



박수현 연구원

수업 진행방향



학습목표

- 반복문의 필요성을 이해한다.
- 반복문의 종류와 특성을 안다.
- 반복문을 활용하여 간단한 예제를 작성할 수 있다.

-반복문 :

어떤 조건에 만족할 때까지 같은 처리를 반복하여 실행하는 구조

while문

do-while문

for문

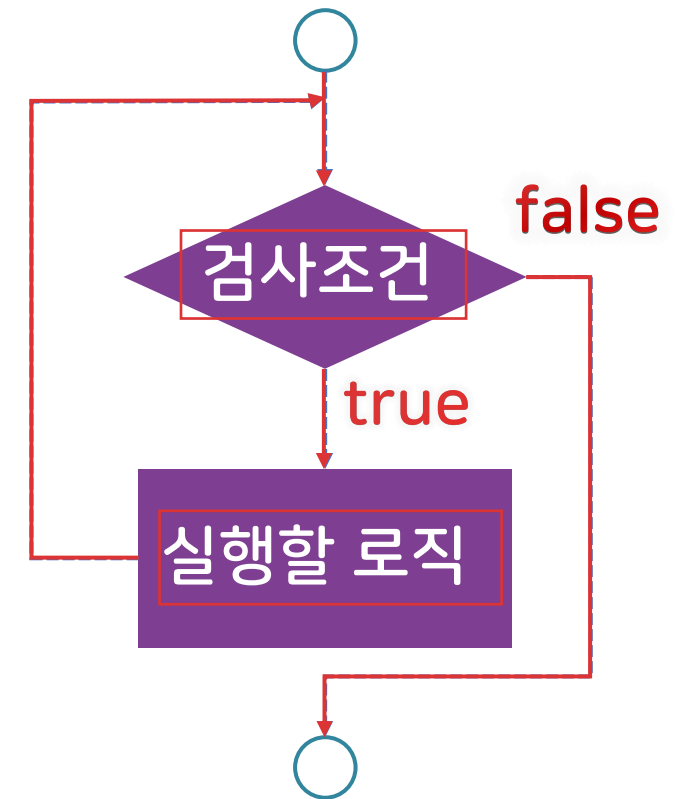
1. while문

- 정확하게 몇 번 반복해야 할 지 정해지지 않은 경우에 사용.

<while문의 구조>

```
while (검사조건) {  
    검사조건이 true일 동안  
    실행될 로직  
}
```

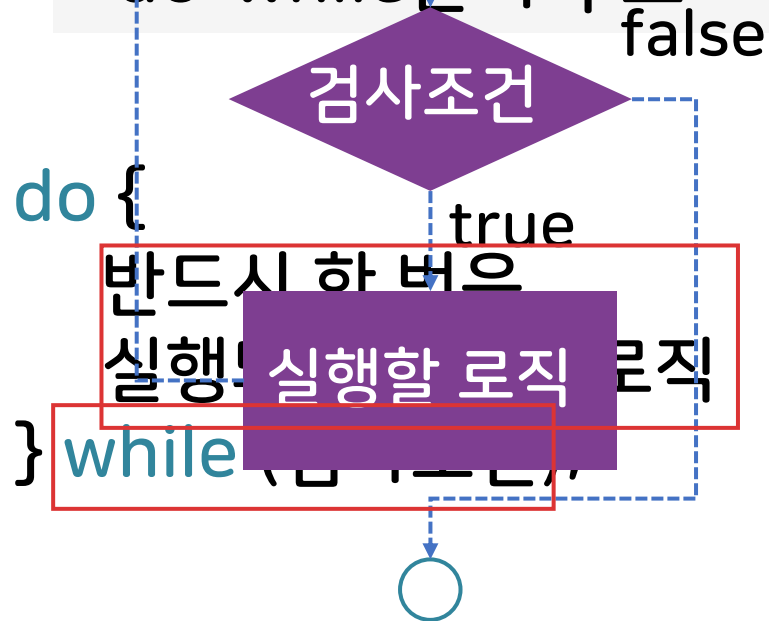
<while문의 흐름도>



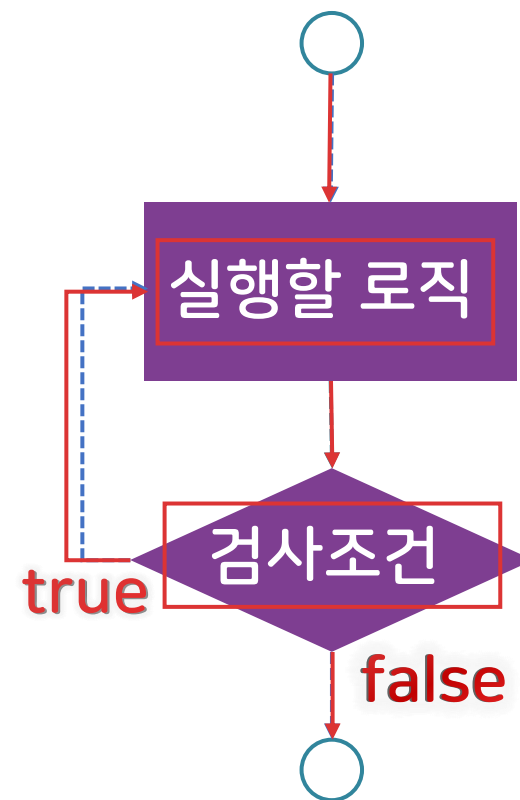
2. <while문의 흐름도>

- 정작이게 몇 번 반복해야 할지 정해지지 않은 경우에 사용.

<do-while문의 구조>



<do-while문의 흐름도>



다이어트 관리 프로그램

1. 현재 몸무게와 목표몸무게를 입력 받고 주차 별 감량 몸무게를 입력 받으세요.
2. 목표몸무게를 달성하면 축하한다는 문구를 출력하고 입력을 멈추세요!

현재몸무게 : 80

목표몸무게 : 70

1주차 감량 몸무게 : 2

2주차 감량 몸무게 : 3

3주차 감량 몸무게 : 4

4주차 감량 몸무게 : 5

66kg 달성!! 축하합니다!

0 부터 100 까지의 점수를 계속 입력받다가 범위를 벗어나는 수가 입력되면 그 이전까지 입력된 자료의 합계와 평균을 출력하는 프로그램을 작성하시오.

숫자를 입력하세요

75

숫자를 입력하세요

25

숫자를 입력하세요

55

숫자를 입력하세요

101

|합계 : 155

평균 : 51.666668

Login프로그램

로그인 프로그램을 만들어 보자.

아이디와 비밀번호를 각각 입력 받고 일치할 경우 "로그인성공!"
일치하지 않은 경우에는 "로그인 실패!"

아이디 : Hello 비밀번호 : 1234

Console

```
<terminated> Login [Java Application] C:\Program File  
아이디를 입력해 주세요 >> bye  
비밀번호를 입력해 주세요 >> 1122  
아이디와 비밀번호가 잘못되었습니다.
```

Console

```
<terminated> Login [Java Application] C:\Program File  
아이디를 입력해 주세요 >> Hello  
비밀번호를 입력해 주세요 >> 4321  
아이디와 비밀번호가 잘못되었습니다.
```

Console

```
<terminated> Login [Java Application] C:\Program File  
아이디를 입력해 주세요 >> Hello  
비밀번호를 입력해 주세요 >> 1234  
로그인 성공!
```

Login프로그램

로그인이 실패했을 경우에 계속 입력
로그인이 성공하면 프로그램 종료

```
<terminated> Login [Java Application] C:\Program F
```

```
아이디를 입력해 주세요 >>Hello
```

```
비밀번호를 입력해 주세요 >>4321
```

```
아이디와 비밀번호가 잘못되었습니다.
```

```
아이디를 입력해 주세요 >>check
```

```
비밀번호를 입력해 주세요 >>1234
```

```
아이디와 비밀번호가 잘못되었습니다.
```

```
아이디를 입력해 주세요 >>Hello
```

```
비밀번호를 입력해 주세요 >>1234
```

```
로그인 성공!
```

Login프로그램

아이디와 비밀번호가 틀렸을 경우 계속 하시겠습니까?
라는 문장을 출력하세요.

Y를 입력하면 아이디 비밀번호 입력 계속,
N을 입력하면 종료/로그인 성공 시 종료

Console

<terminated> Login [Java Application] C:\Program Fi

아이디를 입력해 주세요 >> Hi
비밀번호를 입력해 주세요 >> 1234
아이디와 비밀번호가 잘못되었습니다.
계속 하시겠습니까? (Y/N) >> Y

아이디를 입력해 주세요 >> Hello
비밀번호를 입력해 주세요 >> 4321
아이디와 비밀번호가 잘못되었습니다.
계속 하시겠습니까? (Y/N) >> N

종료되었습니다.

Console

<terminated> Login [Java Application] C:\Program Fil

아이디를 입력해 주세요 >> Hello
비밀번호를 입력해 주세요 >> 8520
아이디와 비밀번호가 잘못되었습니다.
계속 하시겠습니까? (Y/N) >> Y

아이디를 입력해 주세요 >> Hello
비밀번호를 입력해 주세요 >> 1234
로그인 성공!

종료되었습니다.

(+, -) 계산기 프로그램

- 첫 번째 정수를 입력 받는다.
- 두 번째 정수를 입력 받는다.
- 연산자를 선택한다.
- 선택한 연산자에 따라 연산결과를 출력한다.
- 다시 실행할 것인가를 물어본다.
- "Y" 를 입력하면 다시 실행.
- "N"을 입력하면 반복문 종료.

Console

<terminated> Calculator [Java Application] C:\WProgram File

첫 번째 정수를 입력하세요 >> 50

두 번째 정수를 입력하세요 >> 25

[1]더하기 [2]빼기 >> 1

더하기 연산 결과는 75입니다.

다시 실행하시겠습니까? (Y/N) >> Y

첫 번째 정수를 입력하세요 >> 30

두 번째 정수를 입력하세요 >> 46

[1]더하기 [2]빼기 >> 2

빼기 연산 결과는 -16입니다.

다시 실행하시겠습니까? (Y/N) >> N

종료되었습니다.

PlusGame

랜덤으로 정수 2개를 뽑아 아래와 같이 출력
사용자는 두 수의 합을 입력
두 수의 합과 입력한 수가 일치하면 "Success"
두 수의 합과 입력한 수가 일치하지 않으면 "Fail"을 출력

Problems @ Javadoc Declaration
<terminated> plusGame [Java Application]

```
==Plus Game==  
13+17=30  
Success
```

Problems @ Javadoc Declaration
<terminated> plusGame [Java Application]

```
==Plus Game==  
18+7=30  
Fail
```

난수뽑기

```
Random random = new Random();
```

```
int num = random.nextInt(20)+1;
```



0~19까지 난수 발생

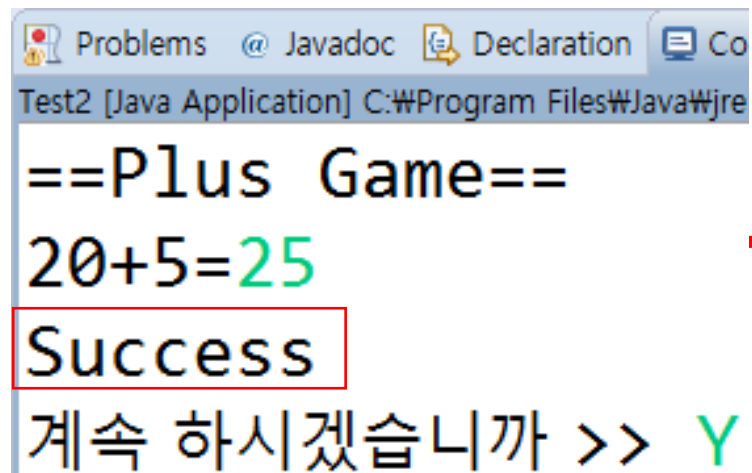
1. 사용자가 원할 때까지 반복
2. Fail 후 계속하기를 선택할 시 기존 숫자를 출력

```
Problems @ Javadoc Declaration Co
Test2 [Java Application] C:\Program Files\Java\jre1
==Plus Game==
4+9=10
Fail
계속 하시겠습니까 >> Y
```

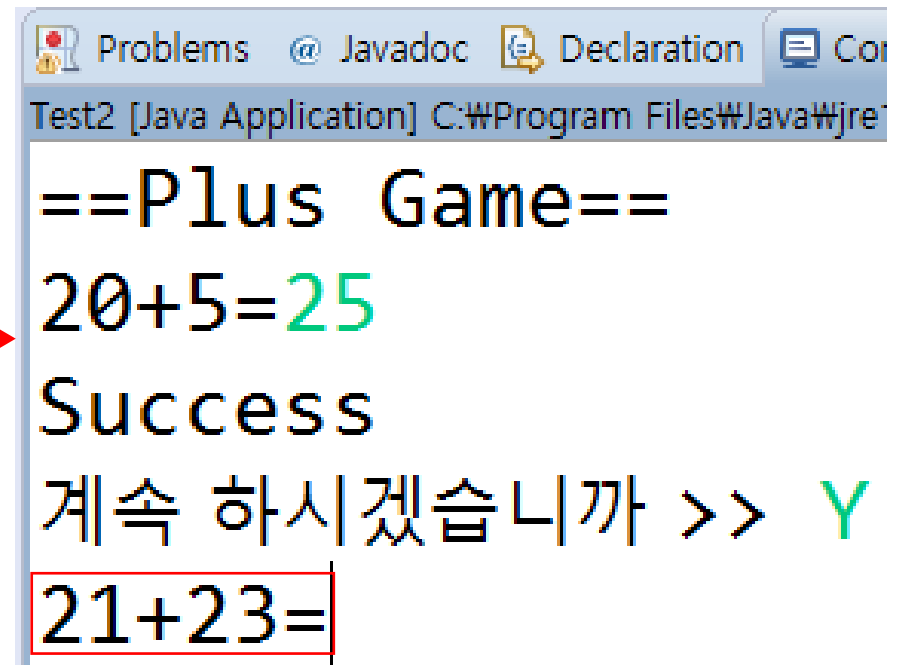


```
Problems @ Javadoc Declaration Cor
Test2 [Java Application] C:\Program Files\Java\jre1
==Plus Game==
4+9=10
Fail
계속 하시겠습니까 >> Y
4+9=13
Success
계속 하시겠습니까 >> Y
6+1=
```

3. Success 후 계속하기를 선택할 시 새로운 숫자를 뽑아서 출력



```
Test2 [Java Application] C:\Program Files\Java\jre...  
==Plus Game==  
20+5=25  
Success  
계속 하시겠습니까 >> Y
```



```
Test2 [Java Application] C:\Program Files\Java\jre...  
==Plus Game==  
20+5=25  
Success  
계속 하시겠습니까 >> Y  
21+23=
```

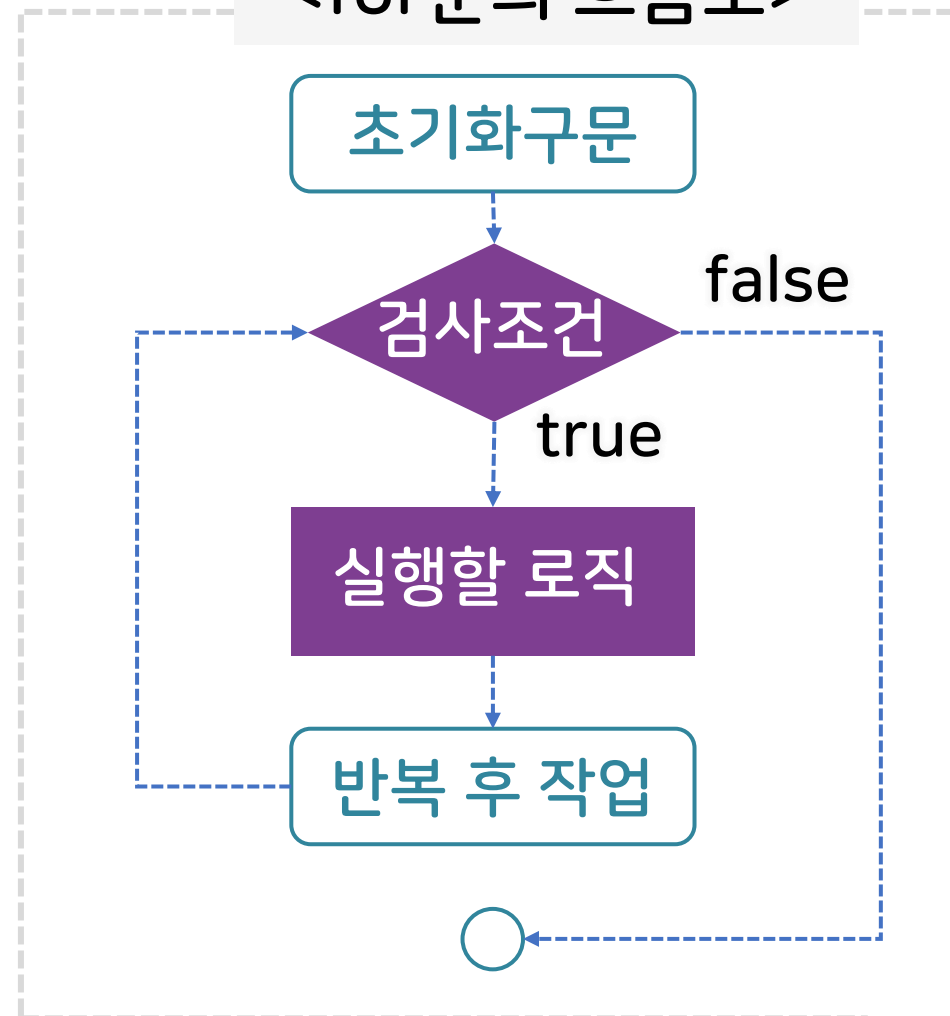

for문

- 주로 반복 횟수가 정해진 경우에 사용함.

<for문의 구조>

```
for (초기화구문; 검사조건; 반복후작업){  
    검사조건이 true일 동안 실행될 로직  
}
```

<for문의 흐름도>



for문을 사용하여 1에서 10까지 출력하시오.

초기화구문 검사조건 반복 후 작업
for (int i = 1 ; i<=10 ; i++) {

System.out.println(i);

실행할 로직
(반복되는 부분)

}

for문을 사용하여 21에서 57까지 출력하시오.

for문을 사용하여 96에서 53까지 출력하시오.

for문을 사용하여 21에서 57까지의 수 중 홀수만 출력하시오.

정수를 입력 받아 1부터 입력 받은 정수까지 차례대로 출력하는 프로그램을 작성하시오.

10

1 2 3 4 5 6 7 8 9 10

100 이하의 두 개의 정수를 입력 받아 작은 수부터 큰 수까지 차례대로 출력하는 프로그램을 작성하시오.

7

5

| 5

6

7

Scanner 클래스를 사용해 숫자 2개를 입력 받아
두 수 사이의 총 합을 출력하는 프로그램을 만들어보자.

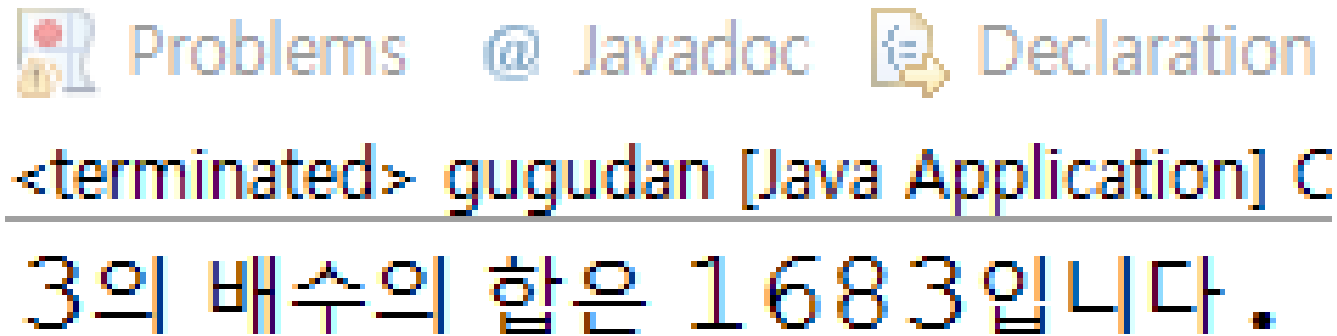
(2와 9를 입력 했을 경우 $2+3+4+5+6+7+8+9$ 의 값을 출력한다.)

1. 1~100까지 3의 배수를 출력하세요.
2. 한 개의 자연수를 입력 받아 그 수의 배수를 차례로 10개 출력하는 프로그램을 작성하시오.

5

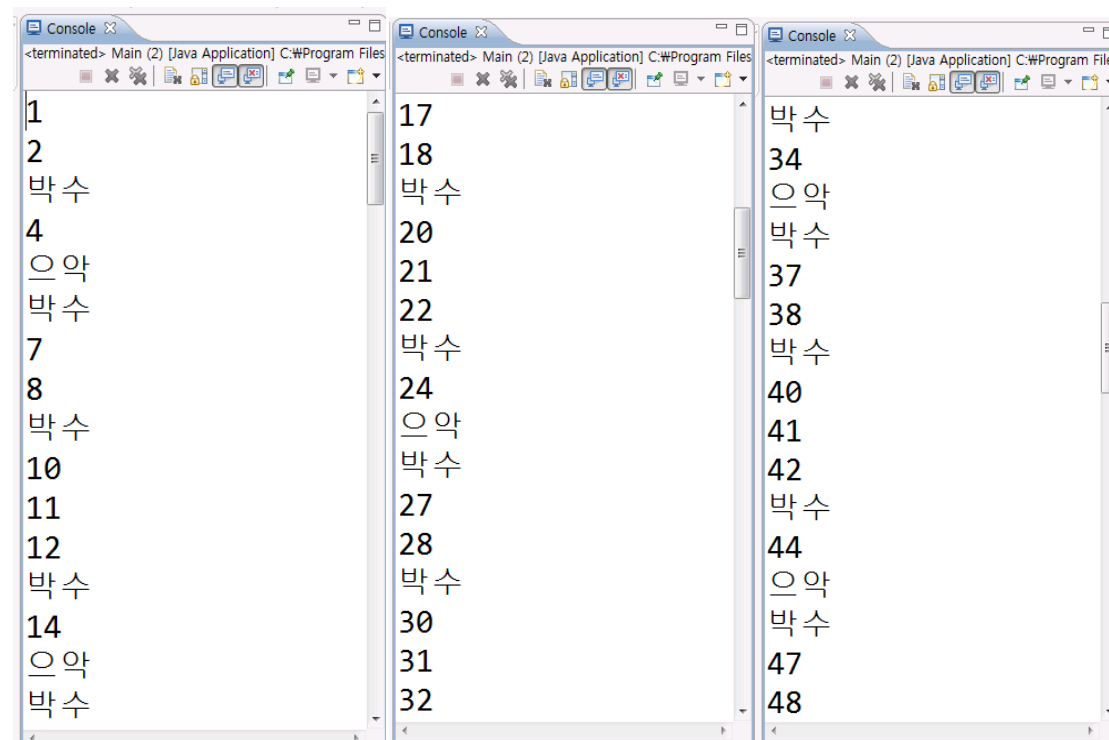
5 10 15 20 25 30 35 40 45 50

1~100까지 3의 배수의 합을 출력하세요.



Problems @ Javadoc Declaration
<terminated> gugudan [Java Application] C
3의 배수의 합은 1683입니다.

- 1부터 50까지 369게임을 출력해 보세요.
- 끝자리 수가 3,6,9일 때는 "박수"를 출력.
 - 끝자리 수가 5일 때는 "으악"을 출력.
 - 그 외의 나머지 수는 그냥 숫자를 출력.



```
<terminated> Main (2) [Java Application] C:\Program Files
1
2
박수
4
으악
박수
7
8
박수
10
11
12
박수
14
으악
박수
17
18
박수
20
21
22
박수
24
으악
박수
27
28
박수
30
31
32
박수
34
으악
박수
37
38
박수
40
41
42
박수
44
으악
박수
47
48
```

for문을 사용하여 1에서 100까지 다음과 같은 순서로 출력하세요.

```
<terminated> Forex01 [Java Application] C:\Program Files\Java\jdk1.8.0_121\bin
1 -2 3 -4 5 -6 7 -8 9 -10 11 -12 13 -14 15 -16 17 -18
```

for문을 사용하여 구구단 2단을 출력하세요.

 Problems @

<terminated> gug

2 * 1 = 2

2 * 2 = 4

2 * 3 = 6

2 * 4 = 8

2 * 5 = 10

2 * 6 = 12

2 * 7 = 14

2 * 8 = 16

2 * 9 = 18

단수와 곱해지길 원하는 수를 입력하여
다음과 같이 출력하세요.

 Problems @ Jav

<terminated> guguda

단 입력 : 4

범위 입력 : 6

4 * 1 = 4

4 * 2 = 8

4 * 3 = 12

4 * 4 = 16

4 * 5 = 20

4 * 6 = 24

구구단 2단~9단까지 출력하세요.

==2 단==

2*1=2

2*2=4

2*3=6

2*4=8

2*5=10

2*6=12

2*7=14

2*8=16

2*9=18

==5 단==

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

==8 단==

8*1=8

8*2=16

8*3=24

8*4=32

8*5=40

8*6=48

8*7=56

8*8=64

8*9=72

==3 단==

3*1=3

3*2=6

3*3=9

3*4=12

3*5=15

3*6=18

3*7=21

3*8=24

3*9=27

==6 단==

6*1=6

6*2=12

6*3=18

6*4=24

6*5=30

6*6=36

6*7=42

6*8=48

6*9=54

==9 단==

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

==4 단==

4*1=4

4*2=8

4*3=12

4*4=16

4*5=20

4*6=24

4*7=28

4*8=32

4*9=36

==7 단==

7*1=7

7*2=14

7*3=21

7*4=28

7*5=35

7*6=42

7*7=49

7*8=56

7*9=63

구구단 2단~9단까지 출력하세요.

Problems @ Javadoc Declaration Console

<terminated> gugudan [Java Application] C:\Program Files\Java\jdk1.8.0_73\bin\javaw.exe (2017. 6. 28. 오후 3:05:49)

2단 :	2*1=2	2*2=4	2*3=6	2*4=8	2*5=10	2*6=12	2*7=14	2*8=16	2*9=18
3단 :	3*1=3	3*2=6	3*3=9	3*4=12	3*5=15	3*6=18	3*7=21	3*8=24	3*9=27
4단 :	4*1=4	4*2=8	4*3=12	4*4=16	4*5=20	4*6=24	4*7=28	4*8=32	4*9=36
5단 :	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
6단 :	6*1=6	6*2=12	6*3=18	6*4=24	6*5=30	6*6=36	6*7=42	6*8=48	6*9=54
7단 :	7*1=7	7*2=14	7*3=21	7*4=28	7*5=35	7*6=42	7*7=49	7*8=56	7*9=63
8단 :	8*1=8	8*2=16	8*3=24	8*4=32	8*5=40	8*6=48	8*7=56	8*8=64	8*9=72
9단 :	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

구구단 2단~9단까지 출력하세요.

Problems @ Javadoc Declaration Console

<terminated> gugudan [Java Application] C:\Program Files\Java\jdk1.8.0_73\bin\javaw.exe (2017. 6. 28. 오후 3:08:42)

2*1=2	3*1=3	4*1=4	5*1=5	6*1=6	7*1=7	8*1=8	9*1=9
2*2=4	3*2=6	4*2=8	5*2=10	6*2=12	7*2=14	8*2=16	9*2=18
2*3=6	3*3=9	4*3=12	5*3=15	6*3=18	7*3=21	8*3=24	9*3=27
2*4=8	3*4=12	4*4=16	5*4=20	6*4=24	7*4=28	8*4=32	9*4=36
2*5=10	3*5=15	4*5=20	5*5=25	6*5=30	7*5=35	8*5=40	9*5=45
2*6=12	3*6=18	4*6=24	5*6=30	6*6=36	7*6=42	8*6=48	9*6=54
2*7=14	3*7=21	4*7=28	5*7=35	6*7=42	7*7=49	8*7=56	9*7=63
2*8=16	3*8=24	4*8=32	5*8=40	6*8=48	7*8=56	8*8=64	9*8=72
2*9=18	3*9=27	4*9=36	5*9=45	6*9=54	7*9=63	8*9=72	9*9=81

다음과 같은 별 모양으로 출력하세요.

```
Problems @ Java  
<terminated> nnnnnn  
*  
  
* *  
  
* * *  
  
* * * *  
  
* * * * *
```


다음과 같은 별 모양으로 출력하세요.

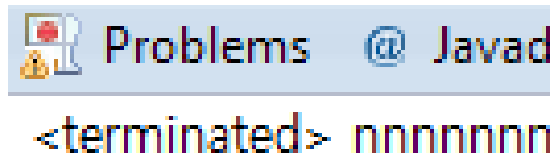
 Problems @ Java

<terminated> nnnnnnn

**

*

다음과 같은 별 모양으로 출력하세요.



```
Problems @ Jav  
<terminated> nnnnnnn
```

*

* *

* * *

* * * *

* * * * *

입력 받은 정수의 약수를 구하세요.


<terminated> gugudan [Java Application] C:\#Progra
정수 입력 : 32
32의 약수 : 1 2 4 8 16 32

2~30까지의 약수를 구하세요.

```
2의 약수 : 1 2
3의 약수 : 1 3
4의 약수 : 1 2 4
5의 약수 : 1 5
6의 약수 : 1 2 3 6
7의 약수 : 1 7
8의 약수 : 1 2 4 8
9의 약수 : 1 3 9
10의 약수 : 1 2 5 10
11의 약수 : 1 11
12의 약수 : 1 2 3 4 6 12
13의 약수 : 1 13
14의 약수 : 1 2 7 14
15의 약수 : 1 3 5 15
16의 약수 : 1 2 4 8 16
17의 약수 : 1 17
18의 약수 : 1 2 3 6 9 18
19의 약수 : 1 19
20의 약수 : 1 2 4 5 10 20
21의 약수 : 1 3 7 21
22의 약수 : 1 2 11 22
23의 약수 : 1 23
24의 약수 : 1 2 3 4 6 8 12 24
25의 약수 : 1 5 25
26의 약수 : 1 2 13 26
27의 약수 : 1 3 9 27
28의 약수 : 1 2 4 7 14 28
29의 약수 : 1 29
30의 약수 : 1 2 3 5 6 10 15 30
```

1~1000까지 숫자 중 완전수인 숫자를 모두 출력하세요.

(완전수 = 자신의 약수 중에서 자신을 제외한 모든 약수의 합이 자신과 같다면 완전수)

ex> - 6의 약수 1, 2, 3, 6

- 자신을 제외한 약수의 합 $1+2+3$
- 자신과 자신을 제외한 약수의 합이 같다
 $1 + 2 + 3 == 6$
- 6은 완전수

 Problems @ Javadoc [

<terminated> Main [Java Ap

6

28

496

while문

반복횟수가 정해지지
않고 조건에 따라
달라지는 경우

```
while (검사조건) {  
    검사조건이 true일 동안  
    실행될 로직  
}
```

do-while문

반복횟수가 정해지지
않고 조건에 따라
달라지는 경우

```
do {  
    반드시 한 번은  
    실행되어야 하는 로직  
} while (검사조건);
```

for문

일정한 반복 횟수가
정해진 경우

```
for (초기화구문; 검사조건; 반복후작업){  
    검사조건이 true일 동안 실행될 로직  
}
```

다음시간에는?

배열