

Programming Practice03

Problem 제출기한: 4월 19일 오전 10시 까지

주의사항 1. 실습 프로그램 및 과제 제출 방법

- 하나의 Java project에 한 차시에 주어진 모든 Programming Practice 문제를 포함하도록 함
- 단, Assignment는 또 다른 Java project를 생성하여 프로그램을 작성함
- 한 문제당 하나의 package를 생성하여 프로그램 작성함
- 제출 시 Java project 파일을 압축하여 제출함

주의사항 2. 입력 및 출력양식 : 주어진 '입출력예시'에 나타난 것과 정확히 똑같아야 점수로 인정됨

- 사용자로부터 값을 입력받고, 입력받은 수가 짝수이면 'Even', 홀수이면 'Odd'를 출력하도록 하는 프로그램에서 입출력예시가 아래와 같이 주어졌을 때

```
Enter an integer: 4  
Output: Even
```

- 정답으로 인정되는 경우

```
Enter an integer: 3  
Output: Odd
```

```
Enter an integer: 6  
Output: Even
```

- 오답으로 인정되는 경우 (예시)

```
Enter an integer: 6  
Output: Odd // Incorrect
```

```
Enter an integer:  
6 // newline with string literal and input value  
Output: Even
```

```
7 // input format is incorrect (need 'Enter an integer:')  
output: Odd // 'Output:', not 'output:'
```

```
Enter an integer: 6  
Output: even // 'Even:', not 'even'
```

```
Enter an integer:6  
Output:even // no space after the colon ':'. space needed after the colon
```

Problem01 사용자로부터 양의 정수 1개를 입력받고, 해당 수가 암스트롱 수 인지 판별하는 ArmstrongNumberChecker 클래스를 작성하세요.

- 암스트롱 수는 숫자의 각 자릿수만큼 거듭제곱한 것의 합이 원래의 숫자와 같아질 때를 의미한다.
- 예를 들어, 153은 세자릿 수이므로, 각 자릿수를 세제곱 하면, $153 = 1^3 + 5^3 + 3^3$ 이므로 암스트롱 수이다.
- 예를 들어, 8208은 네자릿 수이므로, 각 자릿수를 네제곱 하면, $8208 = 8^4 + 2^4 + 0^4 + 8^4$ 이므로 암스트롱 수이다.
- 암스트롱 수인 경우 “Yes”를 출력하고, 아닌 경우 “No”를 출력한다.

Table 1: 입출력 예시 (* ‘Input: ’, ‘Output: ’이 반드시 표시되어야 정답으로 인정됨)

입력	출력
Input: 9474	Output: Yes
Input: 1	Output: Yes
Input: 51	Output: No
Input: 3	Output: Yes
Input: 370	Output: Yes

Problem02 사용자로부터 정수 1개 (N)을 입력받고, 해당 숫자가 소수(Prime number)인지 판단하는 PrimeChecker 클래스를 작성하세요.

- N 이 자연수가 아닌 경우, -1을 출력 후 다시 입력 받는다.
- 소수는 1과 자기 자신만 약수로 갖는 수이다.
- 1은 소수가 아니다.
- 소수인 경우 “Yes”, 아닌 경우 “No”를 출력한다.

Table 2: 입출력 예시 (* ‘Input: ’, ‘Output: ’이 반드시 표시되어야 정답으로 인정됨)

입력	출력
Input: 5	Output: Yes
Input: 1	Output: No
Input: -4	Output: -1
Input: 15	Output: No

Problem03 사용자로부터 숫자 1개를 입력받고, 해당 숫자의 구구단 테이블을 출력하는 MultiTable 클래스를 작성하세요.

- 입력된 숫자가 2부터 9사이의 수가 아닌 경우, -1을 출력하고 다시 입력받는다.
- 숫자와 연산자는 공백 한칸으로 구분한다.

Table 3: 입출력 예시 (* 'Input: ', 'Output: '이 반드시 표시되어야 정답으로 인정됨)

입력	출력
Input: 5	Output:
	5 * 1 = 5
	5 * 2 = 10
	5 * 3 = 15
	5 * 4 = 20
	5 * 5 = 25
	5 * 6 = 30
	5 * 7 = 35
	5 * 8 = 40
	5 * 9 = 45
Input: -3	Output: -1
Input: 9	Output:
	9 * 1 = 9
	9 * 2 = 18
	9 * 3 = 27
	9 * 4 = 36
	9 * 5 = 45
	9 * 6 = 54
	9 * 7 = 63
	9 * 8 = 72
9 * 9 = 81	

Problem04 사용자로부터 두 숫자 x, y 를 입력받고, 해당 (x, y) 좌표가 몇 사분면 위의 점인지 출력하는 CoordiChecker 클래스를 작성하세요.

- $-1,000 \leq x \leq 1,000; x \neq 0$
- $-1,000 \leq y \leq 1,000; y \neq 0$
- 입력받는 두 숫자는 쉼표(,)와 공백 하나를 기준으로 구분된다.

Table 4: 입출력 예시 (* 'Input: ', 'Output: '이 반드시 표시되어야 정답으로 인정됨)

입력	출력
Input: 5, 3	Output: 1
Input: -94, 5	Output: 2

Programming Assignment 02

Assignment 제출기한: 4월 19일 오전 10시 까지

Assignment01 사용자로부터 직각삼각형의 높이(N)을 입력받고, 1부터 9까지의 숫자를 입출력 예시와 같은 직각삼각형에 순차적으로 출력하는 DisplayTriNumber 클래스를 작성하세요.

- 숫자 9 다음은 다시 처음부터 1로 시작한다.
- 자연수 외 입력은 고려하지 않는다.
- 숫자 사이의 공백은 \t이다.

Table 5: 입출력 예시 (* 'Input: ', 'Output: '이 반드시 표시되어야 정답으로 인정됨)

입력	출력
	Output:
	1
Input: 5	2 3
	4 5 6
	7 8 9 1
	2 3 4 5 6

Assignment02 사용자로부터 양의 정수 1개 (N)을 입력받고, N 을 소인수를 소인수 분해한 후 소인수의 요소들을 출력하는 PrimeFactorization 클래스를 작성하세요.

- $1 \leq N \leq 10,000$
- 소인수 분해란 주어진 수를 소수의 곱으로만 이루어지도록 만드는 것이다.
- 예를 들어, 10을 소인수 분해하면 $2 * 5$ 로 나타낼 수 있고, 20을 소인수 분해하면 $2 * 2 * 5$ 로 나타낼 수 있다.
- 각 소인수가 오름차순으로 출력되어야 한다.
- 연산자와 피연산자는 사이는 공백 1칸을 갖는다.
- 양의 정수가 아닌 입력은 고려하지 않는다.

Table 6: 입출력 예시 (* 'Input: ', 'Output: '이 반드시 표시되어야 정답으로 인정됨)

입력	출력
Input: 20	Output: 20 = 2 * 2 * 5

Input: 320

Output: $320 = 2 * 2 * 2 * 2 * 2 * 2 * 2 * 5$

Assignment03 사용자로부터 정수 1개 (N)을 입력받고, 2부터 N 까지의 숫자 중 소수(Prime number)를 모두 찾아 출력하는 AllPrimeNumber 클래스를 작성하세요.

- N 은 2이상 100이하의 정수이다.
- 출력 시 숫자 간 공백 1칸을 두고 출력한다.
- 범위 밖의 수가 입력되면 -1을 출력 후 다시 입력 받는다.

Table 7: 입출력 예시 (* 'Input: ', 'Output: '이 반드시 표시되어야 정답으로 인정됨)

입력	출력
Input: 5	Output: 2 3 5
Input: 15	Output: 2 3 5 7 11 13
Input: 1	Output: -1
Input: 3	Output: 2 3

Assignment04 사용자로부터 삼각형의 두 변의 길이(정수)를 입력받고, 삼각형의 각 변에서 가장 "긴 변의 길이는 나머지 두 변보다 작아야한다." 라는 조건에 맞는 길이가 정수인 변의 개수를 계산하는 TriChecker 클래스를 작성하세요.

- 예를 들어, 입력이 3과 6인 경우, 1) 6이 가장 긴 변이라면 $3+4$, $3+5$, $3+6$ 의 세 가지 경우가 가능하고, 2) 6이 가장 긴 변이 아니라면, $3+6=9$ 보다 작은 정수형 변의 길이인 7, 8인 두 가지 경우가 가능하므로 그 개수는 5개 이다.
- 범위 밖의 수는 입력되지 않는다고 가정한다.

Table 8: 입출력 예시 (* 'Input: ', 'Output: '이 반드시 표시되어야 정답으로 인정됨)

입력	출력
Input: 3 6	Output: 5
Input: 11 7	Output: 13
Input: 1 2	Output: 1