

[문제 1] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오**1. 조건**

다음 배열의 내용을 실행 결과와 같이 출력 되도록 프로그램을 작성 하시오.

```
int[] arr = {10, 20, 30, 40, 50, 60, 70, 80, 90, 100};
```

2. 구현 클래스

Package명	Class명	method	설명
workshop	Test01	<u>+main(String args[]): void</u>	main 함수 안에서 모든 코드 작업 진행

3. 실행 결과

출력은 아래와 같이 이루어 진다.

```
100 90 80 70 60 50 40 30 20 10
```

[문제 2] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오.

1. 조건

프로그램 실행 시 eclipse argument로 5~10까지의 정수형 데이터를 입력 받는다.
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
workshop	Test02	<u>+main(String args[]): void</u>	Calc 객체를 생성 하고 calculate() 함수 를 이용하여 결과를 받아 출력 한다.
	Calc	+calculate(int data):int	int형 정수를 입력 받아 1부터 입력 받 은 숫자까지 짝수의 합을 구하여 리턴한다.

3. 실행 결과

출력은 아래와 같이 이루어 진다.

입력 값: 5

짝수: 2 4

결과: 6

[문제 3] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오.

1. 조건

eclipse에서 argument로 1~5까지의 정수형 데이터 하나를 입력 받아 입력 받은 수부터 10까지 합을 구한다.

단, 3의 배수와 5의 배수는 합에서 제외 한다.

2. 구현 클래스

Package명	Class명	method	설명
workshop	Test03	<u>+main(String args[]): void</u>	main 함수 안에서 모든 코드 작업 진행

3. 실행 결과

출력은 아래와 같이 이루어 진다.

5을 입력 할 경우

7 + 8

결과 : 15

[문제 4] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오.

1. 조건

- 1) eclipse Argument 에서 1~5까지의 두 개의 정수형 데이터를 입력 받는다.
- 2) 입력 받은 데이터가 2개 미만 또는 2개 초과로 입력 하면 “다시 입력 하세요” 출력
- 3) 1~5이외의 숫자가 입력 될 경우 “숫자를 확인 하세요” 출력
- 4) 입력 받은 두 개의 정수를 이용하여 2차원 배열을 생성한다.
- 5) 2차원 배열에 1~5까지의 랜덤한 숫자(중복허용)를 넣는다.
- 6) 배열의 내용을 출력 한다.
- 7) 배열의 총합과 평균을 실수형으로 출력 한다.

2. 구현 클래스

Package명	Class명	method	설명
workshop	Test04	<u>+main(String args[]): void</u>	main 함수 안에서 모든 코드 작업 진행

3. 실행 결과

출력은 아래와 같이 이루어 진다.

2 2 를 입력 할 경우

```
2 2
5 5

sum=14.0
avg=3.5
```

[문제 5 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
- account:String - balance:int - interestRate:double
+ Account() + Account(account:String, balance:int, interestRate:double) + calculateInterest():double + deposit(money:int):void + withdraw(money:int):void + getXXX() + setXXX()

AccountTest
+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</u> <u>"출금 할 수 없습니다."를 출력한다.</u>
	AccountTest	+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

* getXXX/setXXX는 필요 시 만들어서 사용한다

4. AccountTest 클래스 구조

```
public class AccountTest {  
    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 6] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오.

1. 조건

7번에서 생성 한 Account Class를 이용하여 다음 조건에 맞는 프로그램을 작성 하시오.

- 1) Account 객체 5개를 저장하기 위한 배열을 선언한다.
- 2) for 문을 이용하여 5개의 Account 객체를 생성하고 배열에 저장 한다.
 - 계좌번호: 221-0101-211X(X 부분은 1부터 5까지의 정수를 사용한다)
 - 잔액 및 이자율은 모두 100000원, 4.5% 이다.
- 3) Account class에 Account의 모든 정보를 출력 할 수 있는 accountInfo()를 추가 한다.


```
( + accountInfo() : void )
```

 - 출력예: 계좌번호: 221-0101-2111 잔액: 100000원 이자율: 4.5%
- 4) for문을 이용하여 생성된 배열의 모든 정보를 출력 한다.(출력 시 accountInfo()함수 이용)
- 5) for문을 이용하여 모든 Account 객체의 이자율을 3.7% 변경 하고 이자를 화면에 출력 한다.

2. 구현 클래스

Package명	Class명	Method	설명
account	AccountTest2	<u>+main(args:String[]): void</u>	main 함수 안에서 Account객체 변수를 선언하고 생성 하여 동작 시킨다

3. AccountTest2 클래스 구조

```
public class AccountTest2 {
    public static void main(String args[]) {

        // 5개의 Account 형 객체 배열 선언
        // for문을 이용하여 Account 객체를 생성 후 배열에 저장
        // for문을 이용하여 Account 정보 출력
        // for문을 이용하여 이자율을 변경하고 Account 정보와 이자를 화면에 출력
    }
}
```

4. 실행 결과 예

계좌번호: 221-0101-2111 잔액: 100000원 이자율: 4.5%

계좌번호: 221-0101-2112 잔액: 100000원 이자율: 4.5%

계좌번호: 221-0101-2113 잔액: 100000원 이자율: 4.5%

계좌번호: 221-0101-2114 잔액: 100000원 이자율: 4.5%

계좌번호: 221-0101-2115 잔액: 100000원 이자율: 4.5%

계좌번호: 221-0101-2111 잔액: 100000원 이자율: 3.7% 이자: 3700원

계좌번호: 221-0101-2112 잔액: 100000원 이자율: 3.7% 이자: 3700원

계좌번호: 221-0101-2113 잔액: 100000원 이자율: 3.7% 이자: 3700원

계좌번호: 221-0101-2114 잔액: 100000원 이자율: 3.7% 이자: 3700원

계좌번호: 221-0101-2115 잔액: 100000원 이자율: 3.7% 이자: 3700원