

반복문 실습 문제

▶ 실습문제7

메소드 명 : `public void practice7(){`

정수 두 개와 연산자를 입력 받고 입력된 연산자에 따라 알맞은 결과를 출력하세요.

단, 해당 프로그램은 연산자 입력에 "exit"라는 값이 들어올 때까지 무한 반복하며
exit가 들어오면 "프로그램을 종료합니다."를 출력하고 종료합니다.

또한 연산자가 나누기이면서 두 번째 정수가 0으로 들어오면

"0으로 나눌 수 없습니다. 다시 입력해주세요."를 출력하며,

없는 연산자가 들어올 시 "없는 연산자입니다. 다시 입력해주세요."라고 출력하고

두 경우 모두 처음으로 돌아가 사용자가 다시 연산자부터 입력하도록 하세요.

[다음 장 출력 예시 참고]

▶ 실습문제7

메소드 명 : public void practice7(){

연산자(+, -, *, /, %) : +

정수1 : 10

정수2 : 4

10 + 4 = 14

연산자(+, -, *, /, %) : /

정수1 : 10

정수2 : 4

10 / 4 = 2

연산자(+, -, *, /, %) : ^

정수1 : 10

정수2 : 4

없는 연산자입니다. 다시 입력해주세요.

연산자(+, -, *, /, %) : /

정수1 : 10

정수2 : 0

0으로 나눌 수 없습니다. 다시 입력해주세요.

연산자(+, -, *, /, %) : exit

프로그램을 종료합니다.

▶ 실습문제8

메소드 명 : public void practice8(){}
다음과 같은 실행 예제를 구현하세요.

ex.

정수 입력 : 4

*

**

▶ 실습문제9

메소드 명 : `public void practice9(){}
다음과 같은 실행 예제를 구현하세요.`

ex.

정수 입력 : 4

**

*

▶ 실습문제10

메소드 명 : `public void practice10(){}`

사용자로부터 입력 받은 하나의 값이 소수인지 판별하는 프로그램을 구현하세요.

단, 입력한 수가 2보다 작은 경우 "잘못 입력하셨습니다."를 출력하세요.

* '소수'는 n을 1부터 n까지로 나누었을 때 나누어 떨어지는 수가 1과 n뿐인 수를 말한다.

ex) 2, 3, 5, 7, 11 ...

ex.

숫자 : 5 숫자 : 9 숫자 : 0

소수입니다. 소수가 아닙니다. 잘못 입력하셨습니다.

▶ 실습문제11

메소드 명 : `public void practice11(){`

2부터 사용자가 입력한 수까지의 소수를 모두 출력하고 소수의 개수를 출력하세요.

단, 입력한 수가 2보다 작은 경우 "잘못 입력하셨습니다."를 출력하세요.

ex.

숫자 : 11

2 3 5 7 11

2부터 11까지 소수의 개수는 5개입니다.

▶ 실습문제12

메소드 명 : `public void practice12(){`

1부터 사용자에게 입력 받은 수까지 중에서

- 1) 2와 3의 배수를 모두 출력하고
- 2) 2와 3의 공배수의 개수를 출력하세요.

* '공배수'는 둘 이상의 수의 공통인 배수라는 뜻으로 어떤 수를 해당 수들로 나눴을 때 모두 나머지가 0이 나오는 수

ex.

자연수 하나를 입력하세요 : 15

2 3 4 6 8 9 10 12 14 15

count : 2