

서로 다른 Student 객체를 생성 하여 모든 과목의 평균, 학점을 조회하는 프로그램을 작성 하시오.

1. 사용 데이터

name	korean	english	math	science
Kim	100	90	95	89
Lee	60	70	99	98
Park	68	86	60	40

Student
- name:String - korean:int - english:int - math:int - science:int
+ Student() + Student (name:String,korean:int,english:int,math:int, science:int) + getAvg():double + getGrade():String + getXXX() + setXXX()

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

Package명	Class명	method	설명
student	Student	+Student()	기본 생성자
		+ Student (name:String,korean:int,english:int,math:int, science:int)	5개의 클래스 변수를 받는 생성자
		+getAvg():double	평균값 리턴
		+getGrade():String	학점 리턴
	Test01	<u>+main(String args[]): void</u>	main 함수 안에서 Student 객체를 생성하여 동작 시킨다

* getX/set은 모두 구현 하시오.

4. TestStudent 클래스 구조

Student 객체를 생성 하고 각각의 정보를 출력 한다.

```
public class TestStudent {  
    public static void main(String args[]) {  
        // Student 객체를 3개 생성한다.  
        // 각각의 객체의 평균과 학점을 화면에 출력 한다.  
    }  
}
```

5. 실행 결과

실행 결과 예)

```
Kim 평균: 93.5 학점: A학점  
Lee 평균: 81.75 학점: B학점  
Park 평균: 63.5 학점: C학점
```

[문제 2] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오.

서로 다른 Book 객체를 생성 하고 각각의 정보를 출력 하는 프로그램을 작성 하시오.

1. 사용 데이터

아래와 같이 3개의 Book Object를 생성 하여 프로그램을 동작 시킨다

bookName	bookPrice(원)	bookDiscountRate(%)
SQL Plus	50000	5.0
Java 2.0	40000	3.0
JSP Servlet	60000	6.0

2. 구현 클래스

Book
- bookName:String - bookPrice:int - bookDiscountRate:double
+ Book() + Book(bookName:String,bookPrice:int,bookDiscountRate:double) + getDiscountBookPrice():double + getXXX() + setXXX()

TestBook
main(args:String[]):void

3. 구현 클래스

Package명	Class명	method	설명
book	Book	+Book()	기본 생성자
		+Book(bookName:String,bookPrice:int,bookDiscountRate:double)	3개의 클래스 변수를 받는 생성자
		+getDiscountBookPrice():double	인하된 책 가격을 리턴
	BookTest	+main(String args[]): void	main 함수 안에서 Book 객체를 생성하여 동작 시킨다

* class 명과 method 명은 변경 하지 않는다.

* 위에 선언한 클래스 변수와 클래스 함수만을 이용한다.

* getXXX/setXXX는 모두 만들어서 사용한다.

4. TestBook 클래스 구조

3개의 Book 객체를 생성 하여 각각의 정보를 화면에 출력 한다.

```
public class TestBook {  
    public static void main(String args[]) {  
  
        // 3개의 Book 객체를 생성한다  
        // 각각의 Book 객체의 정보를 실행 결과 예와 같이 출력 한다.  
    }  
}
```

5. 실행 결과

실행 결과 예)

책이름	가격	할인율	할인후금액
SQL Plus	50000원	5.0%	47500.0원
Java 2.0	40000원	3.0%	38800.0원
JSP Servlet	60000원	6.0%	56400.0원

[문제 3] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오

1. 조건

고정 금리의 계좌를 생성 하여 현재 잔액을 기준으로 이자를 계산하는 프로그램을 작성 한다.

사용 데이터

Account(계좌번호): 441-0290-1203, balance(잔액): 500000원, interestRate(이율): 7.3%

2. 구현 클래스

Account	TestAccount
- account:String - balance:int - interestRate:double + Account() + Account(account:String, balance:int, interestRate:double) + calculateInterest():double + deposit(money:int):void + withdraw(money:int):void	+main(args:String[]):void

3. 구현 클래스

Package명	Class명	Method	설명
account	Account	+Account()	기본 생성자
		+Account(account:String, balance:int, interestRate:double)	3개의 클래스변수를 받는 생성자
		+calculateInterest():double	현재 잔액을 기준으로 이자를 계산 한다
		+deposit(money:int):void	입금을 통해 잔액정보를 증가 시킨다
		+withdraw(money:int):void	출금을 통해 잔액정보를 감소 시킨다 <u>단, 출금을 통해 잔액이 0보다 작으면 "출금 할 수 없습니다."를 출력한다.</u>
	TestAccount	+main(args:String[]): void	main 함수 안에서 Account객체 변수를 선언하고 생성 하여 동작 시킨다

* class 명과 method 명은 변경 하지 않는다

* 위에 선언한 클래스 변수와 클래스 함수만을 이용한다

* getX/setXXX는 필요 시 만들어서 사용한다

4. TestAccount 클래스 구조

```
public class TestAccount {  
    public static void main(String args[]) {  
        Account account;  
        // account 객체 생성  
        // account 기본 정보 출력  
        // account 에서 600000원 출금  
        // account 에 20000원 입금  
        // account 변경 정보 출력  
        // 이자 출력 - 현재 잔고를 기준으로 고객에게 줄 이자 금액을 출력 한다  
    }  
}
```

5. 실행 결과

이자 계산: 현재잔고 x 이자율

실행 결과 예)

계좌정보: 441-0290-1203 현재잔액: 500000

출금 할 수 없습니다.

계좌정보: 441-0290-1203 현재잔액: 520000

이자: 37960.0

[문제 4] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오.

1. 조건

- 1) eclipse Argument 에서 월 기본급을 입력 받는다. (정수형으로 입력)
 - 2) 입력된 기본급을 기준으로 직원들의 연봉을 계산하는 프로그램을 작성 한다.
- * 연봉 계산 기준
- 1) 기본급: 입력 데이터
 - 2) 보너스: 분기별로 월 기본급의 20% 지급
 - 3) 세금: 기본급은 10% 공제, 보너스는 5.5% 공제
- 1년 동안 세금을 제외한 총 지급 금액을 계산 한다.

3. 클래스 다이어그램

Company
<ul style="list-style-type: none"> - salary:double - income:double - afterTaxIncome:double - bonus:double - afterTaxBonus:double
<ul style="list-style-type: none"> + Company() + Company(salary:double) + getXXX() + setXXX()

TestCompany
<ul style="list-style-type: none"> +main(args:String[]):void

2. 구현 클래스

Package명	Class명	method	설명
company	TestCompany	+main(String args[]): void	main 함수 안에서 모든 코드 작업 진행
	Company	+Company()	기본 생성자
		+Company(salary:double)	1개의 클래스 변수를 받는 생성자
		+getIncome():double	연 기본급의 합을 계산하여 리턴
		+getAfterTaxIncome():double	연 기본급의 합에서 세금을 공제한 금액을 리턴
		+getBonus():double	연 보너스의 합을 계산하여 리턴
		+getAfterTaxBonus():double	연 보너스의 합에서 세금을 공제한 금액을 리턴

3. 실행 결과

모든 계산은 실수로 표현 한다.

100입력 시 출력 결과는 아래와 같다.

연 기본급 합: 1200.0 세후: 1080.0
연 보너스 합: 80.0 세후: 75.6
연 지급액 합: 1155.6

[문제 5] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오.

1. 조건

프로그램 실행 시 eclipse argument로 5~10까지의 정수형 데이터를 입력 받는다.

5~10 이외의 정수형이 입력되면 "다시 입력하세요"라고 출력 한다.

입력 받은 정수 값을 Calc class의 calculate() 함수를 이용하여 1부터 입력 받은 숫자까지 짝수만 더하는 프로그램을 작성 한다.

2. 구현 클래스

Package명	Class명	method	설명
calc	TestCalc	<u>+main(String args[]): void</u>	Calc 객체를 생성 하고 calculate() 함수를 이용하여 결과를 받아 출력 한다.
	Calc	+calculate(data:int):int	int형 정수를 입력 받아 1부터 입력 받은 숫자까지 짝수의 합을 구하여 리턴한다.

3. 실행 결과

출력은 아래와 같이 이루어 진다.

입력 값: 5

결과: 6

[문제 6] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오

3개의 Student 객체를 생성 하여 배열에 저장 한 후 나이, 신장, 몸무게의 평균을 출력 한다

1. 사용 데이터

아래와 같이 3개의 Student Object를 생성 하여 프로그램을 동작 시킨다

name	나이	신장	몸무게
홍길동	15	171	81
한사람	13	183	72
임걱정	16	175	65

2. 구현 클래스

Student
- name:String - age:int - height:int - weight:int + Student() + Student(name:String,age:int,height:int,weight:int) + getXXX() + setXXX()

StudentTest
+main(args:String[]):void

3. 구현 클래스

Package명	Class명	method	설명
school	Student	+ Student()	기본 생성자
		+ Student(name:String,age:int,height:int,weight:int)	4개의 클래스 변수를 받는 생성자
	StudentTest	+main(String args[]): void	main 함수 안에서 Student 타입의 배열을 선언하여 동작 시킨다

* class 명과 method 명은 변경 하지 않는다

* 위에 선언한 클래스 변수와 클래스 함수만을 이용한다

* getXXX와 setXXX는 직접 구현 한다

4. StudentTest 클래스 구조

Student 객체를 담을 수 있는 배열을 선언 하여 3개의 Student 객체를 생성 하여 담는다

```
public class StudentTest {  
    public static void main(String args[]) {  
  
        Student studentArray [] = new Student[3];  
        // Student 객체를 3개 생성하여 배열에 넣는다  
        // 배열에 있는 객체 정보를 모두 출력 한다 - for 문을 이용 할 것  
        // 나이의 평균을 출력한다.  
        // 신장의 평균을 출력한다.  
        // 몸무게의 평균을 출력한다.  
    }  
}
```

5. 실행 결과

실행 결과 예) 평균 값은 소수점 3자리에서 반올림하여 2자리까지 표현한다

이름	나이	신장	몸무게
홍길동	15	171	81
한사람	13	183	72
임걱정	16	175	65

나이의 평균: 14.67

신장의 평균: 176.33

몸무게의 평균: 72.67

[문제 7] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오

3개의 Student 객체를 생성 하여 배열에 저장 한 후 3개의 객체 중 가장 큰 신장과 몸무게, 나이의 평균을 구한다

1. 사용 데이터

아래와 같이 3개의 Student Object를 생성 하여 프로그램을 동작 시킨다

name	나이	신장	몸무게
홍길동	15	170	80
한사람	13	180	70
임걱정	16	175	65

2. 구현 클래스

Student
- name:String - age:int - height:int - weight:int + Student() + Student(name:String,age:int,height:int,weight:int) + studentInfo():String + setXXX + getXXX

StudentTest
+main(args:String[]):void

3. 구현 클래스

Package명	Class명	method	설명
school2	Student	+ Student()	기본 생성자
		+ Student(name:String,age:int,height:int,weight:int)	4개의 클래스 변수를 받는 생성자
		+ studentInfo():String	학생의 모든 정보 리턴
	StudentTest	+main(String args[]): void	main 함수 안에서 Student 타입의 배열을 선언하여 동작 시킨다

* class 명과 method 명은 변경 하지 않는다

4. StudentTest 클래스 구조

Student 객체를 담을 수 있는 배열을 선언 하여 3개의 Student 객체를 생성 하여 담는다

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

        Student studentArray [] = new Student[3];
        // Student 객체를 3개 생성하여 배열에 넣는다
        // 배열에 있는 객체 정보를 모두 출력 한다 - for문을 이용할 것, studentInfo 함수 사용
        // Student 객체들의 나이의 평균 출력
        // Student 객체들의 신장의 평균 출력
        // Student 객체들의 몸무게의 평균 출력
        // 학생들 중 나이가 가장 적은 학생과 많은 학생 출력
        // 학생들 중 신장이 가장 적은 학생과 큰 학생 출력
        // 학생들 중 몸무게가 가장 적은 학생과 많이 나가는 학생 출력

    }
}
```

5. 실행 결과

실행 결과 예)

평균 정보는 모두 double로 계산,

평균 출력은 소수점 이하 4째 자리에서 반올림하고 3째 자리까지 표현

이름	나이	신장	몸무게
홍길동	15	170	80
한사람	13	180	70
임걱정	16	175	65

나이 평균: 14.667

신장 평균: 175.000

몸무게 평균: 71.667

나이가 가장 많은 학생: 임걱정

나이가 가장 적은 학생: 한사람

신장이 가장 큰 학생: 한사람

신장이 가장 작은 학생: 홍길동

몸무게가 가장 많이 나가는 학생: 홍길동

몸무게가 가장 적게 나가는 학생: 임걱정

[문제 8] 아래 조건에 맞는 프로그램을 작성 하시오

3개의 Coffee 객체를 배열에 저장하고 있는 Cafe 객체를 생성 하여 총 금액을 출력하는 프로그램을 작성 한다.

1. 사용 데이터

아래와 같이 3개의 Coffee Object를 생성 하여 프로그램을 동작 시킨다.

name	price(원)
Americano	4000
Caffelatte	5000
Macchiato	6000

2. 구현 클래스

Coffee
- name:String - price:int
+ Coffee() + Coffee(name:String,price:int) + setPrice(price:int):void + getPrice():int + toString():String

CafeTest
+main(args:String[]):void

Cafe
- coffeeList:Coffee[] - index:int
+ Cafe() + totalPrice():int + setCoffee(coffee:Coffee): void + getCoffeeList() : Coffee[]

3. 구현 클래스

Package명	Class명	method	설명
cafe	Coffee	+Coffee ()	기본 생성자
		+Coffee (name:String,price:int)	2개의 클래스 변수를 받는 생성자
		+setPrice(price:int) : void	가격을 셋팅
		+getPrice():int	가격을 리턴
		+toString():String	Coffee 객체의 모든 정보를 조합하여 리턴한다.
	Cafe	+Cafe ()	기본 생성자

		+setCoffee(coffee:Coffee) : void	Coffee 객체를 배열에 저장한다. 배열 저장 위치는 index 값을 이용한다. 배열에 모두 저장되면 “더 이상 저장할 수 없습니다” 를 출력한다.
		+totalPrice() : int	Coffee [] 객체를 반복 수행하면서 가격의 합을 계산하고 리턴한다.
		+getCoffeeList() : Coffee[]	Coffee[] 객체를 리턴한다.
	CafeTest	<u>+main(String args[]): void</u>	main 함수 안에서 Café [] 객체를 선언하여 동작 시킨다

- * class 명과 method 명은 변경 하지 않는다.
- * 위에 선언한 클래스 변수와 클래스 함수만을 이용한다.
- * getXXX/setXXX는 필요 시 만들어서 사용한다.

4. CafeTest 클래스 구조

Cafe 객체를 선언 하여 3개의 Coffee 객체를 생성 하여 담는다.

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

        Cafe cafe = new Cafe();
        // Coffee 객체를 3개 생성하고, setCoffee() 이용하여 Coffee 배열에 넣는다.
        // Coffee 배열에 있는 객체 정보를 모두 출력 한다 - for 문을 이용 할 것
        // Coffee 가격의 합을 출력 한다

    }
}
```

5. 실행 결과

실행 결과 예)

```
Americano      4000
Caffelatte     5000
Macchiato      6000

Coffee 가격의 합 : 15000원
```