

컴퓨터 프로그래밍1

실습 8주차

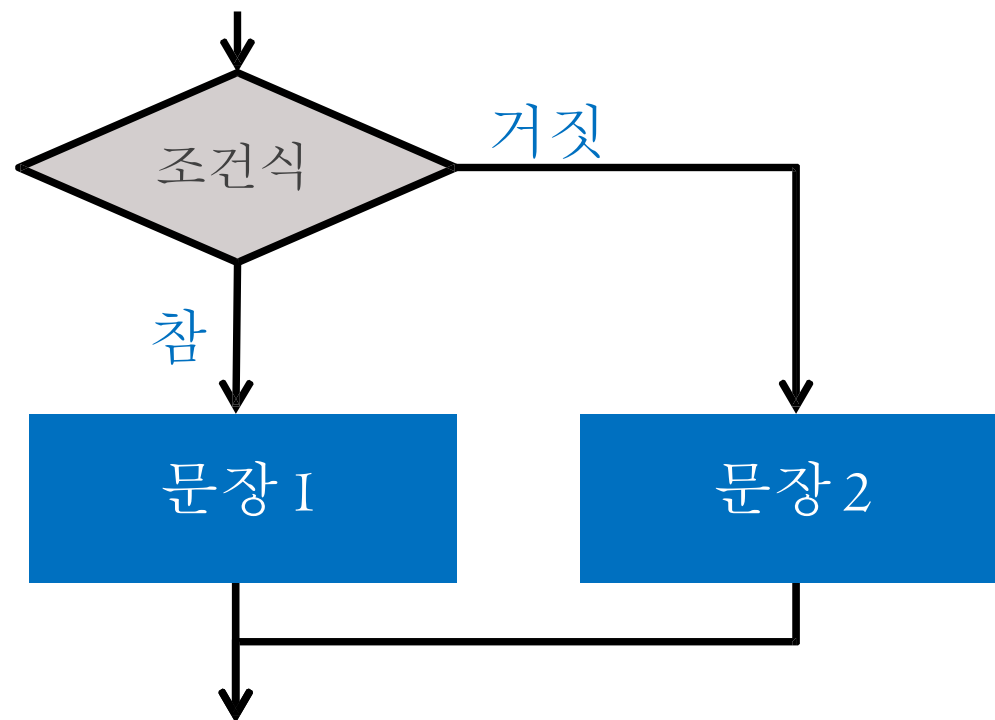
- 조건문 · 반복문 응용 -

조건문 (1/4)

if 문

if-else 문은 다음과 같은 형식을 가짐

```
if(조건식)  
    문장1;  
else  
    문장2;
```



조건식이 참으로 계산되면 문장 I이 실행되고, 거짓으로 계산되면 문장 2가 실행

조건식이 참일 때 실행되는 블록을 then 절, 거짓일 때 실행되는 블록은 else 절

조건문 (2/4)

프로그래밍 예제

```
public class Pay {  
    public static void main(String args[]) {  
        final int RATE = 5000;  
        int pay; // 임금  
        int hours; // 시간  
        Scanner input = new Scanner(System.in);  
  
        System.out.print("시간을 입력하시오: "); // 입력 안내 출력  
        hours = input.nextInt();  
  
        if (hours > 8)  
            pay = RATE * 8 + (int) (1.5 * RATE * (hours - 8));  
        else  
            pay = RATE * hours;  
        System.out.printf("임금은 %d입니다.\n", pay);  
    }  
}
```

시간을 입력하시오: 8
임금은 40000입니다.

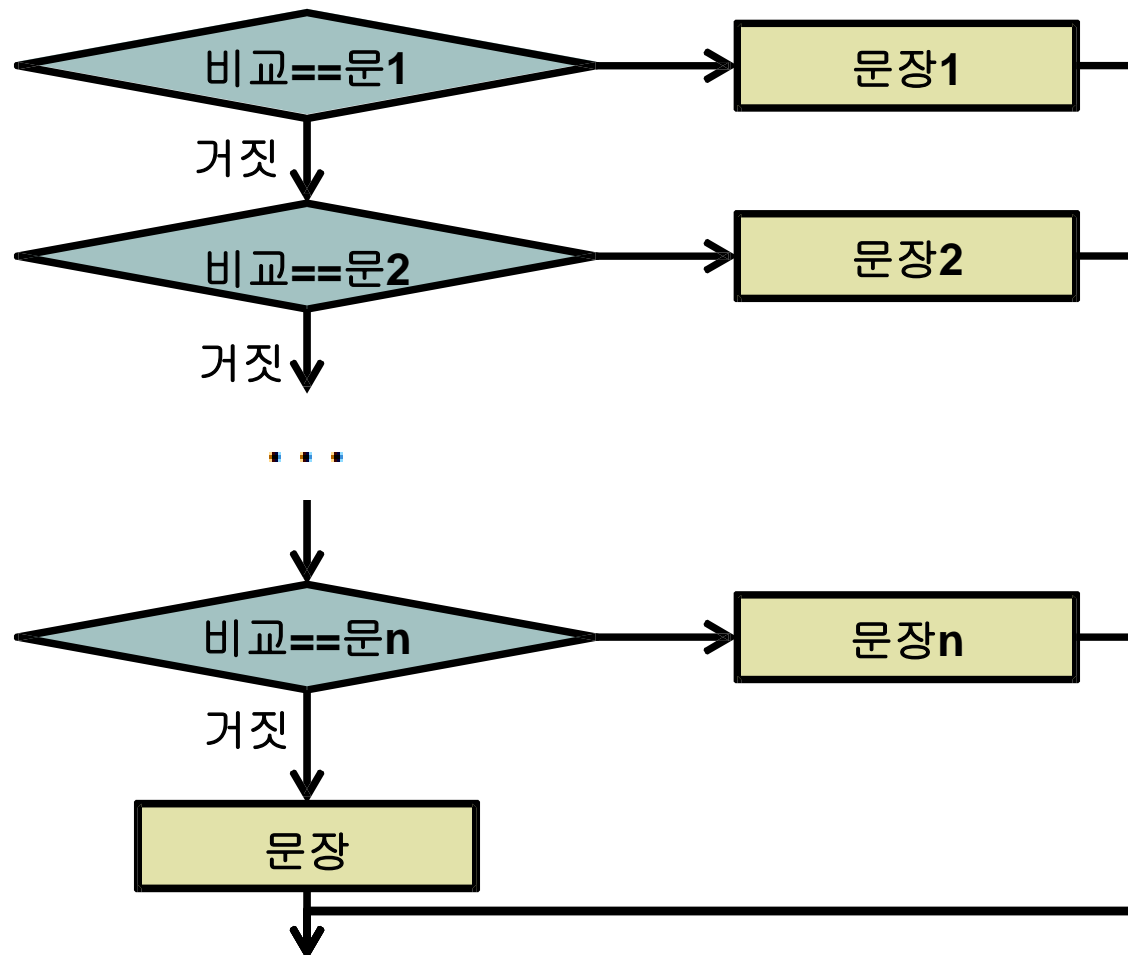
시간을 입력하시오: 9
임금은 47500입니다.

조건문 (3/4)

switch 문

switch 문은 여러 개의 가능한 실행 경로 중에서 하나를 선택하는데 사용됨

```
switch (비교대상) {  
  case 문1:  
    문장1;  
    break;  
  case 문2:  
    문장2;  
    break;  
  ...  
  case 문n:  
    문장n;  
    break;  
  default:  
    문장;  
}
```



조건문 (4/4)

프로그래밍 예제 (p.110)

```
public class SwitchExample {  
    public static void main(String[] args) {  
        int number;  
  
        Scanner scan = new Scanner(System.in);  
        System.out.print("숫자를 입력하시오: ");  
        number = scan.nextInt();  
        switch (number) {  
            → case 0:  
                System.out.println("없음");  
                break;  
            → case 1:  
                System.out.println("하나");  
                → break;  
            case 2:  
                System.out.println("둘");  
                break;  
            default:  
                System.out.println("많음");  
                break;  
        }  
    }  
}
```

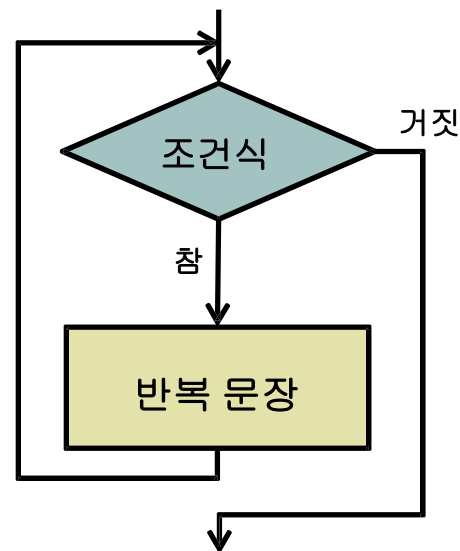
숫자를 입력하시오: 1
하나

반복문 (1/8)

While 문

- while문은 다음과 같은 형식을 가진다.

```
while (조건식) {  
    반복문장  
}
```



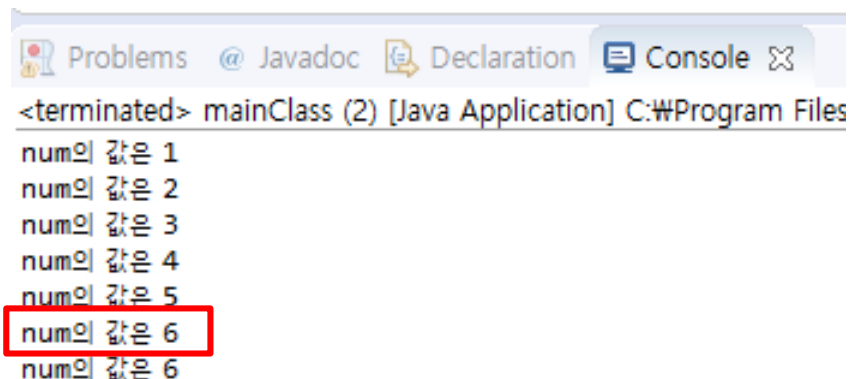
- 조건식이 참이면 문장들의 블록을 반복적으로 실행한다.

반복문 (2/8)

프로그래밍 예제

```
public static void main(String[] args) {  
    int num;  
  
    num = 0;  
  
    while(num < 6){  
        num++;  
        System.out.println("num의 값은 "+num);  
    }  
    System.out.println("num의 값은 "+num);  
}
```

← num이 6보다 작은 경우 반복



Problems @ Javadoc Declaration Console

<terminated> mainClass (2) [Java Application] C:\Program Files

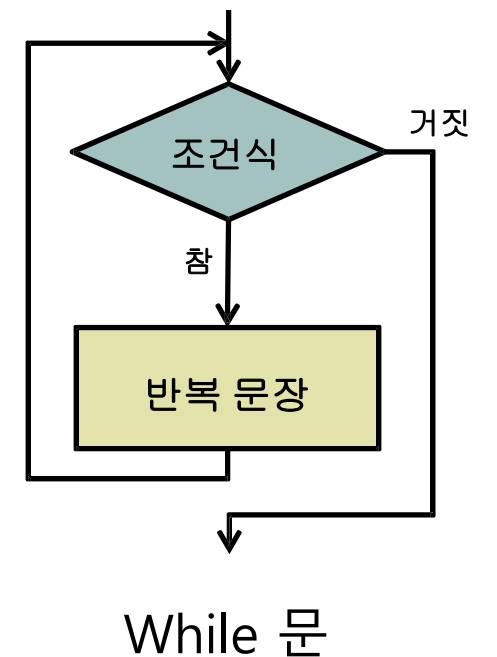
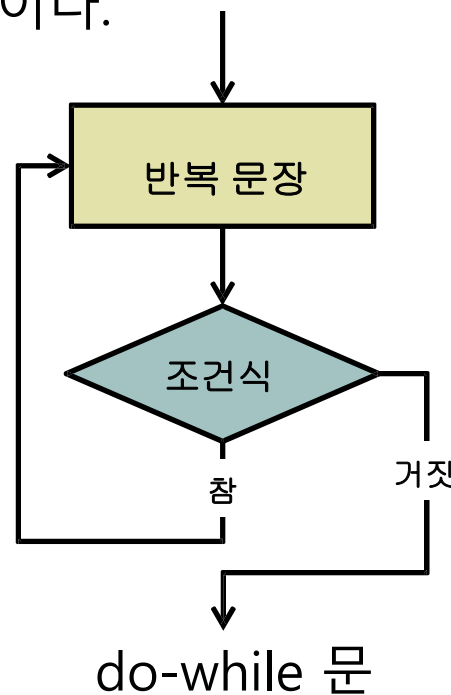
num의 값은 1
num의 값은 2
num의 값은 3
num의 값은 4
num의 값은 5
num의 값은 6
num의 값은 6

반복문 (3/8)

do-while 문

- do-while 문은 while 문과 다르게 반복문장을 먼저 실행하고 조건을 나중에 검사하는 반복 구조이다.

```
do {  
    반복문장;  
} while (조건식);
```



- 즉, do 블록 안에 있는 문장은 적어도 한번은 실행된다.

반복문 (4/8)

프로그래밍 예제

```
public static void main(String[] args) {  
    int answer = 59;  
    int guess;  
    int tries = 0;  
    Scanner in = new Scanner(System.in);
```

```
    do{  
        System.out.println("1 ~ 100 사이 정수를 추측해보세요.");  
        guess = in.nextInt();  
        tries++;  
  
        if(guess > answer){  
            System.out.println("입력한 정수보다 낮습니다.");  
        }else if(guess < answer){  
            System.out.println("입력한 정수보다 높습니다.");  
        }  
    }while(guess != answer);
```

```
    System.out.println("축하합니다. 시도횟수="+tries);  
}
```

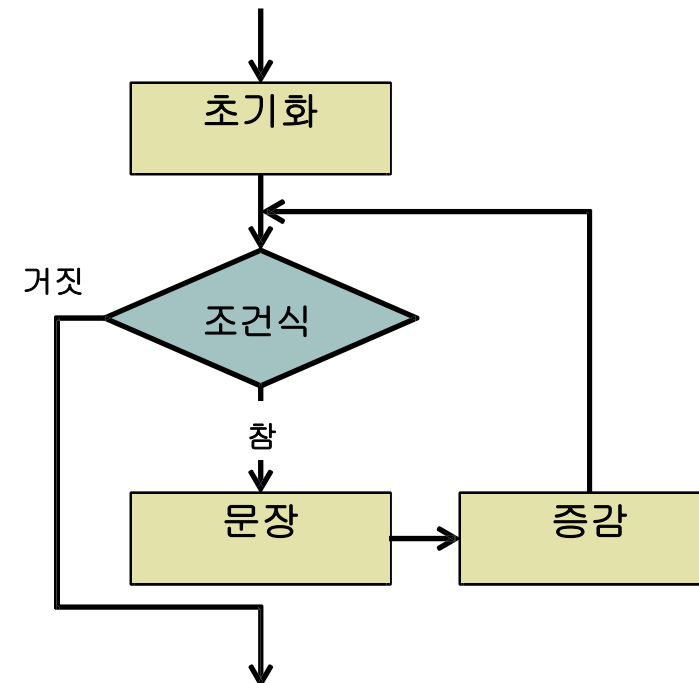
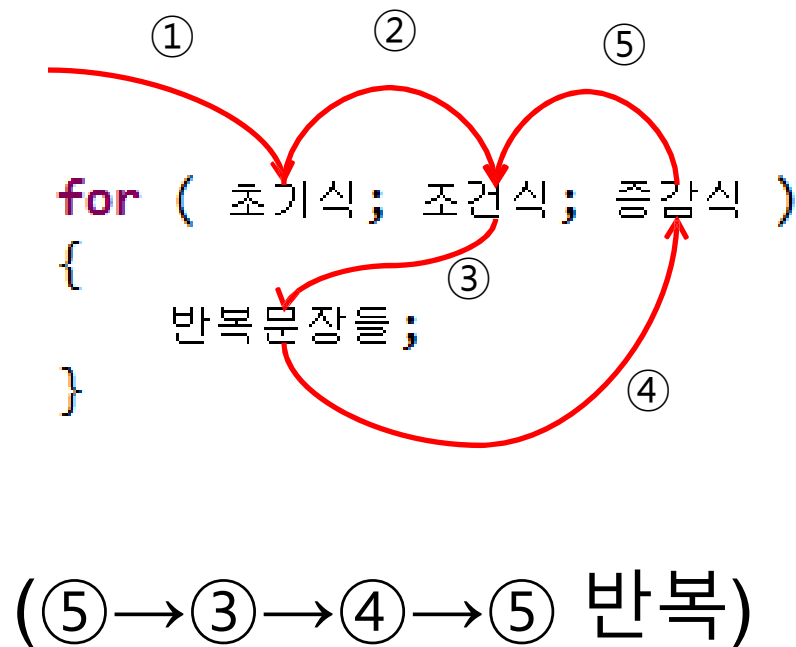
```
Scanner in = new Scanner(System.in);  
1 ~ 100 사이 정수를 추측해보세요.  
72  
입력한 정수보다 낮습니다.  
1 ~ 100 사이 정수를 추측해보세요.  
50  
입력한 정수보다 높습니다.  
1 ~ 100 사이 정수를 추측해보세요.  
59  
축하합니다. 시도횟수=3
```

사용자의 정수를 입력 받아
Answer의 정수와 비교 후
같을 때 까지 반복

반복문 (5/8)

For 문

- for 문은 일정한 횟수만큼 반복할 때 유용하게 사용된다. 흔히 for 루프라고 부른다.
- 초기식 계산 → 조건식 비교 → 반복문 실행 → 증감식 계산 → 조건식 비교 → 반복문 실행(참) → 종료 (거짓)



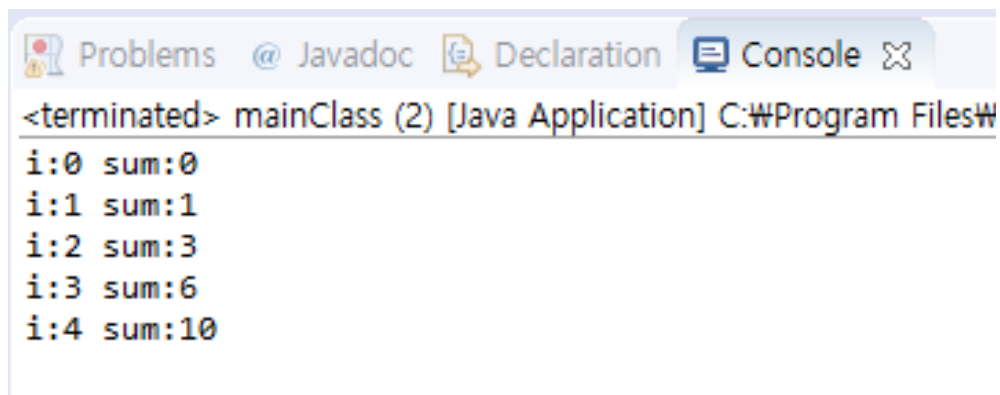
반복문 (6/8)

프로그래밍 예제

```
public static void main(String[] args) {  
    int i;  
    int sum = 0;
```

```
    for(i=0;i<5;i++){  
        sum += i;  
        System.out.println("i:"+i+" sum:"+sum);  
    }
```

i 는 0부터 4까지 증가하며
증가한 i 값을 sum에 합산



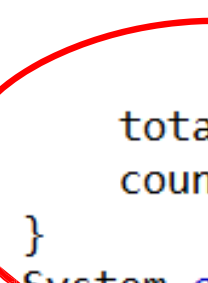
```
<terminated> mainClass (2) [Java Application] C:\Program Files\W  
i:0 sum:0  
i:1 sum:1  
i:2 sum:3  
i:3 sum:6  
i:4 sum:10
```

반복문 (7/8)

Break 문

```
int total = 0;
int count = 0;
Scanner scan = new Scanner(System.in);
while (true) {
    System.out.print("점수를 입력하시오: ");
    int grade = scan.nextInt();
    if (grade < 0)
        break;

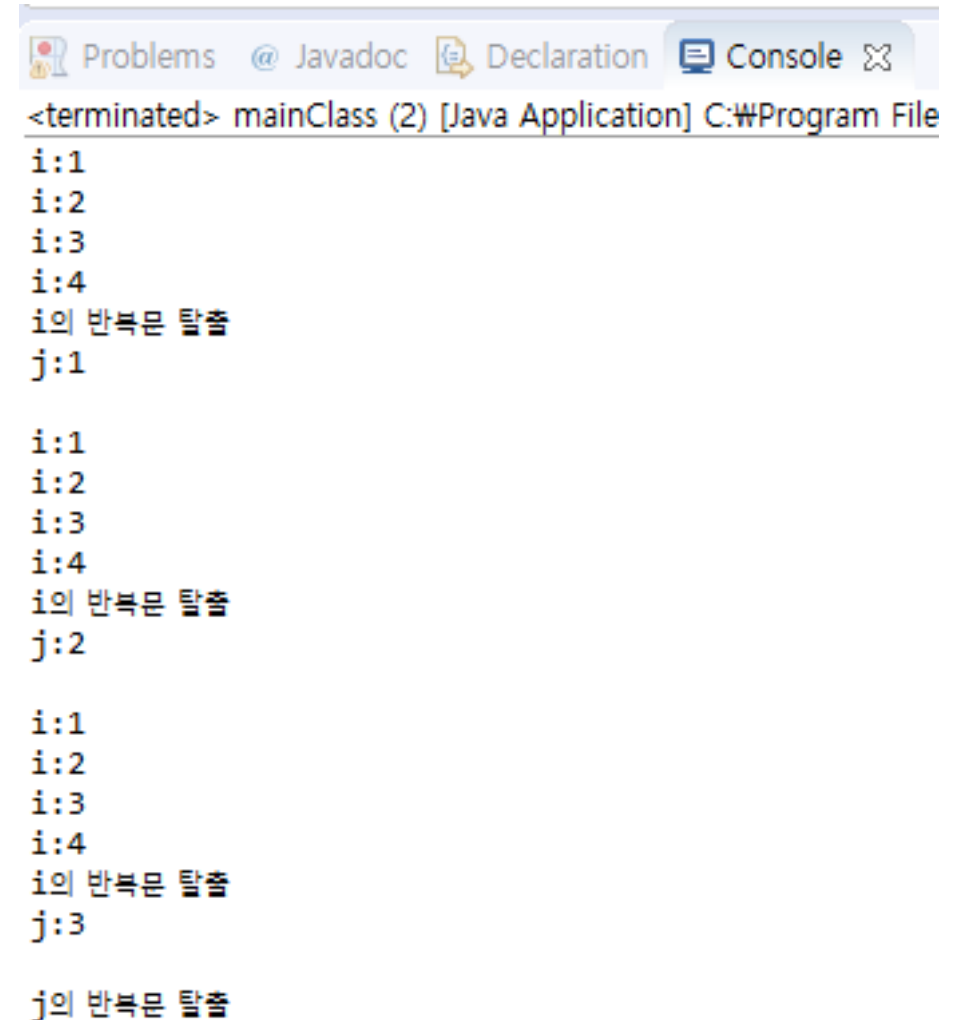
    total += grade;
    count++;
}
System.out.println("평균은 " + total / count);
```



반복문 (8/8)

Break 문

```
public static void main(String[] args) {  
    int i;  
    int j;  
  
    j = 0;  
    while(true){  
        i = 0;  
        while(true){  
            i++;  
            System.out.println("i:"+i);  
            if(i>3)  
                break;  
        }  
        System.out.println("i의 반복문 탈출");  
  
        j++;  
        System.out.println("j:"+j + "\n");  
        if(j>2)  
            break;  
    }  
    System.out.println("j의 반복문 탈출");  
}
```



Problems @ Javadoc Declaration Console

<terminated> mainClass (2) [Java Application] C:\Program File

i:1
i:2
i:3
i:4
i의 반복문 탈출
j:1

i:1
i:2
i:3
i:4
i의 반복문 탈출
j:2

i:1
i:2
i:3
i:4
i의 반복문 탈출
j:3

j의 반복문 탈출

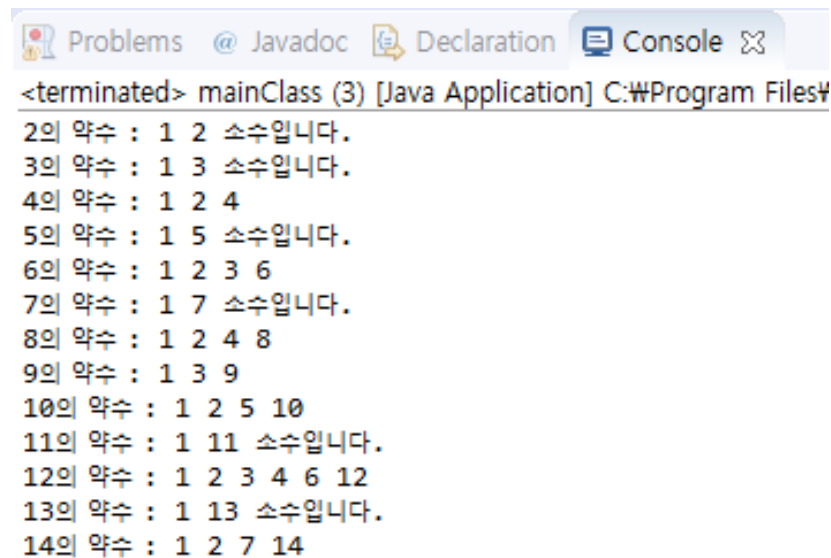
- 반복문에서 break를 만나면 현재 반복하고 있는 한 루프의 밖으로 나감

실습

I부터 100까지 수의 약수 및 소수 찾기

- 반복문과 조건문을 이용하여 1부터 100 사이 수의 약수를 출력하고 소수인 경우 이를 확인할 수 있는 프로그램 작성
- 소수는 1과 자기 자신으로만 나뉘지는 수이다 (ex: 2, 3, 5, 7 ...)
- 힌트 : 반복문은 1~100의 수에 대한 반복문과 현재 비교하려는 수의 약수(나누어지는 수)를 찾는 반복문이 필요하며, 이 약수를 찾기 위한 조건문과 약수의 개수를 이용하여 소수인지를 판별하는 조건문이 필요

실행결과 예시



```
<terminated> mainClass (3) [Java Application] C:\Program Files\
2의 약수 : 1 2 소수입니다.
3의 약수 : 1 3 소수입니다.
4의 약수 : 1 2 4
5의 약수 : 1 5 소수입니다.
6의 약수 : 1 2 3 6
7의 약수 : 1 7 소수입니다.
8의 약수 : 1 2 4 8
9의 약수 : 1 3 9
10의 약수 : 1 2 5 10
11의 약수 : 1 11 소수입니다.
12의 약수 : 1 2 3 4 6 12
13의 약수 : 1 13 소수입니다.
14의 약수 : 1 2 7 14
```

과제

로또 번호 생성 프로그램

1~45의 정수 6개를 중복되지 않게 뽑는 프로그램 작성

(1) Random 클래스를 사용하여 1~45의 임의의 수를 생성

(2) 로또 번호를 저장할 int형 변수 6개를 사용

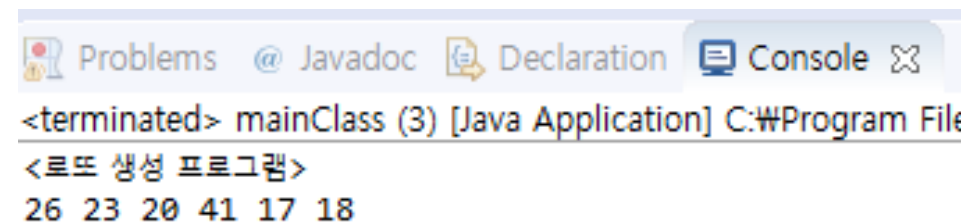
(3) 힌트 : for문, switch문, while문, if문 사용

for : num1~num6의 정수 탐색

switch : for문의 index에 따라 num1 ~ num6에 임의의 값 저장

while, if : 숫자 중복 검사

```
public static void main(String[] args) {  
    Random rand = new Random(System.currentTimeMillis());  
    Scanner in = new Scanner(System.in);  
  
    int num1 = 0;  
    int num2 = 0;  
    int num3 = 0;  
    int num4 = 0;  
    int num5 = 0;  
    int num6 = 0;
```



```
Problems @ Javadoc Declaration Console  
<terminated> mainClass (3) [Java Application] C:\Program File  
<로또 생성 프로그램>  
26 23 20 41 17 18
```

Q&A



수고하셨습니다