

[문제 1] 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 더하고 결과는 아웃풋과 같이 프로그램을 작성한다.

2. 구현 클래스

Package명	Class명	method	설명
com.elite	Test01	<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 숫 자까지 짝수의 합을 구하여 리턴한다.

3. 실행 결과

출력은 아래와 같이 이루어진 다.

forEach(), mapToObj(), filter(), collect(), max(), min() 등을 활용한다.

정수를 입력하세요 (5~10): 8

1부터 8까지의 숫자:

1 2 3 4 5 6 7 8

짝수 판별:

1 (홀수) 2 (짝수) 3 (홀수) 4 (짝수) 5

(홀수) 6 (짝수) 7 (홀수) 8 (짝수)

입력 값: 8

짝수 리스트: [2, 4, 6, 8]

짝수 개수: 4

짝수 중 가장 큰 값: 8

짝수 중 가장 작은 값: 2

짝수의 합: 20

[문제 2] 아래 조건에 맞는 프로그램을 작성 하시오

월을 입력 받아 그 월의 마지막 일자를 출력하는 프로그램을 작성한다

1. 사용 데이터

- 프로그램 실행 시 1개의 숫자를 argument로 받아 들인다
- 1개의 숫자는 1~12까지의 정수형태로만 받아 들인다, 1~12이외의 숫자를 입력 하면 "입력된 값이 잘못 되었습니다"라고 화면에 출력 한다
- 1~12 사이의 숫자인지 검증 → `IntStream.rangeClosed(1, 12).noneMatch(n -> n == month)`
- 각 월의 마지막 날짜 찾기 → `Map<Integer, Integer> monthDays.getDefault(month-1)`
- `filter()`와 `findFirst()`를 사용하여 월의 값 추출 가능
- 각 월의 마지막 날은 아래와 같다

Number of Days	Months	Month Number
30	April, June, September, November	4, 6, 9, 11
31	January, March, May, July, August, October, December	1, 3, 5, 7, 8, 10, 12
28	February	2

2. 구현 클래스

Package명	Class명	method	설명
month	Months	+ <code>getDays(months:int):int</code>	월을 argument로 받아 그 월의 마지막 일자를 리턴 한다
	MonthsTest	+ <code>main(args:String[]):void</code>	main 함수 안에서 Months 객체를 생성하여 작업 진행

* class 명과 method 명은 변경하지 않는다

* Months의 클래스 변수는 필요 시 자유롭게 선언하여 사용한다

3. 실행 결과

4를 입력 하였을 경우

월을 입력하세요 (1~12): 4

입력받은 월: 04월

짝수/홀수 여부: 짝수월

마지막 일자: 30일

[문제 3] 아래 조건에 맞는 프로그램을 작성 하시오

- 3개의 Coffee 객체를 배열에 저장하고 있는 Cafe 객체를 생성하여 총 금액을 출력하는 프로그램을 작성한다.

1. 사용 데이터

아래와 같이 3개의 Coffee Object를 생성하여 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 <u>"더 이상 저장할 수 없습니다"</u> 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 선언하여 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 - forEach문을 이용 할 것
        // Coffee 가격의 합을 출력 한다

    }
}
```

5. 실행 결과

```
Americano      4000
Caffelatte     5000
Macchiato      6000

Coffee 가격의 합: 15000원
```