

23-여름학기 자바프로그래밍(001) Lab0627

소프트웨어학부 컴퓨터과학전공 2016133 이유진

1. 실습 1 BankAccountTest.java

```
1 //이유진 (2016133)
2 //2023-06-27
3 //Lab0627-1: BankAccount 클래스를 이용해 입출금/계좌이체 기능을 구현함: 계좌번호로 객체생조변수 전달 실습
4
5 import java.util.Scanner;
6
7 class BankAccount {
8     private int balance = 0; //잔액
9
10    // 생성자
11    public BankAccount(int balance) {
12        super();
13        this.balance = balance;
14    }
15
16    // 생성자, 접근자 대신 입/출금, 이체 메소드 사용
17
18    void withdraw(int m) { // 출금
19        if (m <= 0) // 매개변수로 들어온 금액이 양수일 때만 처리
20            System.out.println("적용하지 않은 출금 금액!");
21        else {
22            if (this.balance >= m) // 출금
23                this.balance -= m;
24            else // 잔액 부족: 매개변수로 들어온 금액보다 잔액이 크거나 같을 때만 출금 처리
25                System.out.println("잔액 부족!");
26        }
27    }
28
29    void deposit(int m) { // 입금
30        if (m <= 0) // 매개변수로 들어온 금액이 양수일 때만 처리
31            System.out.println("적용하지 않은 입금 금액!");
32        else
33            this.balance += m; // 입금
34    }
35
36    public void transfer(int amount, BankAccount otherAccount) { // 계좌간 이체
37        if (amount <= 0) // 매개변수로 들어온 금액이 양수일 때만 처리
38            System.out.println("적용하지 않은 이체 금액!");
39        else {
40            if (this.balance >= amount) { //현재 계좌에서 amount만큼 빼고 다른 계좌에 amount만큼을 더함
41                this.balance -= amount;
42                otherAccount.balance += amount;
43            }
44            else // 잔액 부족
45                System.out.println("잔액 부족!");
46        }
47    }
48
49    @Override
50    public String toString() {
51        return "BankAccount [balance=" + balance + "]";
52    }
53 }
54
55 public class BankAccountTest {
56
57     public static void main(String[] args) {
58
59         Scanner scan = new Scanner(System.in);
60         System.out.print("계좌 1 잔액 입력: ");
61         int a1 = scan.nextInt();
62         BankAccount myAccount1 = new BankAccount(a1);
63         System.out.print("계좌 2 잔액 입력: ");
64         int a2 = scan.nextInt();
65         BankAccount myAccount2 = new BankAccount(a2);
66
67         System.out.println("myAccount1: " + myAccount1);
68         System.out.println("myAccount2: " + myAccount2);
69         System.out.println();
70
71         System.out.print("계좌 1 --> 계좌 2 이체 금액 입력: ");
72         int t = scan.nextInt();
73         myAccount1.transfer(t, myAccount2);
74         System.out.println("transfer 조출 후");
75
76         System.out.print("계좌 1 --> 계좌 2 이체 금액 입력: ");
77         int t = scan.nextInt();
78         myAccount1.transfer(t, myAccount2);
79         System.out.println("transfer 조출 후");
80         System.out.println("myAccount1: " + myAccount1);
81         System.out.println("myAccount2: " + myAccount2);
82         scan.close();
83     }
84 }
```

```
Console × Problems Debug Shell
<terminated> BankAccountTest [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.e
계좌 1 잔액 입력: 10000
계좌 2 잔액 입력: 2000
myAccount1: BankAccount [balance=10000]
myAccount2: BankAccount [balance=2000]

계좌 1 --> 계좌 2 이체 금액 입력: 3000
transfer 호출 후
myAccount1: BankAccount [balance=7000]
myAccount2: BankAccount [balance=5000]
```

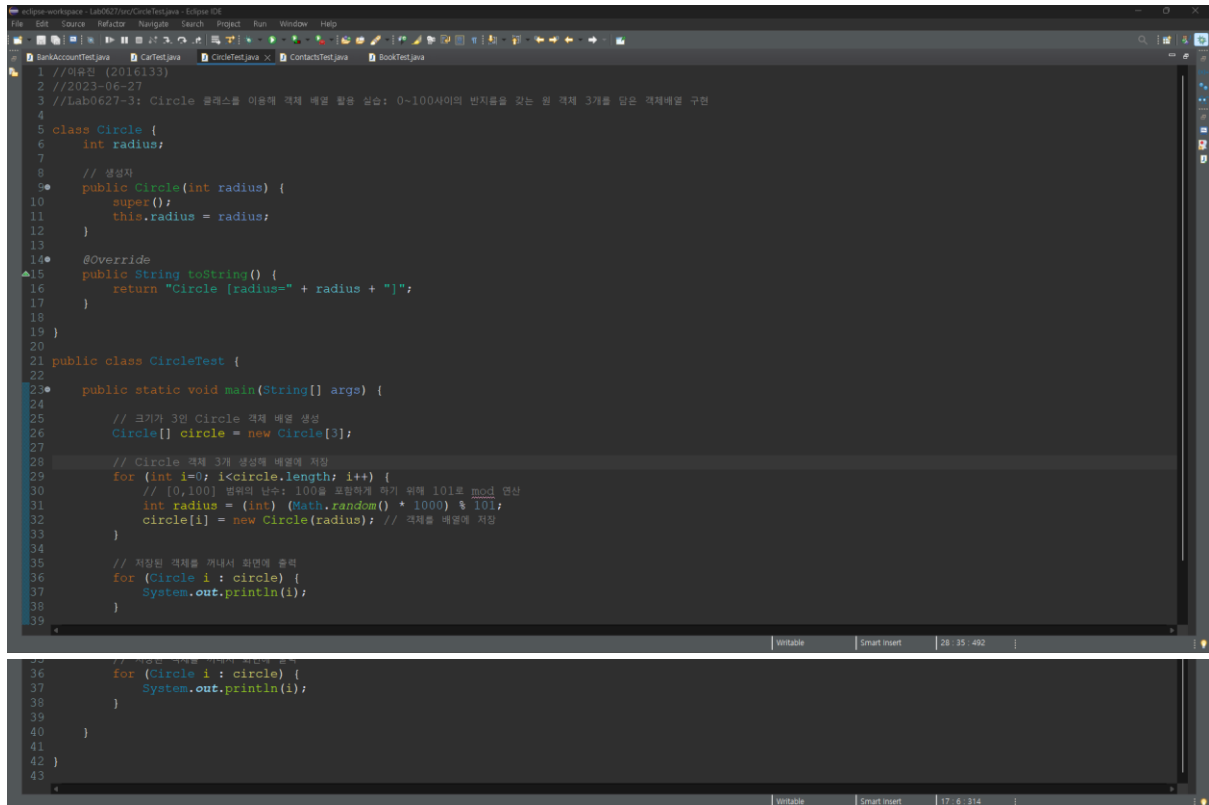
2. 실습 2 CarTest.java

```
1 //이유진 (2016133)
2 //2023-06-27
3 //Lab0627-2: Car 클래스를 이용해 static변수와 메소드 실습
4
5 class Car {
6     private String model;
7     private String make;
8     private static int numberOfCars = 0; //생산된 차의 개수를 저장하는 정적 변수
9
10    // 생성자
11    public Car(String model, String make) {
12        super();
13        this.model = model;
14        this.make = make;
15        numberOfCars++; //생산된 차의 수 증가
16    }
17
18    // 설정자, 접근자
19    public String getModel() {
20        return model;
21    }
22
23    public void setModel(String model) {
24        this.model = model;
25    }
26
27    public String getMake() {
28        return make;
29    }
30
31    public void setMake(String make) {
32        this.make = make;
33    }
34
35    // 접근자 getNumberOfCars() 대신 정적 변수를 외부로 반환해주는 정적 메소드 사용
36    public static int getNumberOfCars() {
37        return numberOfCars;
38    }
39
40 }
```

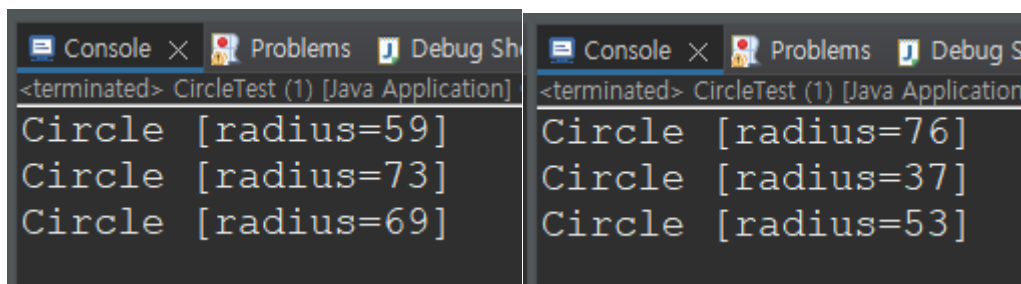
```
18    // 설정자, 접근자
19    public String getModel() {
20        return model;
21    }
22
23    public void setModel(String model) {
24        this.model = model;
25    }
26
27    public String getMake() {
28        return make;
29    }
30
31    public void setMake(String make) {
32        this.make = make;
33    }
34
35    // 접근자 getNumberOfCars() 대신 정적 변수를 외부로 반환해주는 정적 메소드 사용
36    public static int getNumberOfCars() {
37        return numberOfCars;
38    }
39
40    // 설정자 setNumberOfCars()는 생성자에서 numberOfCars를 올리므로 사용하지 않음
41    // 별도의 toString() 메소드 필요하지 않음
42
43 }
44
45 public class CarTest {
46
47     public static void main(String[] args) {
48
49         new Car("3SERIES", "BENZ");
50         new Car("3SERIES", "BENZ");
51         new Car("3SERIES", "BENZ");
52         System.out.println("총 " + Car.getNumberOfCars() + "대의 자동차가 생산되었습니다.");
53     }
54 }
55
56 }
```

```
Console Problems Debug Shell
<terminated> CarTest [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (2023. 6. 28. 오후 3:00)
총 3대의 자동차가 생산되었습니다.
```

3. 실습 3 CircleTest.java



```
1 //이유진 (2016133)
2 //2023-06-27
3 //Lab0627-3: Circle 클래스를 이용해 객체 배열 필드: 0~100사이의 반지름을 갖는 원 객체 3개를 담은 객체배열 구현
4
5 class Circle {
6     int radius;
7
8     // 생성자
9     public Circle(int radius) {
10         super();
11         this.radius = radius;
12     }
13
14     @Override
15     public String toString() {
16         return "Circle [radius=" + radius + "]";
17     }
18 }
19
20
21 public class CircleTest {
22
23     public static void main(String[] args) {
24
25         // 크기가 3인 Circle 객체 배열 생성
26         Circle[] circle = new Circle[3];
27
28         // Circle 객체 3개 생성해 배열에 저장
29         for (int i=0; i<circle.length; i++) {
30             // [0,100] 범위의 난수: 100을 포함하게 하기 위해 101로 mod 연산
31             int radius = (int) (Math.random() * 1000) % 101;
32             circle[i] = new Circle(radius); // 객체를 배열에 저장
33         }
34
35         // 저장된 객체를 꺼내서 화면에 출력
36         for (Circle i : circle) {
37             System.out.println(i);
38         }
39     }
40 }
41
42
43
```



Console Output
<terminated> CircleTest (1) [Java Application]
Circle [radius=59]
Circle [radius=73]
Circle [radius=69]

Console Output
<terminated> CircleTest (1) [Java Application]
Circle [radius=76]
Circle [radius=37]
Circle [radius=53]

4. 실습 4 ContactsTest.java

```
1 //이유진 (2016133)
2 //2023-06-27
3 //Lab0627-4: Contacts 클래스와 ArrayList를 활용한 연락처 저장 프로그램
4
5 import java.util.ArrayList;
6 import java.util.Scanner;
7
8 class Contacts {
9     private String name;
10    private String tel;
11    private String email;
12    private static int count = 0; // 지인들의 수를 저장하는 정적 변수
13
14    // 생성자
15    public Contacts(String name, String tel, String email) {
16        super();
17        this.name = name;
18        this.tel = tel;
19        this.email = email;
20        count++; // 연락처 수 증가
21    }
22
23    // 설정자, 접근자
24    public String getName() {
25        return name;
26    }
27
28    public void setName(String name) {
29        this.name = name;
30    }
31
32    public String getTel() {
33        return tel;
34    }
35
36    public void setTel(String tel) {
37        this.tel = tel;
38    }
39
40    public String getEmail() {
41        return email;
42    }
43
44    public void setEmail(String email) {
45        this.email = email;
46    }
47
48    public static int getCount() {
49        return count;
50    }
51
52    // setCount 설정자는 생성자에서 count를 올리므로 사용하지 않음
53
54    @Override
55    public String toString() {
56        return name + "의 전화번호:" + tel + " 이메일: " + email;
57    }
58 }
59
60 public class ContactsTest {
61
62
63    public static void main(String[] args) {
64
65        Scanner scan = new Scanner(System.in);
66        ArrayList<Contacts> cArr = new ArrayList<>(); // 동적으로 추가하기 위해 ArrayList 사용
67
68        System.out.println("연락처를 입력하십시오 (종료 -1)");
69        while (true) {
70
71            // 사용자로부터 정보를 입력받음
72            System.out.print("이름과 전화번호, 이메일을 입력하십시오: ");
73            String name = scan.next();
74            if (name.equals("-1")) { // 종료 조건
75                System.out.println("지인들의 수는 " + Contacts.getCount() + "입니다.");
76                break;
77            }
78            String tel = scan.next();
79            String email = scan.next();
80            Contacts c = new Contacts(name, tel, email);
81            cArr.add(c);
82        }
83
84        System.out.println("등록된 연락처 목록:");
85        for (Contacts c : cArr) {
86            System.out.println(c.toString());
87        }
88    }
89 }
```

```
63 public static void main(String[] args) {
64     Scanner scan = new Scanner(System.in);
65     ArrayList<Contacts> cArr = new ArrayList<>(); // 동적으로 추가하기 위해 ArrayList 사용
66
67     System.out.println("연락처를 입력하시오 (종료 -1)");
68     while (true) {
69         // 사용자로부터 정보를 입력받음
70         System.out.print("이름과 전화번호, 이메일을 입력하시오: ");
71         String name = scan.next();
72         if (name.equals("-1")) { // 종료 조건
73             System.out.println("지인들의 수는 " + Contacts.getCount() + "입니다");
74             break;
75         }
76         String tel = scan.next();
77         String email = scan.next();
78
79         Contacts c = new Contacts(name, tel, email);
80         if (Contacts.getCount() > 100) { // 종료 조건: 최대 100개 저장
81             System.out.println("저장 개수 초과!");
82             break;
83         }
84         cArr.add(c); // ArrayList에 추가
85     }
86
87     System.out.print("검색할 이름을 입력하시오: ");
88     String key = scan.next();
89     for (int i = 0; i < cArr.size(); i++) {
90         if (key.equals(cArr.get(i).getName())) { // ArrayList에 key를 이름으로 갖는 연락처가 있는 경우
91             System.out.println(cArr.get(i));
92         }
93     }
94     scan.close();
95 }
96 }
97 }
98 }
99 }
100 }
```

```
Console × Problems Debug Shell
<terminated> ContactsTest [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (2023. 6. 28. 오전 1:42:21 - 오전 1:43:30)
연락처를 입력하시오 (종료 -1)
이름과 전화번호, 이메일을 입력하시오: Kim 010-1111-2222 kim@java.com
이름과 전화번호, 이메일을 입력하시오: Lee 010-2222-3333 lee@java.com
이름과 전화번호, 이메일을 입력하시오: Kang 010-5555-7777 kang@java.com
이름과 전화번호, 이메일을 입력하시오: Choe 010-1234-5678 choe@java.com
이름과 전화번호, 이메일을 입력하시오: -1
지인들의 수는 4입니다
검색할 이름을 입력하시오: Kang
Kang의 전화번호: 010-5555-7777 이메일: kang@java.com
```

The image displays two screenshots of the Eclipse IDE, showing the implementation of a book management system in Java.

Top Screenshot: Book.java

```
1 //이유진 (20161133)
2 //2023-06-27
3 //Lab0627-5: Book 클래스와 ArrayList를 활용한 책과 평점 저장 프로그램
4
5 import java.util.ArrayList;
6 import java.util.Scanner;
7
8 class Book {
9     private String title;
10    private int score;
11    private static int cnt = 0; // 현재까지 등록된 책의 수
12
13    // 생성자
14    public Book(String title, int score) {
15        super();
16        this.title = title;
17        this.score = score;
18        cnt++;
19    }
20
21    // 필명자, 점수자
22    public String getTitle() {
23        return title;
24    }
25
26    public void setTitle(String title) {
27        this.title = title;
28    }
29
30    public int getScore() {
31        return score;
32    }
33
34    public void setScore(int score) {
35        this.score = score;
36    }
37
38    public static int getCnt() {
39        return cnt;
40    }
41 }
```

Bottom Screenshot: BookTest.java

```
38 public static int getCnt() {
39     return cnt;
40 }
41
42 //setCnt() 필명자는 생성자에서 cnt를 올리므로 사용하지 않음
43
44 @Override
45 public String toString() {
46     return "Book [title=" + title + ", score=" + score + "]";
47 }
48
49 }
50
51 public class BookTest {
52
53     public static void main(String[] args) {
54
55         Scanner scan = new Scanner(System.in);
56         ArrayList<Book> bookList = new ArrayList<>(); // 등록으로 추가하기 위해 ArrayList 사용
57
58         while (true) {
59             System.out.println("=====");
60             System.out.println("1. 책 등록");
61             System.out.println("2. 책 검색");
62             System.out.println("3. 모든 책 출력");
63             System.out.println("4. 종료");
64             System.out.println("=====");
65             System.out.print("메뉴를 선택하세요: ");
66             int menu = scan.nextInt();
67             scan.nextLine(); // 메뉴 입력 후 입력스트림에 남아있는 줄바꿈문자를 지우기 위한
68
69             switch (menu) {
70                 case 1:
71                     System.out.print("책 제목: ");
72                     String t = scan.nextLine();
73                     System.out.print("책 평점: ");
74                     int s = scan.nextInt();
75
76                     Book b = new Book(t, s); // 2018.11.11
```

```

172 String t = scan.nextLine();
173 System.out.print("책 평점: ");
174 int s = scan.nextInt();
175
176 Book b = new Book(t, s); // 객체 생성
177 bookList.add(b); // ArrayList에 추가
178 break;
179
180 case 2:
181     System.out.print("책 제목: ");
182     String key = scan.next();
183     for (int i = 0; i < bookList.size(); i++) {
184         if (key.equals(bookList.get(i).getTitle())) // ArrayList에 key를 이름으로 갖는 책이 있는 경우
185             System.out.println(bookList.get(i));
186     }
187     break;
188
189 case 3:
190     for (int i = 0; i < bookList.size(); i++) { // ArrayList를 순회하며 출력
191         System.out.println(bookList.get(i));
192     }
193     System.out.println("현재까지 등록된 책의 수: " + Book.getCnt());
194     break;
195
196 case 4:
197     System.out.println("종료!");
198     break;
199
200 // default 선택
201 }
202 if (menu == 4)
203     break;
204 }
205 scan.close();
206 }
207 }
208 }
209 }
210 }

```

```

<terminated> BookTest (1) [Java Application] C:\Program Files\Java\jdk-1
=====
1. 책 등록
2. 책 검색
3. 모든 책 출력
4. 종료
=====
메뉴를 선택하시오: 1
책 제목: 자바 프로그래밍
책 평점: 8
=====
1. 책 등록
2. 책 검색
3. 모든 책 출력
4. 종료
=====
메뉴를 선택하시오: 1
책 제목: Python 기초
책 평점: 9
=====
1. 책 등록
2. 책 검색
3. 모든 책 출력
4. 종료
=====
메뉴를 선택하시오: 3
Book [title=자바 프로그래밍, score=8]
Book [title=Python 기초, score=9]
=====
1. 책 등록
2. 책 검색
3. 모든 책 출력
4. 종료
=====
메뉴를 선택하시오: 2
책 제목: 자바 프로그래밍
Book [title=자바 프로그래밍, score=8]
=====
1. 책 등록
2. 책 검색
3. 모든 책 출력
4. 종료
=====
메뉴를 선택하시오: 4
종료!
=====

```

```

4. 종료
=====
메뉴를 선택하시오: 3
Book [title=자바 프로그래밍, score=8]
Book [title=Python 기초, score=9]
=====
1. 책 등록
2. 책 검색
3. 모든 책 출력
4. 종료
=====
메뉴를 선택하시오: 2
책 제목: 자바 프로그래밍
Book [title=자바 프로그래밍, score=8]
=====
1. 책 등록
2. 책 검색
3. 모든 책 출력
4. 종료
=====
메뉴를 선택하시오: 4
종료!
=====

```