



소프트웨어프로젝트 02분반

project2 report

소프트웨어학부

20181269 박현우

I. 소스 프로그램

```
package rectangle;
```

```
public class Rectangle {  
    float width;  
    float height;
```

```
    Rectangle(float newWidth, float newHeight) {  
        width = newWidth;  
        height = newHeight;  
    }
```

```
    public void print() {  
        System.out.println("width = " + width + ", height = " +  
            height);  
    }
```

```
    public boolean equal(Rectangle r1, Rectangle r2) {  
        if ((r1.width == r2.width) && (r1.height == r2.height))  
            return true;  
        else  
            return false;  
    }
```

```
    public float compare(Rectangle r1, Rectangle r2) {  
        return r1.width*r1.height - r2.width*r2.height;  
    }
```

```
    public void resize(Rectangle r, float ratio) {  
        r.width = ratio;  
        r.height = ratio;  
    }
```

```
}
```

```

public class TestRectangle {
    public static void main(String[] args) {
        Rectangle r1 = new Rectangle(1.0f, 1.0f);
        Rectangle r2 = new Rectangle(1.0f, 2.0f);
        Rectangle r3 = new Rectangle(2.0f, 1.0f);

        if ( r2.equal(r2, r3) )
            System.out.println("two rectangles are equal.");
        else
            System.out.println("two rectangles are different.");

        if ( r2.compare(r2, r3) > 0 )
            System.out.println("the first is larger than the
                                second.");
        else if ( r2.compare(r2, r3) < 0 )
            System.out.println("the first is smaller than the
                                second.");
        else
            System.out.println("two are the same in area.");

        r1.print();
        r1.resize(r1, 2.0f);
        r1.print();
    }
}

```

II. 실행 화면

```
1 package rectangle;
2
3 public class Rectangle {
4     float width;
5     float height;
6
7     Rectangle(float newWidth, float newHeight) {
8         width = newWidth;
9         height = newHeight;
10    }
11
12    public void print() {
13        System.out.println("width = " + width + ", height = " + height);
14    }
15
16    public boolean equal(Rectangle r1, Rectangle r2) {
17        if ((r1.width == r2.width) && (r1.height == r2.height))
18            return true;
19        else
20            return false;
21    }
22
23    public float compare(Rectangle r1, Rectangle r2) {
24        return r1.width*r1.height - r2.width*r2.height;
25    }
26
27    public void resize(Rectangle r, float ratio) {
28        r.width = ratio;
29        r.height = ratio;
30    }
31 }
```

```
1 package rectangle;
2
3 public class TestRectangle {
4     public static void main(String[] args) {
5         Rectangle r1 = new Rectangle(1.0f, 1.0f);
6         Rectangle r2 = new Rectangle(1.0f, 2.0f);
7         Rectangle r3 = new Rectangle(2.0f, 1.0f);
8
9         if ( r2.equal(r2, r3) )
10            System.out.println("two rectangles are equal.");
11        else
12            System.out.println("two rectangles are different.");
13
14        if ( r2.compare(r2, r3) > 0 )
15            System.out.println("the first is larger than the second.");
16        else if ( r2.compare(r2, r3) < 0 )
17            System.out.println("the first is smaller than the second.");
18        else
19            System.out.println("two are the same in area.");
20
21        r1.print();
22        r1.resize(r1, 2.0f);
23        r1.print();
24    }
25 }
```

Problems Javadoc Declaration Console

<terminated> TestRectangle [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (2019. 3. 31. 오후 9:48:17)

two rectangles are different.
two are the same in area.
width = 1.0, height = 1.0
width = 2.0, height = 2.0

III. 설계 노트

직사각형을 만들어내는 class인 Rectangle을 정의하고 그 속성으로 밑변을 나타내는 float형 변수 width, 높이를 나타내는 float형 변수 height를 추가한다. Rectangle 객체를 만들기 위한 생성자를 선언한다.

직사각형의 밑변과 높이 값을 출력하는 함수 print(), 두 직사각형이 같은지를 판별하는 함수 equal(), 두 직사각형의 넓이를 비교하는 함수 compare(), 직사각형의 밑변과 높이 값을 변경하는 함수 resize()를 정의한다.

새로운 class인 TestRectangle을 정의하고 main 메소드에 Rectangle 객체 r1, r2, r3를 만든다. equal 함수를 이용하여 r2와 r3가 같은지를 판별한 후 그 결과를 출력하고, compare 함수를 이용하여 r2와 r3의 넓이를 비교한 후 그 결과를 출력한다. 또한 r1의 밑변과 높이 값을 출력한 후에 resize 함수를 이용하여 r1의 밑변과 높이 값을 변경한 후 다시 r1의 밑변과 높이 값을 출력한다.

C로 작성된 프로그램을 Java로 옮기면서 C와 Java의 차이점에 대해 알 수 있었다. 기존 C 프로그램에는 rectangle이 구조체로 정의되어있고, 관련 함수가 따로 존재했는데 이것들을 Java에서는 클래스 내부에 부속되어 존재하도록 하였다. 그리고 C 프로그램에서 포인터가 사용되었는데, Java 프로그램에서는 reference value를 이용하였다. 그 외에 Java는 C와 달리 객체 지향 방식으로 프로그래밍을 하기 때문에 이를 중점적으로 생각하여 코드를 작성하였다.

IV. Java와 C 프로그램 비교

Java	C
<pre> package rectangle; public class Rectangle { float width; float height; Rectangle(float newWidth, float newHeight) { width = newWidth; height = newHeight; } public void print() { System.out.println("width = " + width + ", height = " + height); } public boolean equal(Rectangle r1, Rectangle r2) { if ((r1.width == r2.width) && (r1.height == r2.height)) return true; else return false; } public float compare(Rectangle r1, Rectangle r2) { return r1.width*r1.height - r2.width*r2.height; } public void resize(Rectangle r, float ratio) { r.width = ratio; r.height = ratio; } } public class TestRectangle { public static void main(String[] args) { Rectangle r1 = new Rectangle(1.0f, 1.0f); Rectangle r2 = new Rectangle(1.0f, 2.0f); Rectangle r3 = new </pre>	<pre> #include <stdio.h> struct rectangle { float width; float height; }; void print(struct rectangle r) { printf("width = %f , height = %f \n", r.width, r.height); } bool equal(struct rectangle r1, struct rectangle r2) { if ((r1.width == r2.width) && (r1.height == r2.height)) return true; else return false; } float compare(struct rectangle r1, struct rectangle r2) { return r1.width*r1.height - r2.width*r2.height; } void resize(struct rectangle *r, float ratio) { r->width *= ratio; r->height *= ratio; } main() { struct rectangle r1 = {1.0, 1.0}; struct rectangle r2 = {1.0, 2.0}; struct rectangle r3 = {2.0, 1.0}; if (equal(r2, r3)) printf("two rectangles are equal. \n"); else printf("two rectangles are different. \n"); if (compare(r2, r3) > 0) </pre>

<pre> Rectangle(2.0f, 1.0f); if (r2.equal(r2, r3)) System.out.println("two rectangles are equal."); else System.out.println("two rectangles are different."); if (r2.compare(r2, r3) > 0) System.out.println("the first is larger than the second."); else if (r2.compare(r2, r3) < 0) System.out.println("the first is smaller than the second."); else System.out.println("two are the same in area."); r1.print(); r1.resize(r1, 2.0f); r1.print(); } } </pre>	<pre> printf("the first is larger than the second. \n"); else if (compare(r2, r3) < 0) printf("the first is smaller than the second. \n"); else printf("two are the same in area. \n"); print(r1); resize (&r1, 2.0); print(r1); } } </pre>
--	---

항목	Java	C
구성 단위	클래스 단위로 되어있어 import를 사용해 포함시킴	소스 파일, 헤더 파일 등 파일 단위로 되어있어 #include를 사용해 포함시킴
데이터 관리	클래스를 사용함	구조체, 공용체를 사용함
메모리 접근 방식	참조를 사용함	포인터를 이용하여 메모리에 접근함
함수	main 함수가 클래스에 부속됨	main 함수가 독립적으로 존재함
프로그래밍 방식	객체 지향 (객체를 먼저 정의하고 그 객체가 가지고 있는 정보와 기능을 정의한 다음에 객체 간의 상호작용을 기술하여 문제를 해결함)	절차 지향 (문제를 해결하기 위해 필요한 여러 기능을 순차적으로 나열하여 기술함)

V. 자체평가표

평가 항목	학생 자체 평가 (리포트 해당 부분 표시 및 간단한 의견)	평가 (빈칸)	점수 (빈칸)
Java Style - 동작? - 요구사항 만족?	C로 작성된 직사각형 관련 프로그램과 동일한 기능을 하는 프로그램을 Java로 구현해냈다. class Rectangle에 메소드들을 포함시켰고, 프로그램의 테스트 동작을 할 때에는 class TestRectangle을 사용하였다.		
두 방법 비교	C와 Java의 가장 큰 차이라고 할 수 있는 객체지향/비객체지향을 중점적으로 비교해보았다. 객체지향에 대한 개념을 완벽하게 설명하지는 못하였다.		
기타			
총평/계	C로 작성된 프로그램을 Java로 새롭게 작성하여 결과를 도출해냈지만, 아직 Java의 OOP에 대해 완전한 이해를 하지 못한 것 같아 아쉽다.		