



Learn JAVA

A Crash Course Guide
to Learn Java in 1 Week

The image features a red arrow pointing to the right, followed by a white Java logo icon, which consists of a blue coffee cup with a red steam or swirl pattern above it.

1장	화면에 문자 표시하기	9장	간단한 클래스 작성
2장	변수 사용	10장	클래스 변수와 클래스 메서드
3장	프로그램의 흐름 : 분기	11장	패키지
4장	프로그램의 흐름 : 반복	12장	클래스의 상속과 다형성
5장	기본 자료형과 연산	13장	추상 클래스
6장	배열	14장	인터페이스
7장	메서드	15장	문자와 문자열
8장	클래스의 기본	16장	예외 처리

기본 자료형과 연산

5장에선 기본 자료형과 연산에 대해 배운다.



- 정수형과 정수 리터럴
- 부동소수점형과 부동소수점 리터럴
- 논리형(Boolean형)과 논리 리터럴
- 연산과 자료형
- 자료형 변환
- 기본형의 확대 변화, 축소 변화
- 확장 표기
- System.out.printf를 사용한 형식화

문제

5-1

8진수 12, 10진수 12, 16진수 12를 각각 10진수로 표시하는 프로그램을 작성하자.

실행 예

8진수 12는 10진수로 10입니다.

10진수 12는 10진수로 12입니다.

16진수 12는 10진수로 18입니다.

문제

5-2

10진 정수를 8진수와 16진수로 표시하는 프로그램을 작성하자.

실행 예

정수:27

8진수는 33입니다.

16진수는 1b입니다.

문제**5-3**

float형 변수와 double형 변수에 값을 읽어서 표시하는 프로그램을 작성하자.

실행 예

변수 x는 float형입니다. 변수 y는 double형입니다.

x : 0.12345678901234567890

y : 0.12345678901234567890

x = 0.12345679

y = 0.12345678901234568

문제**5-4**

논리형 변수에 true나 false를 대입해서 표시하는 프로그램을 작성하자.

실행 결과

b1 = true
b2 = false

문제

5-5

3개의 정수값을 읽어서 합계와 평균을 구하는 프로그램을 작성하자. 평균을 실수로 표시할 것.

실행 예

정수값 x, y ,z의 평균을 구합니다.

x값:7

y값:8

z값:10

x, y, z의 평균은 8.000입니다.

문제

5-6

3개의 정수값의 합계와 평균을 구하는 프로그램을 작성하자. 평균은 캐스트 연산자를 이용해서 구하고 실수로 표시할 것.

실행 예

정수값 x, y, z의 평균을 구합니다.

x값:7

y값:8

z값:10

x, y, z의 평균은 8.333입니다.

문제**5-7**

int형 변수에 실수값을 대입해서 표시하는 프로그램을 작성하자.

실행 결과

a = 10

문제**5-8**

float형의 변수를 0.0부터 1.0까지 0.001씩 증가시키며 표시하고, 그 옆에 int형 변수를 0부터 1000까지 1씩 증가시켜가며 1000으로 나눈 값을 표시하자.

실행 결과

float	int
0.0000000	0.0000000
0.0010000	0.0010000
...	중략...
0.9989907	0.9990000
0.9999907	1.0000000

문제

5-9

0.0부터 1.0까지 0.001단위로 증가시키면서 해당 값의 제곱을 표시하자.

실행 결과

x x의 제곱

0.000 0.0000000

0.001 0.0000010

0.002 0.0000040

...중략...

0.998 0.9959855

0.999 0.9979824

1.000 0.9999814

문제**5-10**

“ABC\n”을 표시하는 프로그램을 작성하자(큰따옴표 등의 기호 문자도 같이 표시할 것).

실행 결과

"ABC\n"

6장

배열

6장에선 동일한 자료형의 집합인 배열에 대해 배운다.

- 기본형과 참조형
- 배열과 다차원 배열
- 구성 요소와 인덱스
- new 연산자를 사용한 객체 생성
- 배열 변수와 참조, null 참조
- 초기값
- 배열 확인 및 확장용 for문
- 선형 탐색 알고리즘
- 가비지 컬렉션

문제**6-1**

자료형은 double이고 요소 수가 5인 배열을 생성하고 이 배열의 모든 요소를 표시하는 프로그램을 작성하자.

실행 결과

a[0] = 0.0

a[1] = 0.0

a[2] = 0.0

a[3] = 0.0

a[4] = 0.0

문제**6-2**

요소 개수가 5개인 int형 배열에 순서대로 5, 4, 3, 2, 1을 대입하는 프로그램을 작성하자.

실행 결과

a[0] = 5

a[1] = 4

a[2] = 3

a[3] = 2

a[4] = 1

문제**6-3**

요소 개수가 5인 double형 배열을 만들고 앞에서부터 1.1, 2.2, 3.3, 4.4, 5.5를 대입해서 표시하는 프로그램을 작성하자.

실행 결과

```
a[0] = 1.1  
a[1] = 2.2  
a[2] = 3.300000000000003  
a[3] = 4.4  
a[4] = 5.5
```

실행 예

요소 수: 12

```
*  
*  
* *  
* * * * *  
* * * * * * *  
* * * * * * * *  
* * * * * * * * *  
-----
```

0 1 2 3 4 5 6 7 8 9 0 1

문제

6-4

int형 배열의 각 요소에 1~10의 난수를 대입하고 각 요소의 값을 막대그래프(기호 문자 *를 나열)로 표시하는 프로그램을 작성하자. 요소 수는 키보드로 입력한다. 마지막에는 인덱스를 10으로 나눈 나머지를 표시할 것.

문제**6-5**

배열의 요소 수가 5인 int형 배열을 5, 4, 3, 2, 1로 순서대로 초기화해서 표시하는 프로그램을 작성하자.

실행 결과

a[0] = 5

a[1] = 4

a[2] = 3

a[3] = 2

a[4] = 1

문제**6-6**

배열의 요소 수와 개별 요소의 값을 입력 받아서 표시하는 프로그램을 작성하자. 표시 형식은 배열 초기값과 같은 형식으로, 각 요소의 값을 쉼표로 연결하고 {}로 감싼 형태이다.

실행 예

요소 수:3

 $a[0] = 5$ $a[1] = 7$ $a[2] = 8$ $a = \{5, 7, 8\}$

문제

6-7

시험 점수의 합계, 평균, 최고점, 최저점을 구하는 프로그램을 작성하자. 사람 수와 점수는 키보드를 통해 입력 받을 것.

실행 예

사람 수:5
점수를 입력하세요.
1번의 점수:72
2번의 점수:54
3번의 점수:68
4번의 점수:32
5번의 점수:92
합계는 318점입니다.
평균은 63.6점입니다.
최고점은 92점입니다.
최저점은 32점입니다.

문제

6-8

배열에서 특정 값을 가지는 요소를 찾아라. 동일 값의 요소가 여러 개 있으면 가장 앞에 위치한 요소를 찾는 프로그램과 가장 뒤에 있는 요소를 찾는 프로그램을 작성하자.

실행 예

요소 수:7

a[0] = 22

a[1] = 57

a[2] = 11

a[3] = 32

a[4] = 91

a[5] = 32

a[6] = 70

찾을 숫자:32

그 값은 a[3]에 있습니다.

문제

6-9

double형인 배열의 모든 요소의 합과 평균을 구하는 프로그램을 작성하자. 요소 수와 모든 요소의 값은 키보드를 통해 입력받을 것.

실행 예

요소 수:5

a[0] = 5.5

a[1] = 7.2

a[2] = 8.1

a[3] = 3.7

a[4] = 9.3

모든 요소의 합은 33.8입니다.

모든 요소의 평균은 6.76입니다.

문제**6-10**

요소의 형이 int형인 배열을 만들어서 모든 요소에 1~10의 난수를 대입하자(1이상 100이하의 값을 대입). 요소 수는 키보드를 통해 입력받을 것.

실행 예

```
요소 수:7  
a[0] = 8  
a[1] = 8  
a[2] = 2  
a[3] = 3  
a[4] = 2  
a[5] = 7  
a[6] = 2
```

문제**6-11**

인접하는 요소가 동일한 값을 가지지 않도록 [문제 6-10]의 프로그램을 수정하자. 예를 들어 {1, 3, 5, 5, 3, 2}가 돼서는 안 된다.

실행 예

요소 수:7

a[0] = 5

a[1] = 7

a[2] = 5

a[3] = 9

a[4] = 3

a[5] = 10

a[6] = 2

문제**6-12**

서로 다른 요소가 동일한 값을 지니지 않도록 [문제 6-10]의 프로그램을 수정하자. 예를 들어 {1, 3, 5, 6, 1, 2}가 되지 않아야 한다(배열의 요소 수는 100이하일 것).

실행 예

요소 수:7

a[0] = 9

a[1] = 10

a[2] = 2

a[3] = 7

a[4] = 8

a[5] = 6

a[6] = 3

문제**6-13**

배열의 나열 순서를 섞는(임의의 순서가 되도록 순서를 바꿈) 프로그램을 작성하자.

실행 예

요소 수:7

a[0] = 22

a[1] = 57

a[2] = 11

a[3] = 32

a[4] = 91

a[5] = 68

a[6] = 70

요소를 섞었습니다.

a[0] = 22

a[1] = 32

a[2] = 68

a[3] = 11

a[4] = 70

a[5] = 57

a[6] = 91

문제

6-14

배열 a의 모든 요소를 역순으로 배열 b에 복사하는 프로그램을 작성하자. 두 배열의 요소 수는 동일하다고 가정해도 좋다.

실행 예

요소 수: 5

a[0] = 11

a[1] = 12

a[2] = 13

a[3] = 14

a[4] = 15

a의 모든 요소를 역순으로 복사했습니다.

b[0] = 15

b[1] = 14

b[2] = 13

b[3] = 12

b[4] = 11

문제

6-15

월을 1~12의 숫자로 표시하고 해당 월의 영어 단어를 입력하는 프로그램을 작성하자.

- 월은 난수로 생성할 것
- 학습자가 원하는 동안은 계속 반복해서 문제를 낼 것
- 동일 월을 연속해서 표시하지 말 것

문자열 s1과 s2가 같은지 판정(문든 문자가 동일)하는 것은 s1.equals(s2)를 사용한다.

실행 예

해당 월의 영어 단어를 입력하시오.

첫 글자는 대문자, 나머지는 소문자로 입력하세요.

12월:December

정답입니다. 다시 한번? 1...Yes/0...No : 1

10월:March

틀렸습니다.

10월:October

정답입니다. 다시 한번? 1...Yes/0...No : 0

문제

6-16

요일을 표시하고 해당 요일의 영어 단어를 입력하는 영어 학습 프로그램을 작성하자.

- 요일은 난수를 사용해서 생성할 것
- 학습자가 원하는 만큼 반복할 것
- 동일 요일을 연속해서 표시하지 말 것

실행 예

요일명을 영어 소문자로 입력하시오.

일요일 : sunday

정답입니다. 다시 한번? 1...Yes/0...No : 1

수요일 : wednesday

정답입니다. 다시 한번? 1...Yes/0...No : 0

문제

6-17

배열 변수의 값을 표시하는 프로그램을 작성하자

실행 예

```
a = [I@7852e922  
a = null
```

문제

6-18

4행 3열의 행렬과 3행 4열의 행렬을 곱하는 프로그램을 작성하자. 각 요소의 값은 키보드를 통해 입력할 것.

실행 예

행렬 a의 요소를 입력하세요.

a[0][0] : 1

a[0][1] : 2

a[0][2] : 3

a[1][0] : 2

a[1][1] : 3

a[1][2] : 1

a[2][0] : 3

a[2][1] : 1

a[2][2] : 2

a[3][0] : 2

a[3][1] : 1

a[3][2] : 3

행렬 b의 요소를 입력하세요.

b[0][0] : 4

b[0][1] : 3

b[0][2] : 2

b[0][3] : 1

b[1][0] : 3

b[1][1] : 2

b[1][2] : 1

b[1][3] : 4

b[2][0] : 2

b[2][1] : 1

b[2][2] : 4

b[2][3] : 3

행렬 a와 b의 곱

16 10 16 18

19 13 11 17

19 13 15 13

17 11 17 15

문제

6-19

6명의 두 과목 점수(국어, 수학)를 읽어 과목별 평균과 학생별 평균을 구하자.

실행 예

6명의 국어, 수학 점수를
입력하세요.

1번 국어:72

수학:80

2번 국어:35

수학:58

3번 국어:82

수학:91

4번 국어:72

수학:45

5번 국어:56

수학:35

6번 국어:89

수학:75

No.	국어	수학	평균
1	72	80	76.0
2	35	58	46.5
3	82	91	86.5
4	72	45	58.5
5	56	35	45.5
6	89	75	82.0
평균		67.7	64.0

문제

6-20

행에 따라 열의 개수가 다른 2차원 배열을 생성하자. 행 수, 열 수, 각 요소의 값은 키보드를 통해 입력할 것.

각 요소의 값을 입력하세요.

c[0][0]:1

c[0][1]:2

c[0][2]:3

c[0][3]:4

c[0][4]:5

c[1][0]:6

c[1][1]:7

c[1][2]:8

c[2][0]:9

c[2][1]:10

c[2][2]:11

c[2][3]:12

실행 예

불규칙한 2차원 배열을 생성합니다.

행 수:3

0행째의 열 수:5

1행째의 열 수:3

2행째의 열 수:4

각 요소의 값을 입력하세요.

배열c의 각 요솟값은 다음과 같습니다.

1 2 3 4 5

6 7 8

9 10 11 12

문제

6-21

학급 수, 각 학급의 학생 수, 그리고 모든 학생의 점수를 읽어서 합계와 평균을 구하는 프로그램을 작성하자. 합계와 평균은 학급 단위와 전교생을 대상으로 하는 것을 각각 구할 것.

실행 예		
학급 수 : 2		
1반의 학생 수 : 3		
1반1번의 점수 : 50		
1반2번의 점수 : 63		
1반3번의 점수 : 72		
2반의 학생 수 : 2		
2반1번의 점수 : 79		
2반2번의 점수 : 43		
반 합계 평균		
-----+		
1반 185	61.7	
2반 122	61.0	
-----+		
합 307	61.4	

· 메서드

· 인수/매개 변수와 값 전달

· 반환값과 return문

· 배열의 값 교환

· void

· 유효 범위

· 다중 정의

· 비트 단위의 논리 연산

· 비트의 시프트

7장

메서드

7장에선 프로그램을 구성하는 부품 중 하나인

메서드의 개념과 작성법, 사용법 등을 배운다.

문제

7-1

입력한 int형 정수값이 음수이면 -1을, 0이면 0을, 양수이면 1을 반환하는 signOf 메서드를 작성하자.

실행 예1

정수 x : 13
signOf(x)는 1입니다.

실행 예2

정수 x : 0
signOf(x)는 0입니다.

실행 예3

정수 x : -15
signOf(x)는 -1입니다.

문제**7-2**

3개의 int형 인수 a, b, c 중 최솟값을 구하는 min 메서드를 작성하자.

```
int min(int a, int b, int c)
```

실행 예

정수 a: 1

정수 b: 3

정수 c: 2

최솟값은 1입니다.

**Do your
homework!**

문제

7-3

3개의 int형 인수 a, b, c에서 중간값을 구하는 med 메서드를 작성하자.

`int med(int a, int b, int c)`

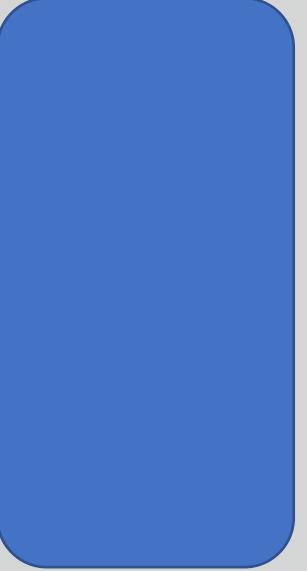
실행 예

정수 a: 1

정수 b: 3

정수 c: 2

중간값은 2입니다.

```
// 3개의 정수값에서 중간값 구하기  
import java.util.Scanner;  
  
class Med3 {  
    //--- a, b, c의 중간값 반환---//  
    static int med3(int a, int b, int c) {  
          
        return b;  
    }  
}
```

```
public static void main(String[] args) {  
    Scanner stdIn = new Scanner(System.in);  
  
    System.out.print("정수 a:"); int a = stdIn.nextInt();  
    System.out.print("정수 b:"); int b = stdIn.nextInt();  
    System.out.print("정수 c:"); int c = stdIn.nextInt();  
  

```

```
}
```

**Do your
homework!**

문제

7-4

1부터 n까지의 정수의 합을 구해서 반환하는 메서드를 작성하자.

```
int sumUp(int n)
```

실행 예

1부터 x까지의 합을 구하자.

x의 값: 5

1부터 5까지의 합은 15입니다.

```
//1부터 n까지의 합 구하기
import java.util.Scanner;

class SumUp1 {
    //--- 1부터 n까지의 합 ---//
    static int sumUp(int n) {
        int sum = 0;
        for (int i = 1; i <= n; i++) {
            sum += i;
        }
        return sum;
    }

    public static void main(String[] args) {
        Scanner stdIn = new Scanner(System.in);

        System.out.println("1부터 x까지의 합을 구하자.");
        int x = stdIn.nextInt();

        System.out.print("1부터 " + x + "까지의 합은 " + sumUp(x) + "입니다.");
    }
}
```

**Do your
homework!**

문제

7-5

“안녕하세요.”라고 표시하는 hello 메서드를 작성하자.

```
void hello()
```

실행 결과

안녕하세요.
안녕하세요.
안녕하세요.

문제

7-6

인수 m에 지정한 달(月)의 계절을 표시하는 printSeason 메서드를 작성하자. m값에 따라 봄(3, 4, 5), 여름(6, 7, 8), 가을(9, 10, 11), 겨울(12, 1, 2) 표시하고 그 이외의 값이 오면 아무것도 표시하지 않는다.

```
void printSeason(int m)
```

실행 예

몇 월입니까(1 ~ 12):4

해당 월의 계절은 봄입니다.

문제

7-7

문자 c를 n개 표시하는 putChar 메서드와 이 메서드를 내부에서 호출해서 문자 '*'를 n개 연속으로 표시하는 putStart 메서드를 작성하자. 또한, 이 메서드들을 사용해서 직삼각형을 만드는 프로그램을 작성하자.

실행 예

왼쪽 아래가 직각인 이등변 삼각형을 표시합니다.
단수는:6

```
*  
**  
***  
****  
*****  
*****
```

문제

7-8

난수($a \leq$ 난수 $\leq b$)를 생성해서 반환하는 `random` 메서드를 작성하자. 메서드 안에서 난수를 생성하는 표준 라이브러리를 호출할 것. (참고, $b \leq a$ 인 경우 a 값을 그대로 반환할 것)

```
int random(int a, int b)
```

실행 예

난수를 생성합니다.

하한값: 10

상한값: 99

생성한 난수는 52입니다.

**Do your
homework!**

문제

7-9

“양의 정수값 : ”이라는 메세지에 정수값을 입력하면 값을 거꾸로 반화하는 `readPlusInt`를 메서드를 작성하자. 0이나 음수(-)가 입력되면 재입력하도록 할 것.

```
int readPlusInt()
```

실행 예

양의 정수값:-125

양의 정수값:0

양의 정수값:521

반대로 읽으면 125입니다.

다시 한 번?<Yes...1/No...0>:

```
// 양의 정수값을 읽어서 뒤에서부터 읽어서 표시
import java.util.Scanner;

class InverseNumber {
    static Scanner stdIn= new Scanner(System.in); •-----①

    //--- 양의 정수값을 읽어서 반환---//
    static int readPlusInt() {
        
        return x;
    }

    public static void main(String[] args) {
        
        System.out.println("입니다.");
    }

    do {
        System.out.print("다시 한 번?<Yes...1/No...0>:");
        x= stdIn.nextInt();
    } while (x!=0 && x!=1);
    } while (x == 1);
}
}
```

문제**7-10**

다음 4개의 계산 문제 중 하나를 무작위로 출제하는 암산 훈련 프로그램을 작성하자. 이때 x, y, z에 3자리의 정수값을 난수로 생성할 것.

$x + y + z$

$x + y - z$

$x - y + z$

$x - y - z$

실행 예

암산 훈련!!

$341 + 616 - 742 = 215$

틀렸습니다!

$341 + 616 - 742 = 216$

다시 한 번?<Yes...1/No...0>:1

$674 + 977 + 760 = 2411$

다시 한 번?<Yes...1/No...0>:0

문제**7-15**

배열 a0이 가진 모든 요소의 합을 구하는 sumOf() 메서드를 작성하자.

```
int sumOf(int[] a)
```

실행 예

요소 수:5

x[0]:22

x[1]:5

x[2]:11

x[3]:32

x[4]:120

모든 요소의 합은 190입니다.

Do your homework!

문제

7-16

배열 a의 요소 중에서 최솟값을 구하는 minOf 메서드를 작성하자.

```
int minOf(int[] a)
```

실행 예

사람 수는: 4
4명의 신장과 체중을 입력하자.

1번의 신장: 175
1번의 체중: 72
2번의 신장: 163
2번의 체중: 82
3번의 신장: 150
3번의 체중: 49
4번의 신장: 181
4번의 체중: 76
가장 키가 작은 사람의 신장: 150cm
가장 마른 사람의 체중: 49kg

```
// 가장 키가 작은 사람의 신장과 가장 마른 사람의 체중을 구한다
import java.util.Scanner;

class minOfHeightWeight {
    //--- 배열 a의 최솟값을 구해서 반환 ---//
    static int minOf(int[] a) {
        int min = a[0];
        for (int i = 1; i < a.length; i++) {
            if (a[i] < min)
                min = a[i];
        }
        return min;
    }

    public static void main(String[] args) {
        Scanner stdIn = new Scanner(System.in);

        System.out.print("사람 수는:");
        int num = stdIn.nextInt();
        System.out.println("명의 신장과 체중을 입력하세요.");
        int minHeight = minOf(height);
        int minWeight = minOf(weight);
        System.out.println("가장 키가 작은 사람의 신장은 " + minHeight + "cm입니다.");
        System.out.println("가장 마른 사람의 체중은 " + minWeight + "kg입니다.");
    }
}
```

```
        int[] height = new int[num];      // 신장용 배열 생성
        int[] weight = new int[num];     // 체중용 배열 생성

        System.out.println(num + "명의 신장과 체중을 입력하자.");
        for (int i = 0; i < num; i++) {
            System.out.print((i + 1) + "번의 신장:");
            height[i] = stdIn.nextInt();
            System.out.print((i + 1) + "번의 체중:");
            weight[i] = stdIn.nextInt();
        }

        System.out.println("가장 키가 작은 사람의 신장은 " + minHeight + "cm입니다.");
        System.out.println("가장 마른 사람의 체중은 " + minWeight + "kg입니다.");
    }
}
```

문제

7-17

배열 a로부터 key와 같은 값을 가지는 요소를 탐색하는 linearSearch 메서드와 linearSearchR 메서드를 작성하자. 단, 키와 같은 값을 가지는 요소가 여러 개인 경우 linearSearch는 가장 앞에 위치한 요소를 찾으며, linearSearchR은 가장 뒤에 위치한 요소를 찾을 것.

```
int linearSearch(int[] a, int key)
```

```
int linearSearchR(int[] a, int key)
```

실행 예

요소 수: 6

x[0]: 5

x[1]: 22

x[2]: 74

x[3]: 32

x[4]: 120

x[5]: 22

찾는 값: 22

해당 값의 요소가 여러 개 존재합니다.

가장 앞에 위치한 값은 x[1]에 있습니다.

가장 뒤에 위치한 값은 x[5]에 있습니다.

문제

7-18

Do your
homework!

배열 a로부터 요소 a[idx]를 삭제하는 aryRmv 메서드를 작성하자.

```
void aryRmv(int[] a, int idx)
```

a[idx]의 삭제는, 그 뒤에 있는 요소들을 앞으로 하나씩 이동해서 할 것. 이동한 후에 비게 되는 마지막 요소 a[a.length - 1]의 값은 이동하기 전의 마지막 값을 유지할 것.

예) 배열 a의 요소가 {1, 3, 4, 7, 9, 11}일 때에 aryRmv(a, 2)라고 호출한 후에는 배열 a의 요소는 {1, 3, 7, 9, 11, 11}이 된다.

실행 예

요소 수:6

a[0]:1

a[1]:3

a[2]:4

a[3]:7

a[4]:9

a[5]:11

삭제할 요소의 인덱스:2

a[0] = 1

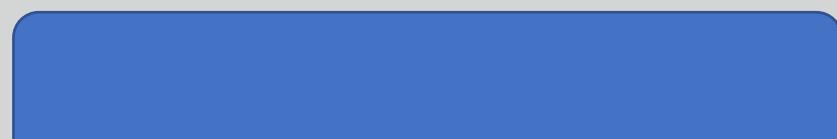
a[1] = 3

a[2] = 7

a[3] = 9

a[4] = 11

a[5] = 11

```
// 배열의 요소 삭제  
import java.util.Scanner;  
  
class AryRmv {  
    //--- 배열 a에서 a[idx]를 삭제(뒤에 있는 요소들을 하나씩 앞으로 이동  
    static void aryRmv(int[] a, int idx) {  
          
    }  
    }  
  
    public static void main(String[] args) {  
        Scanner stdIn = new Scanner(System.in);  
  
        System.out.print("요소 수:");  
        int num = stdIn.nextInt();  
        int[] a = new int[num];          // 요소 수가 num인 배열  
          
        System.out.print("삭제할 요소의 인덱스:");  
        int idx = stdIn.nextInt();  
          
    }  
}
```

문제

7-19

배열 a에서 요소 a[idx]부터 n개의 요소를 삭제하는 aryRmvN 메서드를 작성하자.

```
void aryRmv(int[] a, int idx, int n)
```

삭제는 a[idx]보다 뒤에 있는 모든 요소를 하나씩 앞으로 이동해서 할 것. 이동 대상이 아닌 요소는 기존 값을 그대로 유지할 것.

예) 배열 a의 요소가 {1, 3, 4, 7, 9, 11}일 때 aryRmvN(a, 1, 3)라고 호출하면 배열 a의 요소는 {1, 9, 11, 7, 9, 11}이 된다.

실행 예	삭제를 시작할 인덱스:1	삭제할 요소의 수:3
요소 수:6	a[0] = 1	
a[1]:1	a[1] = 9	
a[2]:3	a[2] = 11	
a[3]:4	a[3] = 7	
a[4]:7	a[4] = 9	
a[5]:9	a[5] = 11	
a[6]:11		

문제

7-20

배열 a의 요소 a[idx]에 x를 삽입하는 aryIns 메서드를 작성하자.

```
void aryIns(int[] a, int idx, int x)
```

삽입 시에는 a[idx] ~ a[a.length-2]를 하나씩 뒤로 이동시켜야 한다.

예) 배열 a의 요소가 {1, 3, 4, 7, 9, 11}일 때에 aryIns(a, 2, 99)라고 호출하면 a의 요소가 {1, 3, 99, 4, 7, 9}가 된다.

실행 예	
요소 수:	6
a[0]:	1
a[1]:	3
a[2]:	4
a[3]:	7
a[4]:	9
a[5]:	11

삽입할 요소의 인덱스: 2

삽입할 값: 99

a[0] = 1

a[1] = 3

a[2] = 99

a[3] = 4

a[4] = 7

a[5] = 9

실행 예

배열 a의 요소 수:7

a[0]:1

a[1]:2

a[2]:3

a[3]:4

a[4]:5

a[5]:6

a[6]:7

배열 b의 요소 수:5

b[0]:5

b[1]:4

b[2]:3

b[3]:2

b[4]:1

배열 a와 b의 전체 요소를 교환했습니다.

a[0] = 5

a[1] = 4

a[2] = 3

a[3] = 2

a[4] = 1

a[5] = 6

a[6] = 7

b[0] = 1

b[1] = 2

b[2] = 3

b[3] = 4

b[4] = 5

문제

7-21

배열 a와 배열 b의 전체 요솟값을 교환하는 aryExchng 메서드를 작성하자.

```
void aryExchng(int[] a, int[] b)
```

두 배열의 요소 수가 같지 않다면 작은 쪽의 배열 수에 맞추어 교환할 것.

예) 배열 a의 요소가 {1, 2, 3, 4, 5, 6, 7}이고 배열 b의 요소가 {5, 4, 3, 2, 1}일 때에, aryExchange(a, b)를 호출하면 배열 a는 {5, 4, 3, 2, 1, 6, 7}이 되고 배열 b는 {1, 2, 3, 4, 5}가 돼야 한다.

**Do your
homework!**

문제

7-22

배열 a와 같은 배열(요소 수가 같고 모든 요소의 값이 같은 배열)을 생성해서 반환하는 arrayClone 메서드를 작성하자.

```
int[] arrayClone(int[] a)
```

실행 예

```
요소 수:5
x[0]:11
x[1]:22
x[2]:33
x[3]:44
x[4]:55
```

배열 x를 복사해서 배열 y를 생성했습니다.

y[0] = 11

y[1] = 22

y[2] = 33

y[3] = 44

y[4] = 55

문제

7-23

배열 a의 요소 중에서 값이 x인 모든 요소의 인덱스를 앞에서부터 순서대로 저장해서 반환하는 arraySrchIdx를 작성하자.

```
int[] arraySrchIdx(int[] a)
```

예) 배열 a의 요소가 {1, 5, 4, 8, 5, 5, 7}이고 arraySrchIdx(a, 5)를 호출한 경우, 반환할 배열은 {1, 4, 5}가 된다(값이 5인 요소의 인덱스를 나열한 것).

실행 예

요소 수:7

x[0]:1

x[1]:5

x[2]:4

x[3]:8

x[4]:5

x[5]:5

x[6]:7

탐색할 값:5

일치하는 요소의 인덱스

1

4

5

문제

7-24

배열 a에서 요소 a[idx]를 삭제한 배열을 반환하는 arrayRmvOf를 작성하자.

```
int[] arrayRmvOf(int[] a, int idx)
```

삭제는 a[idx]보다 뒤에 있는 모든 요소를 하나씩 앞으로 이동시킬 것.

예) 배열 a의 요소가 {1, 3, 4, 7, 9, 11}일 때에 arrayRmvOf(a, 2)를 호출한 경우, 반환할 배열의 요소는 {1, 3, 7, 9, 11}이다.

실행 예

요소 수:6
x[0]:1
x[1]:3
x[2]:4
x[3]:7
x[4]:9
x[5]:11

삭제할 요소의 인덱스:2
y[0] = 1
y[1] = 3
y[2] = 7
y[3] = 9
y[4] = 11

문제

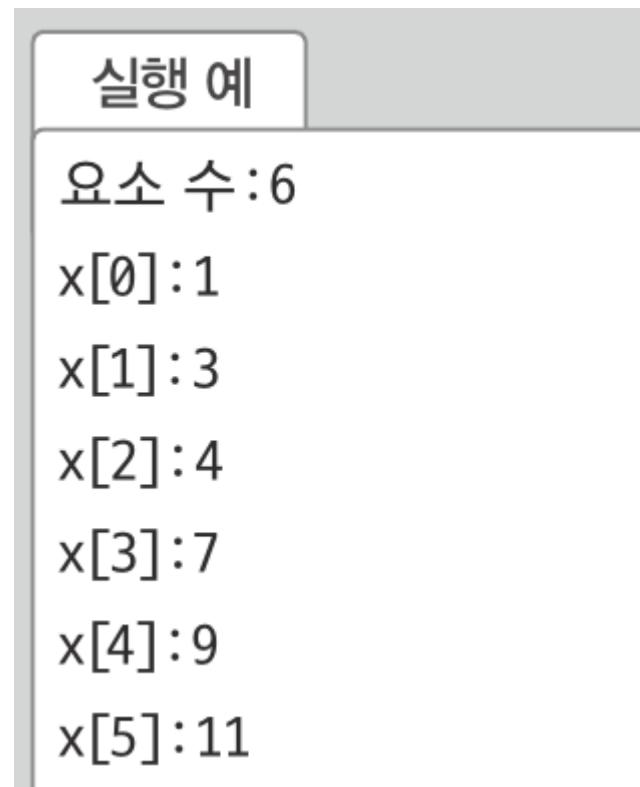
7-25

배열 a에서 요소 a[idx]부터 n개의 요소를 삭제한 배열을 반환하는 arrayRmvOfN 메서드를 작성하자.

```
int[] arrayRmvOfN(int[] a, int idx, int n)
```

삭제는 a[idx]보다 뒤에 있는 모든 요소를 n개 앞으로 이동해서 할 것.

예) 배열 a의 요소가 {1, 3, 4, 7, 9, 11}일 때에 arrayRmvOfN(a, 1, 3)을 호출한 경우, 반환할 배열의 요소는 {1, 9, 11}이다.



삭제를 시작할 인덱스:1

삭제할 요소의 개수:3

y[0] = 1

y[1] = 9

y[2] = 11

문제

7-26

배열 a의 요소 a[idx]에 x를 삽입해서 배열로 반환하는 arrayIns0f를 작성하자.

```
int[] arrayIns0f(int[] a, int idx, int x)
```

삽입할 때는 a[idx]뒤에 있는 모든 요소를 하나씩 뒤로 이동시킬 것.

예) 배열 a의 요소가 {1, 3, 4, 7, 9, 11}일 때에 arrayIns0f(a, 2, 99)를 호출할 경우, 반환할 배열의 요소는 {1, 3, 99, 4, 7, 9, 11}이 된다.

실행 예	
요소 수:	6
x[0]:	1
x[1]:	3
x[2]:	4
x[3]:	7
x[4]:	9
x[5]:	11

삽입할 인덱스: 2

삽입할 값: 99

y[0] = 1

y[1] = 3

y[2] = 99

y[3] = 4

y[4] = 7

y[5] = 9

y[6] = 11

**Do your
homework!**

문제

7-27

행렬 x와 y의 합을 구해서 z에 저장하는 addMatrix 메서드를 작성하자.

```
boolean addMatrix(int[][] x, int[][] y, int[][] z)
```

3개 배열의 요소 수가 같으면 계산해서 true를 반환하고 같지 않으면 계산없이 false를 반환할 것.

실행 예

행렬a

1 2 3

4 5 6

행렬b

6 3 4

5 1 2

행렬c

7 5 7

9 6 8

```
// 배열의 복사본 작성  
import java.util.Scanner;  
  
class arrayClone {  
    //--- 배열 a의 복사본을 생성해서 반환 ---//  
    static int[] arrayClone(int[] a) {  
  
        return c;  
    }  
}
```

```
public static void main(String[] args) {  
    Scanner stdIn = new Scanner(System.in);  
  
    System.out.print("요소 수:");  
    int num = stdIn.nextInt();  
  
    int[] x = new int[num];           // num 크기의 배열 생성  
  
    }  
}
```

```
// 두 행렬의 합을 구한다
class AddMatrix {
    //--- 행렬x와 y의 합을 z에 대입 ---/
    static boolean addMatrix(int[][] x, int[][] y, int[][] z) {
        for (int i = 0; i < x.length; i++) {
            for (int j = 0; j < x[0].length; j++) {
                z[i][j] = x[i][j] + y[i][j];
            }
        }
        return true;
    }

    //--- 행렬 m의 모든 요소를 표시 ---/
}

public static void main(String[] args) {
    int[][] a = { {1, 2, 3}, {4, 5, 6} };
    int[][] b = { {6, 3, 4}, {5, 1, 2} };
    int[][] c = new int[2][3];

    for (int i = 0; i < a.length; i++) {
        for (int j = 0; j < a[0].length; j++) {
            c[i][j] = a[i][j] + b[i][j];
        }
    }
}
```

문제

7-28

행렬 x와 y의 합을 저장해서 2차원 배열로 반환하는 메서드를 작성하자(행 수 및 열 수가 동일한 x, y를 전달한다고 가정해도 좋다).

```
int[][] addMatrix(int[][] x, int[][] y)
```

실행 예

행렬의 행 수:2

행렬의 열 수:3

a[0][0]:1

a[0][1]:2

a[0][2]:3

a[1][0]:4

a[1][1]:5

a[1][2]:6

b[0][0]:6

b[0][1]:3

b[0][2]:4

b[1][0]:5

b[1][1]:1

b[1][2]:2

행렬a

1 2 3

4 5 6

행렬b

6 3 4

5 1 2

행렬c

7 5 7

9 6 8

문제**7-29**

2차원 배열 a와 동일한 배열(요소 수가 같고 모든 요소의 값도 같다)을 생성해서 반환하는 aryClone2를 작성 하자.

```
int[][] aryClone2(int[][] a)
```

실행 예

행렬의 행 수:2
행렬의 열 수:3
a[0][0]:1
a[0][1]:2
a[0][2]:3
a[1][0]:4
a[1][1]:5
a[1][2]:6

행렬a
1 2 3
4 5 6
행렬a의 복사본
1 2 3
4 5 6

문제

7-30

2개의 int형 정수 a, b의 최솟값, 3개의 int형 정수 a, b, c의 최솟값, 배열 a의 최솟값을 각각 구하자. 다음에 정의된 메서드 형식을 사용할 것.

```
int min(int a, int b) int min(int a, int b, int c) int min(int[] a)
```

실행 예

x값:13

y값:52

z값:11

배열 a의 요소 수:4

a[0]:3

a[1]:7

a[2]:1

a[3]:8

x, y의 최솟값은 13입니다.

x, y, z의 최솟값은 11입니다.

배열 a의 최솟값은 1입니다.

**Do your
homework!**

문제

7-31

int형 변수 x의 절댓값, long형 변수 x의 절댓값, float형 변수 x의 절댓값, double형 변수 x의 절댓값을 구하는 다중 메서드를 작성하자.

int absolute(int x)

float absolute(float x)

long absolute(long x)

double absolute(double x)

실행 예

```
int    형 정수 a의 값:5
long   형 정수 b의 값:-8
float  형 실수 c의 값:-13.5
double 형 실수 d의 값:27.4
a의 절댓값은 5입니다.
b의 절댓값은 8입니다.
c의 절댓값은 13.5입니다.
d의 절댓값은 27.4입니다.
```

```
// 정수 및 실수의 절댓값을 구하는 메서드(다중 정의)
import java.util.Scanner;
```

```
class Absolute {
    //--- x의 절댓값을 반환 ---//
```

```
}
```

```
public static void main(String[] args) {
    Scanner stdIn = new Scanner(System.in);
    System.out.print("int 형 정수 a의 값:"); int a = stdIn.nextInt();
    System.out.print("long 형 정수 b의 값:"); long b = stdIn.nextLong();
    System.out.print("float 형 실수 c의 값:"); float c = stdIn.nextFloat();
    System.out.print("double형 실수 d의 값:"); double d = stdIn.nextDouble();
    System.out.println("a의 절댓값은 " + absolute(a) + "입니다.");
    System.out.println("b의 절댓값은 " + absolute(b) + "입니다.");
    System.out.println("c의 절댓값은 " + absolute(c) + "입니다.");
    System.out.println("d의 절댓값은 " + absolute(d) + "입니다.");
}
```

문제**7-32**

[문제 7-12])의 printBits 메서드는 int형 정수값의 내부 비트를 표시하는 것이었다. 다른 정수형에 대해서도 동일한 처리를 하는 다중 정의 메서드를 작성하자.

```
void printBits(byte x)  
void printBits(int x)
```

```
void printBits(short x)  
void printBits(long x)
```

실행 예

```
byte 형 정수 a의 값:30  
short 형 정수 b의 값:300  
int 형 정수 c의 값:30000  
long 형 정수 d의 값:3000000  
a의 비트:00011110  
b의 비트:0000000100101100  
c의 비트:00..000111010100110000  
d의 비트:00.. 1011011100011011000000
```

문제

7-33

int형의 1차원 배열과 int형의 2차원 배열(행에 따라 열 수가 다를 가능성이 있음)의 모든 요솟값을 표시하는
다중 정의 메서드를 작성하자.

```
void printArray(int[] a) void printArray(int[][] a)
```

1차원 배열 표시에선 각 요소 사이에 문자 1개분의 공간을 둘 것. 또한, 2차원 배열 표시에선 각 열의 숫자가
왼쪽에 맞추어 정렬되도록 최소한의 공간을 둘 것.

예를 들면 다음과 같이 표시

1차원 배열의 표시 예

12 536 -8 7 2

2차원 배열의 표시 예

32 -1 32 45 67

535 99999 2

2 5 -123 9

실행 예

1차원 행렬 x의 요소 수:4

x[0]:12

x[1]:536

x[2]:-8

x[3]:7

2차원 배열 y의 행 수:3

0행째 열 수:5

1행째 열 수:3

2행째 열 수:4

각 요소의 값을 입력하자.

y[0][0]:32

y[0][1]:-1

y[0][2]:32

y[0][3]:45

y[0][4]:67

y[1][0]:535

y[1][1]:99999

y[1][2]:2

y[2][0]:2

y[2][1]:5

y[2][2]:-123

y[2][3]:9

1차원 배열x

12 536 -8 7

2차원 배열y

32 -1 32 45 67

535 99999 2

2 5 -123 9

8장

클래스의 기본

객체 지향 프로그래밍의 근간이 되는 기술이 바로 클래스이다.

사람과 자동차를 객체로 만드는 프로그램을 통해

클래스의 기본 개념을 학습한다.

- 클래스
- 클래스형 변수
- 멤버 접근 연산자
- 인스턴스와 객체
- 필드와 인스턴스 변수
- 생성자와 인스턴스 메서드
- this
- 데이터 은닉과 캡슐화

문제**8-1**

이름, 신장, 체중 등을 멤버로 가지는 '사람 클래스'를 작성하자(필드는 자유롭게 설정할 것).

실행 결과

이름:길동

신장:170cm

체중:60kg

이름:철수

신장:166cm

체중:72kg

문제**8-2**

앞 페이지에서 다룬 문제점들이 해결되도록 Human 클래스를 수정하자. 클래스에는 필드뿐만 아니라 생성자와 메서드도 정의할 것.

실행 결과

이름:길동

신장:170cm

체중:63kg

이름:철수

신장:166cm

체중:67kg

문제

8-3

자동차 클래스 Car를 작성하자.

이름, 번호(이상 String형) / 폭, 높이, 길이(이상 int형) / 현재 위치 X좌표, Y좌표, 탱크 용량, 남은 연료, 연비(이상 double형)를 필드를 가진다. 현재 위치 X, Y좌표를 불러오기, 남은 연료 불러오기, 사양 표시하기, 지시한 X, Y 방향만큼 자동차 이동하기, 급유하기 등의 메서드를 가진다.

- 생성 시에는 X좌표, Y좌표는 모두 0이며, 그 이외의 필드에는 주어진 값을 설정할 것.

문제**8-4**

앞 문제에서 만든 자동차 클래스 Car를 이용하는 프로그램을 작성한다.

실행 결과

이름 : k3

번호 : 서울999-99

전폭 : 1660mm

전고 : 1500mm

전장 : 3640mm

탱크 : 40.0리터

연비 : 12.0km/리터

이름 : k5

번호 : 서울999-98

전폭 : 1660mm

전고 : 1525mm

전장 : 3695mm

탱크 : 41.0리터

연비 : 12.0km/리터

간단한 클래스 작성

9장

- 접근자(게터와 세터)
- 클래스형 변수와 인스턴스 참조
- 클래스 인스턴스의 배열
- 복제 생성자
- 문자열화와 `toString` 메서드
- 클래스 인스턴스의 비교
- has-A 관계

문제

9-1

X좌표와 Y좌표로 구성된 2차원 좌표 Coordinate 클래스를 작성하자. X좌표와 Y좌표의 값을 반환하는 메서드와 설정하는 메서드를 정의할 것.

실행 예1

좌표 p를 입력하세요.

X좌표:7.5

Y좌표:3.2

좌표 q를 입력하세요.

X좌표:7.5

Y좌표:3.2

p != q입니다.

p와 q가 같습니다.

p와 q가 같습니다.

실행 예2

좌표 p를 입력하세요.

X좌표:7.5

Y좌표:3.2

좌표 q를 입력하세요.

X좌표:0.0

Y좌표:0.0

p != q입니다.

p와 q가 다릅니다.

p와 q가 다릅니다.

문제**9-2**

[문제 9-1]의 문제점들을 해결하도록 좌표 클래스 Coordinate을 수정하자.

문제

9-3

[문제 8-2]에서 작성한 '사람 클래스'의 배열을 생성하는 프로그램을 작성하자. 생성 시에 요소를 초기화하는 것과 생성 후에 요소에 값을 대입하는 것 등 여러 개의 패턴을 만들 것.

실행 예

배열 q의 요소 수:3	No.0 철수 170cm 70kg
q[0]	No.1 길동 160cm 59kg
이름:michale	■배열 q
신장:155	No.0 michale 155cm 50kg
체중:50	...중략...
...중략...	■배열 x
배열 y의 행 수:3	0행
y[0]의 열 수:2	No.0 김철수 175cm 52kg
y[0][0]	No.1 김영희 169cm 60kg
이름:patrick	1행
신창:180	No.0 홍길동 178cm 70kg
체중:70	...중략...
...중략...	■배열 y
■배열 p	...중략...

문제

9-4

앞 문제에서 만든 자동차 클래스 Car를 이용하는 프로그램을 작성한다.

실행 예

day1을 입력하자.

년: 2017

월: 5

일: 25

day1 = 2017년 05월 25일(목)

day2를 day1과 같은 날짜로 만들었습니다.

day2 = 2017년 05월 25일(목)

: day1과 day2가 같습니다.

d1 = 0001년 01월 01일(월)

d2 = 2017년 01월 01일(일)

d3 = 2017년 10월 01일(일)

d4 = 2017년 10월 15일(일)

a[0] = 0001년 01월 01일(월)

a[1] = 0001년 01월 01일(월)

a[2] = 0001년 01월 01일(월)

문제**9-5**

다음의 은행 계좌 클래스에 계좌 개설일 필드와 `toString` 메서드를 추가하시오. 생성자를 변경하거나 계좌 개설일의 게터(계좌 개설일 필드가 참조하는 날짜 인스턴스의 복사본을 반환) 등을 추가할 것.

문제

9-6

문제 8-2에서 작성한 '사람 클래스'에 생일 필드와 `toString` 메서드를 추가하자.

- 생성자를 변경하거나 생일의 게터 메서드 등을 추가할 것

실행 결과

```
gildong = {길동:170cm 60kg  
1975년 03월 12일(수)출생}  
chulsu  = {철수:166cm 72kg  
1987년 10월 07일(수)출생}
```

문제

9-7

시작일과 종료일로 구성되는 ‘기간’ 클래스 Period를 작성하자. 필드는 다음과 같으며, 생성자나 메서드 등은 자유롭게 정의한다.

```
class Period {  
    private Day from;          // 시작일  
    private Day to;            // 종료일  
}
```

실행 결과

태조 = {1392년 08월 05일(토)~1398년 10월 14일(월)}

정종 = {1398년 10월 14일(월)~1400년 11월 28일(화)}

태종 = {1400년 11월 28일(화)~1418년 09월 09일(토)}

세종 = {1418년 09월 09일(토)~1450년 03월 30일(화)}

10장

클래스 변수와 클래스 메서드

- 클래스 변수(정적 필드)
- 클래스 메서드(정적 메서드)
- 정적 초기값과 인스턴스 초기값
- 유틸리티 클래스

10장에선 클래스 변수와 클래스 메서드를 학습한다. 개별 클래스의 인스턴스에 속하는 인스턴스 변수, 인스턴스 메서드와 달리 클래스에 속해서 해당 클래스의 인스턴스로부터 공유되는 변수, 메서드를 가리킨다.

문제

10-1

[문제 8-2]의 사람 클래스인 Human을 각각의 인스턴스에 식별 번호가 부여되도록 수정하자. 인스턴스 생성 시마다 1, 2, … 형식으로 연속된 값을 부여할 것.

실행 결과

이름:홍길동

신장:170cm

체중:60kg

번호:1

이름:김철수

신장:166cm

체중:72kg

번호:2

문제

10-2

앞 문제의 Human 클래스에서 식별 번호와 관련된 처리만 추출해서, 일련 번호를 생성하는 Id 클래스를 작성하자.

실행 결과

a의 식별 번호 : 1

b의 식별 번호 : 2

Id.counter = 2

a.counter = 2

b.counter = 2

문제

10-3

앞 문제의 일변 번호 클래스 Id에, 마지막 부여한 식별 번호를 반환하는 클래스 메서드 getMaxId를 추가하자.

```
static int getMaxId()
```

이 메서드는 인스턴스를 n개 생성한 시점에 호출하면 n을 반환해야 한다.

실행 결과

```
a의 식별 번호:1  
b의 식별 번호:2  
마지막으로 부여한 식별 번호 = 2  
마지막으로 부여한 식별 번호 = 2  
마지막으로 부여한 식별 번호 = 2
```

문제

10-4

앞 문제에서 작성한 클래스 Id를 다음과 같이 변경한 클래스 ExId를 작성하자. 인스턴스를 생성할 때마다 식별 번호를 n씩 증가시켜서 부여한다(n은 양의 정수). n값은 지정하지 않으면 1로 설정하고 메서드를 통해서 확인 및 변경할 수 있다. 예를 들어 인스턴스를 3개 생성한 후에 n을 4로 변경한 경우, 인스턴스에 부여하는 식별 번호는 생성된 순으로 1, 2, 3, 7, 11, 15…가 된다.

실행 결과

a의 식별 번호 : 1

b의 식별 번호 : 2

c의 식별 번호 : 3

d의 식별 번호 : 7

e의 식별 번호 : 11

f의 식별 번호 : 15

마지막에 부여한 식별 번호 = 15

다음에 부여할 식별 번호 = 19

문제**10-5**

네 개의 정수형(byte, short, int, long)이 표현할 수 있는 최솟값과 최댓값을 표시하자.

실행 결과

정수형의 표현 범위

byte 형:-128 ~ 127

short형:-32768 ~ 32767

int 형:-2147483648 ~ 2147483647

long 형:-9223372036854775808 ~ 9223372036854775807

문제

10-6

실수값을 읽어서 절댓값과 제곱근을 구하고, 해당 값을 반지름으로 하는 원의 면적을 구하자.

실행 예

실수값: 5.5

절댓값: 5.5

제곱근: 2.345207879911715

면 적: 95.03317777109123

문제

10-7

배열의 값 중에서 최솟값과 최댓값을 구하는 메서드를 모은 유ти리티 클래스 `MinMax`를 작성하자.

실행 예		
x값:3		x, y의 최솟값은 3입니다.
y값:4		x, y의 최댓값은 4입니다.
z값:1		x, y, z의 최솟값은 1입니다.
배열 a의 요소 수:6		x, y, z의 최댓값은 4입니다.
a[0]:2		배열 a의 최솟값은 1입니다.
a[1]:1		인덱스는 { 1 4 }입니다.
a[2]:3		배열 a의 최댓값은 3입니다.
a[3]:2		인덱스는 { 2 5 }입니다
a[4]:1		
a[5]:3		

문제

10-9

`int`형 `X`, `Y`좌표로 구성되는 2차원 좌표 클래스를 작성하자.

한쪽 좌표만 지정하거나 양쪽 모두 지정하지 않아도 인스턴스가 생성되도록 할 것(지정하지 않은 경우에는 0으로 설정). 인스턴스의 누적 생성 횟수가 프로그램 실행 일과 동일할 때(예를 들어 2018년 10월 3일에 실행 했다면 3개째 인스턴스 생성 시)에 “당첨! 오늘 3개의 좌표가 생성됐습니다.”라고 표시할 것.

실행 예
0 (0, 0)
1 (0, 0)
당첨!!오늘 3개의 좌표가 생성됐습니다.
2 (0, 0)
... 이하 생략...

실행 예

계좌 개설 수:2
[0]...간이 [1]...상세:0
계좌 정보를 입력하세요.
명 의:홍길동
번 호:12345
은행 계좌를 개설해주셔서 감사합니다.
계좌 기본 정보:{홍길동, 123456, 0}
식별 번호:1
개설 일:2018년 09월 17일(월)
[0]...간이 [1]...상세:1
계좌 정보를 입력하세요.
명 의:김철수
번 호:654321
잔 고:3000
개설 연도:2018
개설 월:3
개설 일:5
은행 계좌를 개설해주셔서 감사합니다.
계좌 기본 정보:{김철수, 654321, 3000}
식별 번호:2
개설 일:2018년 03월 05일(월)

문제**10-10**

인스턴스가 생성될 때마다 “은행 계좌를 개설해주셔서 감사합니다.”하고 표시하도록, [문제 9-5]의 Account 클래스를 수정하자. 표시는 인스턴스 초기화 블록에서 할 것. Day 클래스는 [문제 10-8]의 버전을 사용할 것.

· 패키지

· 형 임포트 선언(단일/온 디맨드)

· 정적 임포트 선언(단일/온 디맨드)

· 패키지와 디렉터리

· 패키지와 접근 제어

· 고유한 패키지명

11장

패키지

데이터와 메서드를 묶어서 캡슐화한 것이 ‘클래스’이다.

이 클래스를 모아서 캡슐화한 것이 ‘패키지’이다.

11장에선 패키지 사용법 및 작성법 등을 학습한다.

문제**11-1**

오늘의 운세 프로그램을 작성하자. 오늘 날짜와 대길(大吉), 길(吉), 중길(中吉), 소길(小吉), 흉(凶) 중 하나의 운세를 표시할 것.

실행 예

오늘은 2018년 09월 19일입니다.
오늘의 운세는 길입니다.

문제**11-2**

실수값을 읽어서 절댓값과 제곱근을 구하자. 또한, 해당 실수값이 반지름인 원의 넓이를 구해서 표시하자.

실행 예

실 수: 5.5

절댓값: 5.5

제곱근: 2.345207879911715

넓 이: 95.03317777109123

문제**11-3**

프로그램을 실행하면 “오늘은 yyyy년 mm월 dd일입니다.”라고 한 번만 표시하며, 인스턴스를 생성할 때마다 각 인스턴스에 yyyyymmdd01, yyyyymmdd02, … 식으로 식별 번호를 부여하는 일련 번호 클래스 DateId를 작성하자([문제 10-3]의 Id 클래스를 수정해서 사용해도 좋다). 클래스의 패키지명은 id로 하고 DateId 클래스를 테스트하기 위한 프로그램 DateIdTester는 무명 패키지에 포함할 것.

실행 예

오늘은 2018년 09월 19일입니다.
a의 식별 번호 : 2018091901
b의 식별 번호 : 2018091902
c의 식별 번호 : 2018091903

12장

클래스의 상속과 다형성

12장에선 기존 클래스를 상속해서 새로운 클래스를 만드는
클래스 상속과 이것을 응용한 다형성에 대해 학습한다.

- 파생에 의한 자원 상속
- 상위/슈퍼 클래스, 하위/서브 클래스
- super(…)를 사용한 슈퍼 클래스의 생성자 호출
- super를 사용한 슈퍼 클래스의 멤버 접근
- is-A와 다형성
- 참조형의 캐스트(업/다운)
- 오버라이드와 @override 어노테이션
- final 클래스
- 상속과 접근성

문제**12-1**

다음 자동차 클래스 Car에 총 주행 거리를 나타내는 필드와 그 값을 확인하는 메서드를 추가해서 ExCar 클래스를 작성하자. Car 클래스에서 파생시킬 것.

▶ [문제 8-3)]에서 작성한 자동차 클래스(Ver. 1)을 다음과 같이 변경한다.

- 번호(number), 탱크 용량(tankage), 연비(sfc) 필드와 급유를 위한 refuel 메서드를 삭제한다. 연비는 1로 간주한다.
- 구입일 필드인 purchaseDay의 게터인 getPurchaseDay를 추가한다.
- 위 변경에 따라 생성자와 move 메서드를 수정한다.

문제

12-2

자동차 클래스 ExCar형의 인스턴스를 만들어 현재 위치와 남은 연료량, 구입일을 표시하자.

실행 예

현재위치 : (0.00,0.00)

남은연료 : 90.00리터

구 입 일 : 2015년 12월 24일(목)

문제**12-3**

자동차 클래스 ExCar형의 인스턴스를 만들고, putSpec 메서드를 호출해서 사양을 표시하는 프로그램을 작성하자.

문제**12-4**

슈퍼 클래스인 Car형의 클래스형 변수는 서브 클래스인 ExCar형의 인스턴스를 참조할 수 있다. 이것을 프로그램을 작성해서 확인하자.

문제**12-5**

서브 클래스인 ExCar형의 인스턴스를 참조하는 Car형의 클래스형 변수에 putSpec 메서드를 호출하면 어떻게 동작하는지 프로그램을 통해 확인하자.

문제**12-6**

서브 클래스인 클래스 ExCar형의 인스턴스를 참조하는 Car형의 클래스형 변수에, 메서드 move와 getTotalMileage를 호출한 경우의 동작을 확인하자.

문제**12-7**

아래에 있는 TimeAccount 클래스는 [문제 9-5]의 은행 계좌 클래스(Ver. 1)로부터 파생된 정기예금 은행 계좌 클래스이다. 은행 계좌 클래스형 변수 a, b의 보통예금과 정기예금의 잔고 합계를 비교한 결과를 반환하는 compBalance를 작성하자.

```
static int compBalance(Account a, Account b)
```

합계 금액을 비교해서 a가 많으면 1, 같으면 0, b가 많으면 -1를 반환할 것. 만약 a나 b의 참조 대상이 정기예금이 없는 Account형 인스턴스이면 보통예금의 합계를 비교할 것.

- 추상 클래스
- 추상 메서드
- 메서드 구현
- 문서화 주석
- javadoc 툴

13장

추상 클래스

13장에서 학습할 것은 실체가 아닌 개념을 나타내기 위한
추상 클래스와 추상 메서드이다. 추상 클래스나 추상 메서드를 이용하면
앞 장에서 학습한 다형성의 활용 폭이 넓어진다.

문제**13-1**

개와 고양이 클래스를 각각 Dog와 Cat으로 작성하자. 모두 동물 클래스인 Animal로부터 파생시킬 것. 동물 클래스는 추상 클래스로 작성하고, 동물의 이름 필드(생성자에서 값을 설정), 이름 게터, 짖기 메서드를 지닐 것. 또한, 개의 종류를 나타내는 문자열 필드를 Dog에 추가하고, 나이를 나타내는 정수형 필드를 Cat에 추가할 것 (모두 생성자에서 값을 설정한다).

실행 결과

뭉치 멍멍!

마이클 냐옹!

문제

13-2

앞 문제의 동물 클래스에 다음과 같이 `toString`과 `introduce` 메서드를 추가하자.

- `toString`: Dog 클래스에선 'type'의 name이라는 문자열(예: "치와와의 뭉치")을 반환하고, Cat 클래스에선 'age살의 name'이라는 문자열(예: "7살의 마이클")을 반환하자.
- `introduce`: 모든 동물 클래스에 공통. `toString`의 반환 문자열에 "이다."를 더해서 표시하고 `bark` 메서드를 사용해서 짖을 것(예: "치와와의 뭉치이다.멍멍!"이라고 표시).

실행 결과

```
a[0] = 치와와의 뭉치  
a[1] = 7살의 마이클  
a[0] = 치와와의 뭉치이다.멍멍!  
a[1] = 7살의 마이클이다.냐옹!
```

문제**13-3**

다음의 도형 클래스들(Shape, Point, AbstLine, HorzLine, VertLine, Rectangle)을 테스트하는 프로그램을 작성하자. Shape형의 배열을 이용해서 인스턴스를 생성하고 출력할 것. 단, 개별 요소가 참조하는 인스턴스는 프로그램 내에서 부여하는 것이 아니라 키보드를 통해 입력받을 것.

실행 결과

도형의 개수:4

1번 도형의 종류(1...점/2...수평선/3...수직선/4...사각형) :1

2번 도형의 종류(1...점/2...수평선/3...수직선/4...사각형) :2

길이는:8

3번 도형의 종류(1...점/2...수평선/3...수직선/4...사각형) :3

길이는:3

4번 도형의 종류(1...점/2...수평선/3...수직선/4...사각형) :4

너비:4

높이:3

Point

+

HorzLine(length:8)

VertLine(length:3)

|

|

|

Rectangle(width:4, height:3)

문제**13-4**

도형 클래스군에 직각 이등변 삼각형을 나타내는 클래스군을 추가하자. 좌하가 직각인 것, 좌상이 직각인 것, 우하가 직각인 것, 우상인 직각인 것을 추가할 것. 직각 이등변 삼각형을 나타내는 추상 클래스를 만들고 이 클래스에서 개별 클래스를 파생해서 만들 것.

실행 결과

삼각형의 개수 : 4	*
1번 삼각형 ... 중략... : 1	**
길이:2	
2번 삼각형 ... 중략... : 2	TriangleLU(length:3)
길이:3	***
3번 삼각형 ... 중략... : 3	**
길이:4	*
4번 삼각형 ... 중략... : 4	
길이:3	TriangleRB(length:4)
TriangleLB(length:2)	... 이하 생략 ...

문제**13-5**

가위바위보 게임에서 ‘플레이어’를 나타내는 추상 클래스를 정의하자. 클래스로부터 다음 클래스를 파생시킬 것.

- 사람 플레이어 클래스(낼 손을 키보드를 통해 입력 받음)
- 컴퓨터 플레이어 클래스(낼 손을 난수를 사용해 생성)

문제

13-6

앞 문제에서 작성한 플레이어 클래스를 이용해서 가위바위보 게임을 하는 프로그램을 작성하자.

실행 예

```
가위바위보!!! 0...바위/1...가위/2...보:1  
나는 가위이고, 당신은 가위입니다.  
비겼습니다.  
다시 한번? (0) 아니오 (1)네:1  
가위바위보!!! 0...바위/1...가위/2...보:2  
나는 가위이고, 당신은 보입니다.  
당신이 졌습니다.  
다시 한번? (0) 아니오 (1)네:0
```

문제**13-7**

[문제 9-6]에서 작성한 ‘사람 클래스’에 javadoc 주석을 부여한 (Ver.5)를 작성하자. 그리고 javadoc 툴을 사용해서 문서를 생성하자.

The screenshot shows a Java API documentation page for the `Human` class. The top navigation bar includes links for PACKAGE, CLASS (which is selected), USE, TREE, DEPRECATED, INDEX, and HELP. Below the navigation bar are links for PREV CLASS, NEXT CLASS, FRAMES, NO FRAMES, and ALL CLASSES. Further down are links for SUMMARY, NESTED, FIELD, CONSTR, and METHOD, and DETAIL: FIELD, CONSTR, and METHOD.

Class Human

`java.lang.Object`
Human

`public class Human`
extends java.lang.Object

클래스 Human(Vers5)는 인간 클래스이다.

Author:
시|비|타

Field Summary

Fields

Modifier and Type	Field and Description
private Day	birthday 생일을 나타내는 Day형 필드

14장

인터페이스

14장에선 인터페이스를 학습한다. 그대로는 사용할 수 없는 인터페이스는 클래스를 만들 때에 ‘구현’을 통해 사용할 수 있다. 인터페이스의 구현은 파생에 의한 클래스 계층 관계와는 또다른 관계를 클래스 사이에 부여한다.

- 인터페이스 선언
- 인터페이스 구현
- 클래스 파생과 인터페이스 구현
- 인터페이스 상속

문제

14-1

'입기'와 '벗기' 메서드를 가지는 Wearble 인터페이스를 작성하자. 또한, 이 인터페이스를 구현한 다음 2개의 클래스를 작성하자.

- 헤드폰 클래스 Headphone(볼륨 조절 메서드를 지님)
- 웨어러블 컴퓨터 클래스 WearableComputer(재시작 메서드를 지님)

실행 결과

헤드폰을 착용했습니다.
컴퓨터를 실행했습니다.
헤드폰을 벗었습니다.
컴퓨터를 껐습니다.

문제

14-2

다음의 스킨을 변경하는 Skinnable 인터페이스를 구현한 클래스를 작성하자. 검정/빨강/초록/파랑/노랑을 선택할 수 있도록 changeSkin 메서드를 구현할 것.

실행 결과

x의 스킨은 YELLOW입니다.
y의 스킨은 GREEN입니다.

문제**14-3**

다음은 면적을 구하는 `getArea` 메서드를 지니는 2차원 인터페이스 `Plane2D`와 이 인터페이스를 구현한 사각형 클래스(문제 13-3), 그리고 평행사변형 클래스이다. 이 도형 클래스들을 사용하는 프로그램 예를 작성하자.

실행 결과

a[0]의 면적=10

a[1]의 면적=10

문제

14-4

다음은 애완견 클래스 Pet으로부터 로봇 애완견 클래스 RobotPet을 파생한다. RobotPet 클래스를 확장해서 스킨을 변경할 수 있는 로봇형 애완견 클래스를 작성하자. Skinnable 인터페이스[문제 14-2]를 구현할 것.

실행 결과

- 나의 이름은 Kurt입니다.
- 주인님의 이름은 아이입니다.
 - 나는 로봇! 이름은 R2D2.
 - 주인님의 이름은 루크。
 - 나는 로봇! 이름은 OSX5.
 - 주인님의 이름은 Apple。
 - 스킨은 노랑입니다.
- 나의 이름은 마이클입니다.
- 주인님의 이름은 영남입니다.

문제**14-5**

스킨 변경이 가능하고 탈착이 가능한 헤드마운트 디스플레이 클래스 HeadMountedDisplay를 작성하자. 인터페이스인 Wearable[문제 14-1]과 Skinnable[문제 14-2]를 구현할 것.

실행 결과

디스플레이를 착용했습니다.
디스플레이를 벗었습니다.
YELLOW DISPLAY
디스플레이를 착용했습니다.
디스플레이를 벗었습니다.
BLACK DISPLAY

문제**14-6**

아래에 있는 DVDPlayers 클래스를 사용한 프로그램 예를 작성하자.

```
// 플레이어 인터페이스 Players
public interface Players {
    void play();      // 재생
    void stop();      // 정지
}
```

15장

문자와 문자열

사람과 프로그램 간에 정보를 교환할 때 필요한 문자와

문자열을 학습한다. 문자는 char형으로 표현하고

문자열은 String형으로 표현한다.

- char형

- 문자와 문자 리터럴

- 유니코드(Unicode)와 ASCII 코드, 유니코드 확장

- String형

- 문자열과 문자열 리터럴

- 문자열 배열

- 커맨드라인 인수

문제**15-1**

문자와 문자 코드를 이용해서 영문, 숫자 및 기호를 표시하자.

실행 결과

! 0021

" 0022

0023

...중략...

} 007d

~ 007e

문제**15-2**

String형의 문자열을 생성하는 프로그램을 작성하자. 생성은 여러 가지 방법을 사용할 것.

실행 예

문자열 : HAL

s1 = ABC

s2 =

s3 = ABCDEFGHIJ

s4 = FGH

s5 = XYZ

s6 = HAL

문제**15-3**

String형 변수가 null 참조 또는 빈 문자열 대입 그리고 다른 String형 변수가 참조하는 문자열을 대입하는 경우 어떻게 되는지 프로그램을 작성해 비교해보자.

실행 결과

문자열 s1 = null

문자열 s2 =

문자열 s3 = ABC

문자열 s4 = ABC

문자열 s5 = XYZ

s3와 s4는 같은 문자열 리터럴을 참조하고 있다.

문제**15-4**

문자열을 읽어서 그 문자열을 역순으로 표시하는 프로그램을 작성하자.

실행 예

문자열:AB가나

반대로 읽으면 나가BA입니다.

문제**15-5**

문자열을 읽어서 포함된 모든 문자의 문자 코드를 표시하자.

실행 예

문자열s : AB가나

s[0] = A 41

s[1] = B 42

s[2] = 가 AC00

s[3] = 나 B098

문제**15-6**

문자열 s1 안에 문자열 s2가 포함됐는지 확인하는 프로그램을 작성하자. 포함돼 있지 않으면 “s1 안에 s2가 포함돼 있지 않습니다.”고 표시하고, 포함된 경우는 일치하는 부분을 위아래 정렬되도록 가로로 표시할 것.

실행 예

문자열 s1 : ABCDEFGHI

문자열 s2 : EFG

ABCDE
FGHI

EFG

문제**15-7**

키보드에서 입력한 2개의 문자열이 같은지 판정하는 프로그램을 작성하자.

실행 예

```
문자열 s1 : ABC  
문자열 s2 : ABC  
s1 != s2입니다.  
s1과 s2의 내용이 같습니다.
```

문제**15-8**

두 문자열의 대소 관계를 판정하는 프로그램을 작성하자.

실행 예

문자열 s1:ABC

문자열 s2:XYZ

s1이 작다.

문제**15-9**

부동소수점값 x에서 소수점 이하 부분 p자리로, 그리고 전체를 적어도 w자리로 표시하는 메서드 printDouble을 작성하자.

```
printDouble(double x, int p, intw)
```

실행 예

```
실수값:3.1415926535  
전체 자릿수:9  
소수점 이하 자릿수:4  
3.1416
```

문제**15-10**

String형 배열에 저장된 모든 문자열을 표시하는 메서드를 작성하자. 문자열 표시는 charAt 메서드에 의해 한 문자씩 순회하고 각 문자열을 표시할 때마다 줄 바꿈 문자를 출력할 것.

실행 예

```
문자열의 개수:3  
sx[0] = Turbo  
sx[1] = NA  
sx[2] = DOHC  
Turbo  
NA  
DOHC
```

문제**15-11**

3명이 참여하는 가위바위보 게임을 작성하자. 플레이어 3명 중 2명은 컴퓨터가 하고 남은 1명은 사람이 할 것.
또한, [문제 13-5]에서 작성한 플레이어 클래스를 이용할 것.

실행 예

가위바위보!!! 0...바위／1...가위／2...보:0
컴퓨터1은 바위이고, 컴퓨터2는 바위이며,
당신은 바위입니다.
비겼습니다.
다시 한 번? (0)아니오 (1)네 :0

문제**15-12**

프로그램 실행 시에 커맨드라인에서 부여한 문자열(커맨드라인 인수)를 모두 표시하자.

실행 예

```
java PrintArgs Turbo NA DOHC Y  
args[0] = Turbo  
args[1] = NA  
args[2] = DOHC
```

문제

15-13

커맨드라인 인수로 부여된 값을 반지름으로 하는 원의 둘레와 넓이를 구하자.

실행 예

java Circle 5.5

반지름 5.5인 원의 둘레는 34.56이고 넓이는 95.03입니다.

문제

15-14

커맨드라인 인수로 부여된 모든 숫자를 합해서 표시하는 프로그램을 작성하자. 확장 for문을 사용할 것.

실행 예

```
java SumOfArgs 5.5 3.75 2.25  
합계는 11.5입니다.
```

문제

15-15

커맨드라인 인수로 지정한 월의 달력을 표시하는 프로그램을 작성하자. 커맨드라인에서 연(年)만 지정한 경우는 해당 연의 1월부터 12까지 달력을 표시하고 연과 달이 주어진 경우는 해당 월의 달력을 표시한다. 만약 연과 월 모두 주어지지 않은 경우에는 현재 월의 달력을 표시할 것.

실행 예

java Calendar

2025년 6월 달력

일 월 화 수 목 금 토

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30

실행 예

java Carendar 1963 11

1963년 11월 달력

일 월 화 수 목 금 토

1 2

3 4 5 6 7 8 9

10 11 12 13 14 15 16

17 18 19 20 21 22 23

24 25 26 27 28 29 30

...중략...

21 22 23 24 25 26 27

28 29 30

12월

일 월 화 수 목 금 토

1 2 3 4

5 6 7 8 9 10 11

12 13 14 15 16 17 18

19 20 21 22 23 24 25

26 27 28 29 30 31

- ▶ 인수를 부여하지 않고 실행하면 현재 월의 달력이 표시됨

- ▶ 연과 월을 부여해서 실행하면 해당 연,월의 달력 표시

- ▶ 연을 지정해서 실행하면 해당 연의 1월부터 12월까지의 달력이 표시된다

16장

예외 처리

프로그램상에서 예기치 못한 상태나 대응하기 곤란한 상황이 발생할 수 있다.

이 경우 치명적인 상황으로 이어지는 것을 막고 정상 상태로 회복시키는

'예외 처리'를 배운다.

- 런타임 오류와 예외
- 예외와 예외 처리
- 예외 던지기와 throw문
- 예외 포착(try문/try블록)
- 예외 핸들러(catch절)
- 검사 예외
- 비검사 예외

문제**16-1**

입력한 두 정수값의 곱과 몫을 구하는 프로그램을 작성하자. 다양한 값을 입력해서 프로그램의 동작을 확인해보자.

실행 예

x값: 7

y값: 5

$x * y = 35$

$x / y = 1$

문제

16-2

[문제 16-2]의 프로그램을 수정해서 정수값의 곱과 몫을 구하는 처리를 작성하고, 각각을 순차적으로 호출해서 결과를 표시하도록 독립된 메서드로 작성하자. 그리고 예외를 캐치하도록 수정하자.

실행 예①

```
x값 : 7  
y값 : ABC  
입력 오류 발생.java.util.InputMismatchException  
java.util.InputMismatchException  
    at java.util.Scanner.throwFor(Unknown Source)  
    at java.util.Scanner.next(Unknown Source)  
    at java.util.Scanner.nextInt(Unknown Source)  
    at java.util.Scanner.nextInt(Unknown Source)  
    at MulDiv2.main(MulDiv2.java:22)  
프로그램을 종료합니다.
```

부적합값 입력

입력 부적합 예외

실행 예②

```
x값 : 7  
y값 : 0  
x * y = 0
```

```
산술 오류 발생.java.lang.ArithmaticException: / by zero  
java.lang.ArithmaticException: / by zero  
    at MulDiv2.div(MulDiv2.java:10)  
    at MulDiv2.muldiv(MulDiv2.java:14)  
    at MulDiv2.main(MulDiv2.java:23)  
프로그램을 종료합니다.
```

0 입력

0(zero) 나눗셈으로 발생한 산술 예외

문제

16-3

Exception 예외와 RuntimeException 예외를 던지고 캐치하는 프로그램을 작성하자.

- 키보드를 통해 입력받은 값에 따라 던지거나 별도의 핸들러로 캐치할 것

실행 예

SW : 1

검사 예외 발생!!

SW : 2

비검사 예외 발생!!

문제**16-4**

프로그램의 이전 실행 날짜 및 시간, 종료 날짜 및 시간을 표시하는 프로그램을 작성하자. 처음 프로그램을 실행할 때는 “이 프로그램을 실행하는 것은 처음입니다”고 표시할 것.

a 처음 프로그램을 실행한 경우

실행 예

이 프로그램을 실행하는 것은 처음입니다.

b 2회째 실행한 경우

실행 예

이전 시간은 2018년 09월 30일 20시 04분 37초였습니다.

문제

16-5

2개의 정수를 덧셈하는 메서드를 작성하자. 더할 값과 연산 결과가 모두 한자리가 되도록 하는 것이 전제이며 그렇지 않은 경우는 예외를 발생시킬 것.

실행 예1

정수 a:52

정수 b:5

범위 밖 예외가 발생했습니다.

범위 밖 값:52

실행 예2

정수 a:7

정수 b:5

범위 밖 예외가 발생했습니다.

범위 밖 값:12

문제

16-6

배열의 요소를 반대로 나열하는 메서드를 작성하자. 존재하지 않는 요소에 접근하는 버그를 심어서 프로그램 동작을 확인할 것.

실행 예1

요소 수:5

x[0]:10

x[1]:73

x[2]:2

x[3]:-5

x[4]:42

java.lang.ArrayIndexOutOfBoundsException: 5

at ReverseArray1.swap(ReverseArray1.java:7)

at ReverseArray1.reverse(ReverseArray1.java:14)

at ReverseArray1.main(ReverseArray1.java:36)

문제

16-7

앞 문제의 reverse 메서드에서 `ArrayIndexOutOfBoundsException`를 캐치할 때 다른 예외를 던지도록 프로그램을 수정하자.

실행 예1

요소 수:5

x[0]:10

x[1]:73

x[2]:2

x[3]:-5

x[4]:42

예외 :

`java.lang.RuntimeException: reverse의 버그?`

예외의 원인:

`java.lang.ArrayIndexOutOfBoundsException: 5`