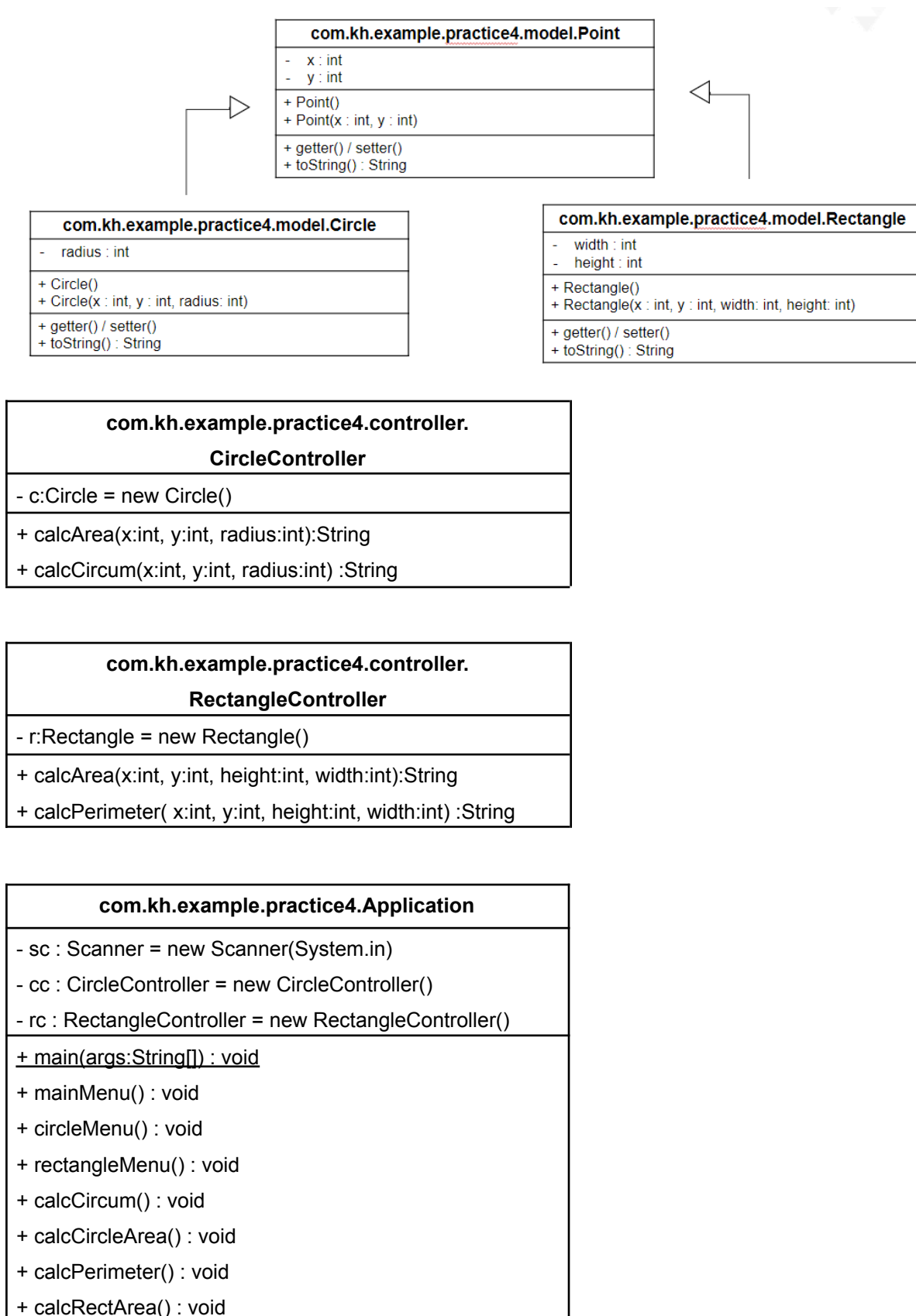


다음과 같은 조건을 만족하는 프로그램을 작성 하시오

1. 구현 클래스 다이어그램 (Class Diagram)



2. 구현 클래스 설명

Package명	Class명	Method	설명
com.kh.example .practice4.controller	CircleController	+calcArea(x:int, y:int, radius:int):String	받은 매개변수를 이용하여 필드들을 초기화하고 초기화한 정보와 면적 반환 // 면적 : $PI * 반지름 * 반지름$
		+calcCircum(x:int, y:int, radius:int):String	받은 매개변수를 이용하여 필드들을 초기화하고 초기화한 정보와 둘레 반환 // 둘레 : $PI * 반지름 * 2$
	RectangleController	+calcArea(x:int, y:int, height:int, width:int):String	받은 매개변수를 이용하여 필드들을 초기화하고 초기화한 정보와 면적 반환 // 면적 : 너비 * 높이
		+calcPerimeter(x:int, y:int, height:int, width:int):String	받은 매개변수를 이용하여 필드들을 초기화하고 초기화한 정보와 둘레 반환 // 둘레 : $2 * (너비 + 높이)$
com.kh.example .practice4	Application	<u>+main(args:String[]):void</u>	mainMenu() 실행
		+mainMenu():void	메인메뉴 출력
		+circleMenu():void	원 메뉴 출력
		+rectangleMenu():void	사각형 메뉴 출력
		+calcCircum():void	x, y 좌표와 반지름을 받아 원의 정보와 원 둘레 출력
		+calcCircleArea():void	x, y 좌표와 반지름을 받아 원의 정보와 원 넓이 출력
		+calcPerimeter():void	x, y 좌표와 높이, 너비를 받아 사각형의 정보와 사각형 둘레 출력
		+calcRectArea():void	x, y 좌표와 높이, 너비를 받아 사각형의 정보와 사각형 넓이 출력

3. class 구조

```
public class Application {  
    public static void main(String[] args) {  
        // PointMenu의 mainMenu() 실행  
    }  
  
    public void mainMenu() {  
        ===== 메뉴 =====  
        1. 원                circleMenu()  
        2. 사각형            rectangleMenu()  
        9. 끝내기  
        메뉴 번호 :  
    }  
  
    public void circleMenu(){  
        ===== 원 메뉴 =====  
        1. 원 둘레            calcCircum()  
        2. 원 넓이            calcCircleArea()  
        9. 메인으로  
        메뉴 번호 :  
    }  
  
    public void retangleMenu() {  
        ===== 메뉴 =====  
        1. 사각형 둘레        calcPerimeter()  
        2. 사각형 넓이        calcRectArea()  
        3. 메인으로  
        메뉴 번호 :  
    }  
}
```

```

public void calcCircum(){
    x 좌표 :
    y 좌표 :
    반지름 :
    // 데이터를 CircleController(cc)의 calcCircum()의 매개변수로 보내 반환 값 출력
}
public void calcCircleArea(){
    x 좌표 :
    y 좌표 :
    반지름 :
    // 데이터를 CircleController(cc)의 calcArea()의 매개변수로 보내 반환 값 출력
}
public void calcPerimeter(){
    x 좌표 :
    y 좌표 :
    높이 :
    너비 :
    // 데이터를 RectangleController(rc) calcPerimeter()의 매개변수로 보내 반환값 출력
}
public void calcRectArea(){
    x 좌표 :
    y 좌표 :
    높이 :
    너비 :
    // 데이터를 RectangleController(rc) calcRectArea()의 매개변수로 보내 반환값 출력
}
}

```

5. 실행 결과 화면

```
===== 메뉴 =====
1. 원
2. 사각형
9. 끝내기
메뉴 번호 : 1
===== 원 메뉴 =====
1. 원 둘레
2. 원 넓이
9. 예인으로
메뉴 번호 : 1
x 좌표 : 2
y 좌표 : 4
반지름 : 1
위치 : (2, 4), 반지름 : 1 / 둘레 : 6.283185307179586
===== 메뉴 =====
1. 원
2. 사각형
9. 끝내기
메뉴 번호 : 1
===== 원 메뉴 =====
1. 원 둘레
2. 원 넓이
9. 예인으로
메뉴 번호 : 2
x 좌표 : 2
y 좌표 : 4
반지름 : 1
위치 : (2, 4), 반지름 : 1 / 넓이 : 3.141592653589793
===== 메뉴 =====
1. 원
2. 사각형
9. 끝내기
메뉴 번호 : 2
===== 사각형 메뉴 =====
1. 사각형 둘레
2. 사각형 넓이
9. 예인으로
메뉴 번호 : 1
x 좌표 : 2
y 좌표 : 4
높이 : 1
너비 : 3
위치 : (2, 4), 너비 : 3, 높이 : 1 / 둘레 : 8
===== 메뉴 =====
1. 원
2. 사각형
9. 끝내기
메뉴 번호 : 2
===== 사각형 메뉴 =====
1. 사각형 둘레
2. 사각형 넓이
9. 예인으로
메뉴 번호 : 2
x 좌표 : 2
y 좌표 : 4
높이 : 1
너비 : 3
위치 : (2, 4), 너비 : 3, 높이 : 1 / 넓이 : 3
```