

Java 프로그래밍

6주차 과제

학 과	소프트웨어전공
학 번	2126097
이 름	이동민
제 출 일	2024. 04. 14.

문제 1번: 아래 실행 결과와 같이 출력하는 다음 main()을 가진 Song 클래스를 작성하라.
Song 클래스는 노래 제목 title 필드, 생성자, getTitle() 메소드로 구성된다.

```
public static void main(String[] args) {  
    Song mySong = new Song("Nessun Dorma");  
    Song yourSong = new Song("공주는 잠 못 이루고");  
    System.out.println("내 노래는 " + mySong.getTitle());  
    System.out.println("너 노래는 " + yourSong.getTitle());  
}
```

```
내 노래는 Nessun Dorma  
너 노래는 공주는 잠 못 이루고
```

Source Code

```
public class Song {  
  
    String title;  
  
    Song(String title) {  
        this.title = title;  
    }  
  
    String getTitle() {  
        return title;  
    }  
  
    public static void main(String[] args) {  
        Song mySong = new Song("Nessun Dorma");  
        Song yourSong = new Song("공주는 잠 못 이루고");  
  
        System.out.println("내 노래는 " + mySong.getTitle());  
        System.out.println("너 노래는 " + yourSong.getTitle());  
  
    }  
}
```

출력 결과

```
내 노래는 Nessun Dorma  
너 노래는 공주는 잠 못 이루고
```

문제 2번: 다음은 이름(name 필드)과 전화번호(tel 필드)를 가진 Phone 클래스이다. 이름과 전화번호를 입력받아 2개의 Phone 객체를 생성하고, 출력하는 main() 메소드를 작성하라.

```
public class Phone {  
    private String name, tel;  
    public Phone(String name, String tel) {  
        this.name = name;  
        this.tel = tel;  
    }  
  
    public String getName() { return name; }  
    public String getTel() { return tel; }  
}
```

```
이름과 전화번호 입력 >>스폰지밥 333-3333  
이름과 전화번호 입력 >>징징이 999-9999  
스폰지밥의 번호 333-3333  
징징이의 번호 999-9999
```

Source Code는 다음 페이지에 있습니다.

```
import java.util.Scanner;

public class Phone {
    private String name, tel;

    public Phone(String name, String tel) {
        this.name = name;
        this.tel = tel;
    }

    public String getName() {
        return name;
    }
    public String getTel() {
        return tel;
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("이름과 전화번호 입력 >> ");
        Phone phone1 = new Phone(sc.next(), sc.next());

        System.out.print("이름과 전화번호 입력 >> ");
        Phone phone2 = new Phone(sc.next(), sc.next());

        System.out.printf("%s의 번호 %s\n", phone1.getName(), phone1.getTel());
        System.out.printf("%s의 번호 %s\n", phone2.getName(), phone2.getTel());
    }
}
```

출력 결과

```
이름과 전화번호 입력 >> 이동민 010-7273-2214
이름과 전화번호 입력 >> 이동수 010-1234-5678
이동민의 번호 010-7273-2214
이동수의 번호 010-1234-5678
```

문제 3번: 사각형을 표현하는 다음 Rect 클래스를 활용하여, Rect 객체 배열을 생성하고, 사용자로부터 4개의 사각형을 입력받아 배열에 저장한 뒤, 배열을 검색하여 사각형 면적의 합을 출력하는 main() 메소드를 가진 RectArray 클래스를 작성하라.

```
class Rect {  
    private int width, height;  
    public Rect(int width, int height) {  
        this.width = width;  
        this.height = height;  
    }  
    public int getArea() { return width*height; }  
}
```

```
1 너비와 높이 >>3 5  
2 너비와 높이 >>3 9  
3 너비와 높이 >>2 7  
4 너비와 높이 >>9 5  
저장하였습니다...  
사각형의 전체 합은 101
```

Source Code는 다음 페이지에 있습니다.

```

import java.util.Scanner;

class Rect {
    private int width, height;

    public Rect(int width, int height) {
        this.width = width;
        this.height = height;
    }

    public int getArea() { return width*height; }
}

public class RectArray {

    public static void main(String[] args) {
        Rect[] rects = new Rect[4];
        Scanner sc = new Scanner(System.in);

        for (int i = 0; i < rects.length; i++) {
            System.out.printf("%d 너비와 높이 >> ", i + 1);
            rects[i] = new Rect(sc.nextInt(), sc.nextInt());
        }

        System.out.println("저장하였습니다...");

        int sum = 0;
        for (int i = 0; i < rects.length; i++) {
            sum += rects[i].getArea();
        }

        System.out.printf("사각형의 전체 합은 %d\n", sum);
    }
}

```

출력 결과

```

1 너비와 높이 >> 4 6
2 너비와 높이 >> 4 10
3 너비와 높이 >> 3 8
4 너비와 높이 >> 10 6
저장하였습니다...
사각형의 전체 합은 148

```

문제 4번: 이름(name)과 전화번호(tel) 필드, 생성자 및 필요한 메소드를 가진 Phone 클래스를 작성하고, 다음 실행 사례와 같이 작동하도록 main()을 가진 PhoneManager 클래스를 작성하라. 한 사람의 전화번호는 하나의 Phone 객체로 다룬다.

Source Code는 다음 페이지에 있습니다.

```
import java.util.Scanner;

class Phone04 { // 2번 문제와 클래스명 중복을 피하기 위해 Phone04 사용
    private String name;
    private String tel;

    Phone04(String name, String tel) {
        this.name = name;
        this.tel = tel;
    }

    public String getName() {
        return name;
    }
    public String getTel() {
        return tel;
    }
}

public class PhoneManager {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("인원수 >> ");
        int people = sc.nextInt();

        Phone04[] phones = new Phone04[people];

        for (int i = 0; i < phones.length; i++) {
            System.out.print("이름과 전화번호(번호는 연속적으로 입력) >> ");
            phones[i] = new Phone04(sc.next(), sc.next());
        }
        System.out.println("저장되었습니다...");

        while (true) {
            System.out.print("검색할 이름 >> ");
            String searchName = sc.next();

            for (int i = 0; i < phones.length; i++) {
                if (phones[i].getName().equals(searchName)) {
                    System.out.printf("%s의 번호는 %s 입니다.\n", phones[i].getName(), phones[i].getTel());
                }
            }

            if (searchName.equals("exit")) {
                System.out.println("프로그램을 종료합니다...");
                break;
            }
        }
    }
}
```


출력 결과

```
인원수 >> 3
이름과 전화번호(번호는 연속적으로 입력) >> 이동민 010-7273-2214
이름과 전화번호(번호는 연속적으로 입력) >> 이동수 010-1234-5678
이름과 전화번호(번호는 연속적으로 입력) >> 이동목 010-8765-4321
저장되었습니다...
검색할 이름 >> 이동민
이동민의 번호는 010-7273-2214 입니다.
검색할 이름 >> 이동수
이동수의 번호는 010-1234-5678 입니다.
검색할 이름 >> exit
프로그램을 종료합니다...
```

문제 5번: CircleManager는 static 메소드를 가진 클래스이다. StaticTest 클래스는 static 메소드를 활용하는 사례를 보여준다. 실행 결과를 참고하여 코드를 완성하라.

Source Code

```
class Circle {
    private int radius;
    public Circle(int radius) { this.radius = radius; }
    public int getRadius() { return this.radius; }
    public void setRadius(int radius) { this.radius = radius; }
}

class CircleManager {
    static void copy(Circle src, Circle dest) {
        dest.setRadius(src.getRadius());
    }
    static boolean equals(Circle a, Circle b) {
        if (a.getRadius() == b.getRadius())
            return true;
        else
            return false;
    }
}

public class StaticTest {

    public static void main(String[] args) {
        Circle pizza = new Circle(5);
        Circle waffle = new Circle(1);

        boolean res = CircleManager.equals(pizza, waffle);
        if (res == true)
            System.out.println("pizza와 waffle 크기 같음");
        else
            System.out.println("pizza와 waffle 크기 다름");

        CircleManager.copy(pizza, waffle);
        res = CircleManager.equals(pizza, waffle);
        if (res == true)
            System.out.println("pizza와 waffle 크기 같음");
        else
            System.out.println("pizza와 waffle 크기 다름");
    }
}
```

출력 결과

```
pizza와 waffle 크기 다름
pizza와 waffle 크기 같음
```

Box의 draw()는 fill 필드에 지정된 문자로 자신을 그린다. 실행 결과를 보면서, 코드를 완성하라.

Source Code

```
public class Box {
    private int width, height;
    private char fillChar;
    public Box() {
        this(10, 1);
    }
    public Box(int width, int height) {
        this.width = width;
        this.height = height;
    }
    public void draw() {
        for (int i = 0; i < height; i++) {
            for (int j = 0; j < width; j++) {
                System.out.print(fillChar);
            }
            System.out.println();
        }
    }
    public void fill(char c) {
        fillChar = c;
    }

    public static void main(String[] args) {
        Box a = new Box();
        Box b = new Box(20, 3);
        a.fill('*');
        b.fill('%');
        a.draw();
        b.draw();
    }
}
```

출력 결과

```
*****  
0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/  
/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/  
  
0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/  
/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/  
  
0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/  
/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/
```

보너스 문제: 두 사람이 번갈아 하는 겜블링 게임을 만들어 보자. 0에서 2 사이의 정수 3개를 랜덤하게 발생시켜 모두 같으면 승리한다. 선수는 Player 클래스로 표현한다. 실행 결과를 참고하여 다음 코드를 완성하라.

Source Code는 다음 페이지에 있습니다.

```
import java.util.Scanner;

class PlayerBonus {
    private String name;

    public PlayerBonus(String name) {
        this.name = name;
    }

    public String getName() {
        return name;
    }
}

public class GamblingGame {

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

        PlayerBonus[] players = new PlayerBonus[2];

        for (int i = 0; i < players.length; i++) {
            System.out.print("선수 이름 입력 >> ");
            players[i] = new PlayerBonus(sc.next());
        }

        int n = 0;

        while (true) {
            System.out.printf("%s씨, <Enter 외 아무키나 치세요>", players[n].getName());
            sc.next();

            int[] val = new int[3];
            for (int i = 0; i < val.length; i++) {
                val[i] = (int)(Math.random() * 3);
                System.out.print(val[i] + "\\t");
            }
            System.out.println();

            if (val[0] == val[1] && val[1] == val[2]) {
                System.out.printf("%s이(가) 승리하였습니다.", players[n].getName());
                break;
            }
            n++;
            n = n % 2;
        }
    }
}
```

출력 결과

선수 이름 입력 >> A

선수 이름 입력 >> B

A씨, <Enter 외 아무키나 치세요>o

1 0 0

B씨, <Enter 외 아무키나 치세요>o

2 1 2

A씨, <Enter 외 아무키나 치세요>o

1 1 1

A이(가) 승리하였습니다.