

# 예제 4-7 : 객체 배열 만들기 연습

1

예제 4-4의 Book 클래스를 활용하여  
2개짜리 Book 객체 배열을 만들고,  
사용자로부터 책의 제목과 저자를  
입력 받아 배열을 완성하라.

- ➔ 배열을 3개로 수정하십시오
- ➔ 첫 번째 책 제목과 저자는 교재의  
제목과 저자로, 나머지는 자유롭게  
입력하여 출력하도록 하시오
- ➔ 출력 시 "저자: ~~, 제목: ~~" 형태  
로 출력하십시오

```
import java.util.Scanner;
class Book {
    String title, author;
    public Book(String title, String author) {
        this.title = title;
        this.author = author;
    }
}

public class BookArray {
    public static void main(String[] args) {
        Book [] book = new Book[2]; // Book 배열 선언

        Scanner scanner = new Scanner(System.in);
        for(int i=0; i<book.length; i++) {
            System.out.print("제목>>");
            String title = scanner.nextLine();
            System.out.print("저자>>");
            String author = scanner.nextLine();
            book[i] = new Book(title, author); // 배열 원소 객체 생성
        }

        for(int i=0; i<book.length; i++)
            System.out.print("(" + book[i].title + ", " + book[i].author + ")");

        scanner.close();
    }
}
```

## 예제 4-8 : 인자로 배열이 전달되는 예

2

char[] 배열을 전달받아 출력하는 printCharArray() 메소드와 배열 속의 공백(' ') 문자를 ';'로 대체하는 replaceSpace() 메소드를 작성하라.

→ ';'를 '/'로 변환하고 'i'는 'o'로 변환하도록 수정하시오

```
public class ArrayParameterEx {  
    static void replaceSpace(char a[]) {  
        for (int i = 0; i < a.length; i++)  
            if (a[i] == ' ')  
                a[i] = ';'  
    }  
    static void printCharArray(char a[]) {  
        for (int i = 0; i < a.length; i++)  
            System.out.print(a[i]);  
        System.out.println();  
    }  
    public static void main (String args[]) {  
        char c[] = {'T','h','i','s',' ','i','s',' ','a',' ','p','e','n','c','i','l','.'};  
        printCharArray(c);  
        replaceSpace(c);  
        printCharArray(c);  
    }  
}
```

## 예제 4-11 : static 멤버를 가진 Calc 클래스 작성

3

전역 함수로 작성하고자 하는 abs, max, min의 3개 함수를 static 메소드로 작성하고 호출하는 사례를 보여라.

```
class Calc {  
    public static int abs(int a) { return a>0?a:-a; }  
    public static int max(int a, int b) { return (a>b)?a:b; }  
    public static int min(int a, int b) { return (a>b)?b:a; }  
}  
  
public class CalcEx {  
    public static void main(String[] args) {  
        System.out.println(Calc.abs(-5));  
        System.out.println(Calc.max(10, 8));  
        System.out.println(Calc.min(-3, -8));  
    }  
}
```

- ➔ 이 문제에서 절대값 대신 두 개의 입력을 받아 차를 구하는 diff(a, b) 메소드를 static으로 선언하는 것으로 수정
- ➔ Main에서 "1. Diff, 2. Max, 3. Min"으로 메뉴를 출력하고 하나의 값을 선택하게 한 후 두 개의 정수를 입력 받아서 그 결과를 출력하도록 하시오.
- ➔ 예를 들어, 1을 선택하고 정수 -2와 3을 입력하면 "Diff는 5 입니다"로 출력하도록 하시오. (메뉴 선택에 따른 값 계산과 출력은 switch 문 사용)

# 예제 4-12 : static을 이용한 환율 계산기

4

static 멤버를 이용하여 달러와 원화를 변환 해주는 환율 계산기를 만들어보자.

```
class CurrencyConverter {
    private static double rate; // 한국 원화에 대한 환율
    public static double toDollar(double won) {
        return won/rate; // 한국 원화를 달러로 변환
    }
    public static double toKWR(double dollar) {
        return dollar * rate; // 달러를 한국 원화로 변환
    }
    public static void setRate(double r) {
        rate = r; // 환율 설정. KWR/$1
    }
}

public class StaticMember {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("환율(1달러)>> ");
        double rate = scanner.nextDouble();
        CurrencyConverter.setRate(rate); // 미국 달러 환율 설정
        System.out.println("백만원은 $" + CurrencyConverter.toDollar(1000000) + "입니다.");
        System.out.println("$100는 " + CurrencyConverter.toKWR(100) + "원입니다.");
        scanner.close();
    }
}
```

➔ 위 프로그램에서, 환율 입력 후 한화 금액을 입력 받아 달러 금액을 출력하고, 달러 금액을 입력 받아 한화 금액을 입력하도록 수정 하시오.