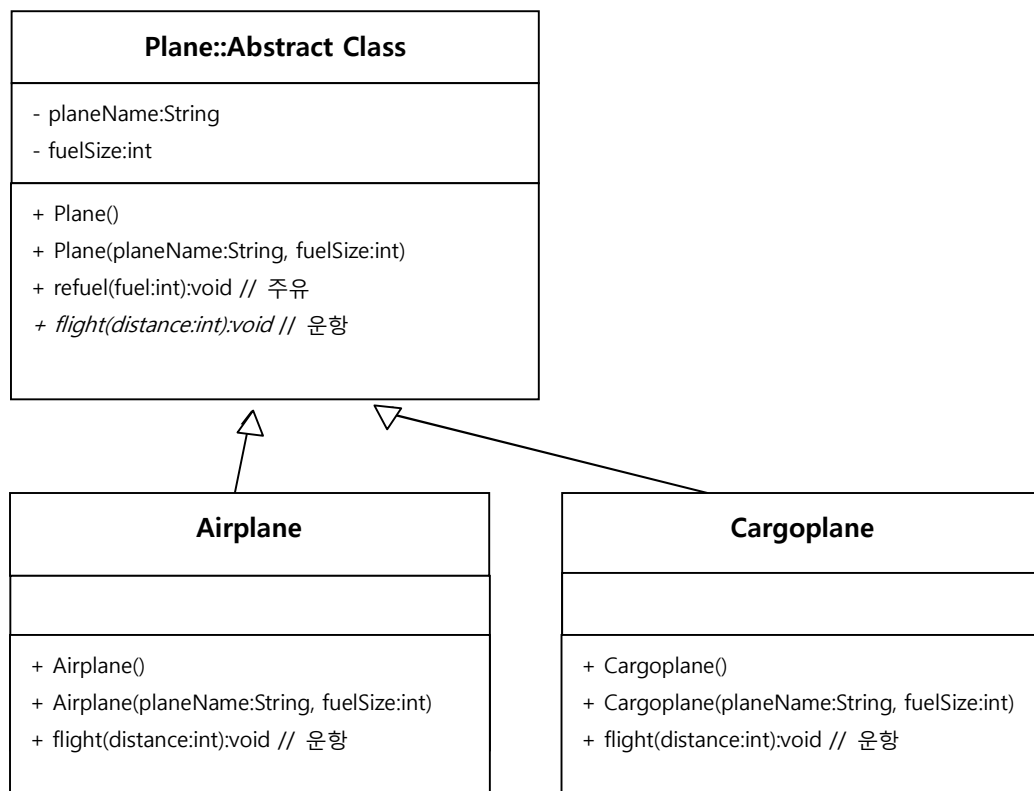
[문제 6] 아래 조건에 맞는 프로그램을 작성하시오

Abstract 클래스에서 상속받은 두 개의 클래스를 구현하여 실행 결과의 예와 같이 나오도록 프로그램을 작성한다.

1. 사용 데이터

class Type	planeName	fuelSize
Airplane	L747	1000
Cargoplane	C40	1000

2. 클래스 다이어그램



* 모든 클래스 변수의 getter, setter 함수는 직접 구현한다.

3. 구현 클래스

Package명	클래스명	메소드	설명
com.elite.test06	Plane	+Plane ()	기본 생성자
		+Plane (planeName:String, fuelSize:int)	2개의 클래스 변수를 받는 생성자
		+refuel (fuel:int) :void	일정 양의 연료 주입, 기존 연료가 증가됨
		+flight (distance:int) :void	일정 거리 만큼 운항, 연료 감소 추상 함수 하위 객체에서 반드시 구현해야 함
	Airplane	+Airplane ()	기본 생성자
		+Airplane (planeName:String, fuelSize:int)	2개의 클래스 변수를 받는 생성자
		+flight (distance:int) :void	10 운항 시 연료 30감소

	Cargoplane	+Cargoplane()	기본 생성자
		+Cargoplane(planeName:String, fuelSize:int)	2개의 클래스 변수를 받는 생성자
		+flight(distance:int):void	10운항 시 연료 50감소
	PlaneTest	+ main(args:String[]):void	main 함수를 이용 하여 동작

- * class 명과 method 명은 변경하지 않는다.
- * 클래스 변수의 getter, setter 함수는 모두 구현한다.

4. PlaneTest 클래스 구조

```
public class PlaneTest {
    public static void main(String args[]) {
        // Airplane과 Cargoplane 객체 생성

        // 생성된 객체의 정보 출력

        // Airplane과 Cargoplane 객체에 100씩 운항

        // 운항 후 객체의 변경된 정보 출력

        // Airplane과 Cargoplane 객체에 200씩 주유

        // 주유 후 객체의 변경된 정보 출력
    }
}
```

5. 실행 결과 예

Plane	fuelSize
L747	1000
C40	1000
100 운항	
Plane	fuelSize
L747	700
C40	500
200 주유	
Plane	fuelSize
L747	900
C40	700

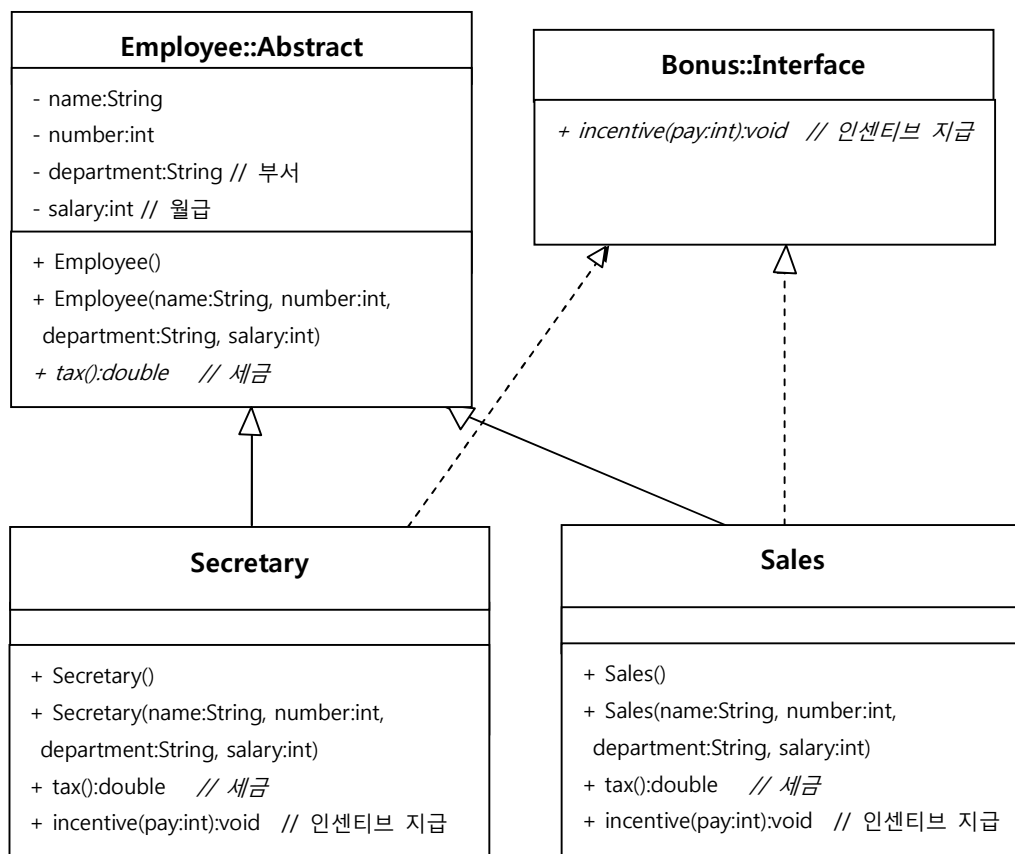
[문제 7] 아래 조건에 맞는 프로그램을 작성 하시오

Abstract 클래스를 상속받고 Interface를 implements한 두 개의 클래스를 구현하여 실행 결과의 예와 같이 나오도록 프로그램을 작성한다.

1. 사용 데이터

name	number	department	salary
Hilery	1	secretary	800
Clinton	2	sales	1200

2. 클래스 다이어그램



* 모든 클래스 변수의 getter, setter 함수는 직접 구현한다.

3. 구현 클래스

Package명	클래스명	메소드	설명
com.elite.test07	Employee	+Employee()	기본 생성자
		+Employee(name:String, number:int, department:String, salary:int)	4개의 클래스 변수를 받는 생성자
		+tax():double	세금을 리턴한다.
	Bonus	+incentive(pay:int):void	인센티브를 지급한다.
	Secretary	+ Secretary()	기본 생성자
		+ Secretary(name:String, number:int, department:String, salary:int)	4개의 클래스 변수를 받는 생성자
		+tax():double	세금을 리턴한다. salary에 10% 징수
		+incentive(pay:int):void	인센티브 지급. 지급되는 pay에 80%가 기존 salary에 더해진다.
	Sales	+ Sales()	기본 생성자
		+ Sales(name:String, number:int, department:String, salary:int)	4개의 클래스 변수를 받는 생성자
		+tax():double	세금을 리턴한다. salary에 13% 징수
		+incentive(pay:int):void	인센티브 지급. 지급되는 pay에 120%가 기존 salary에 더해진다.
	Company	+main(args:String[]):void	main 함수를 이용하여 동작

* class 명과 method 명은 변경하지 않는다.

* 클래스 변수의 getter, setter 함수는 모두 구현한다.

4. Company 클래스 구조

```

public class Company {

    public static void main(String args[]) {
        HashMap<Integer, Employee> map = new HashMap<Integer, Employee>();

        // 1번의 데이터를 기반으로 객체를 생성하여 HashMap에 넣는다.

        // HashMap에 넣을 때 키 값은 각 객체의 Number로 한다.

        // 모든 객체의 기본 정보를 출력한다(for문을 이용하여 출력한다.)

        // 모든 객체에 인센티브 100씩 지급한다.

        // 모든 객체의 정보와 세금을 출력한다 (for문을 이용하여 출력한다.)
    }
}

```

5. 실행 결과

name	department	salary
Hilery	secretary	800
Clinton	sales	1200

인센티브 100 지급

name	department	salary	tax
Hilery	secretary	880	88.0
Clinton	sales	1320	171.6