

클래스와 객체 (멤버 접근)

백석대학교 강윤희



클래스의 구성 요소와 멤버 접근

■ 클래스 내부에서 멤버 접근


this.필드
this.메서드

혹은

필드
메서드

● 예제

```
class Circle {  
    double radius;  
  
    double findArea() {  
        return 3.14 * radius * radius;  
    }  
    void show(double x, double y) {  
        System.out.printf("반지름 = %.1f, 넓이 = %.1f\n", x, y);  
    }  
}
```

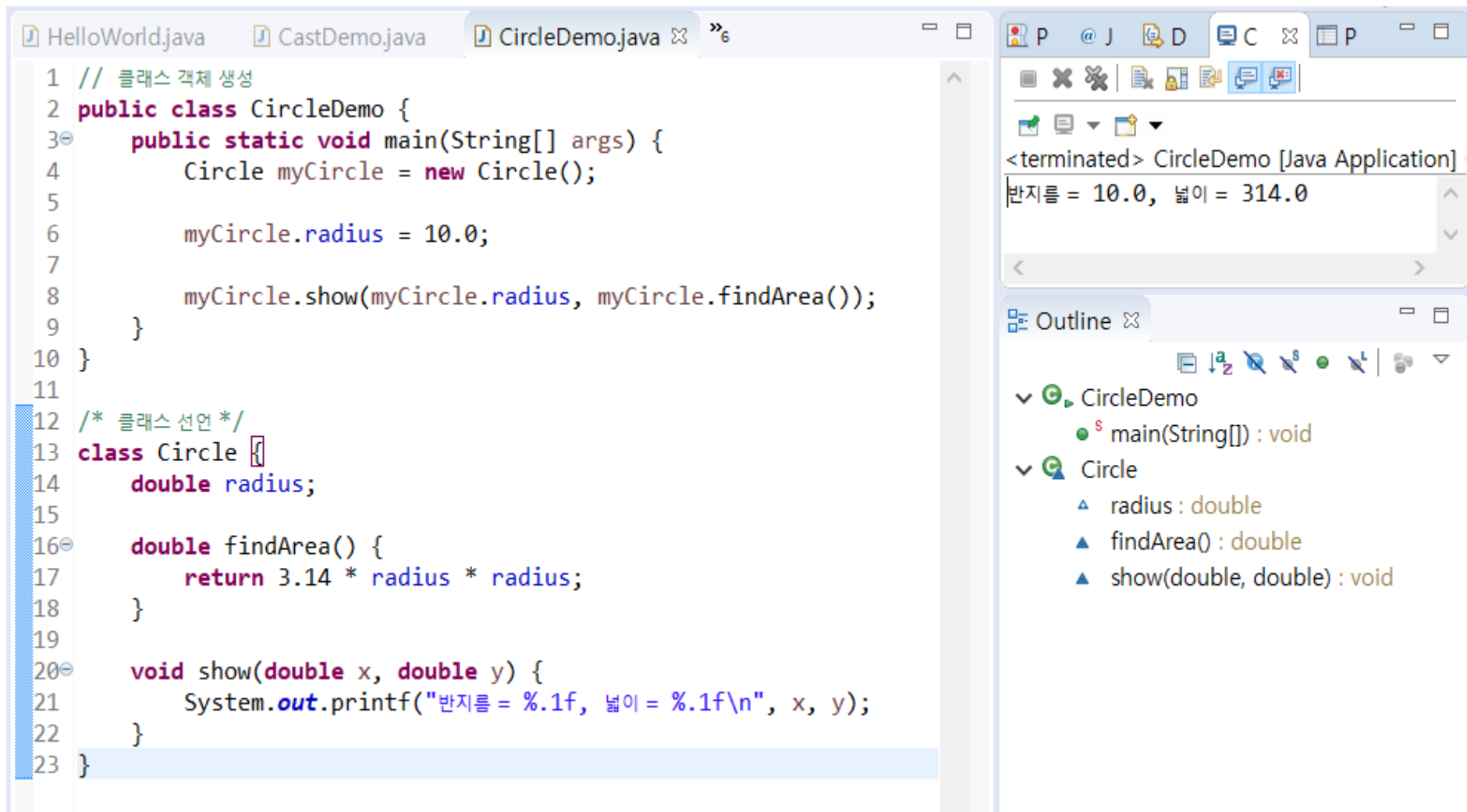


반지름 = 10.0, 넓이 = 314.0

(실습) CircleDemo.java를 사용하여 반지름이 5인 원(Circle)을 생성하고 반지름과 넓이를 출력하시오



클래스의 구성 요소와 멤버 접근



The screenshot displays an IDE with two tabs: 'HelloWorld.java' and 'CastDemo.java'. The active tab is 'CircleDemo.java', which contains the following Java code:

```
1 // 클래스 객체 생성
2 public class CircleDemo {
3     public static void main(String[] args) {
4         Circle myCircle = new Circle();
5
6         myCircle.radius = 10.0;
7
8         myCircle.show(myCircle.radius, myCircle.findArea());
9     }
10 }
11
12 /* 클래스 선언 */
13 class Circle {
14     double radius;
15
16     double findArea() {
17         return 3.14 * radius * radius;
18     }
19
20     void show(double x, double y) {
21         System.out.printf("반지름 = %.1f, 넓이 = %.1f\n", x, y);
22     }
23 }
```

On the right side of the IDE, there is a console window titled '<terminated> CircleDemo [Java Application]'. It shows the output of the program:

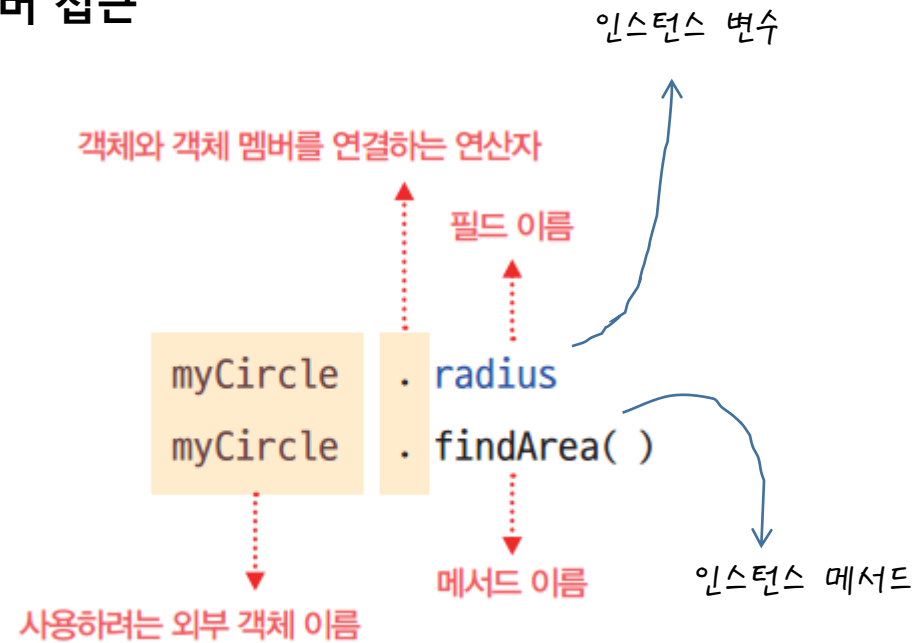
```
반지름 = 10.0, 넓이 = 314.0
```

Below the console window is an 'Outline' view showing the class structure:

- CircleDemo
 - main(String[]) : void
- Circle
 - radius : double
 - findArea() : double
 - show(double, double) : void

클래스의 구성 요소와 멤버 접근

■ 클래스 외부에서 멤버 접근



접근자와 설정자

■ 필요성

- 클래스 내부에 캡슐화된 멤버를 외부에서 사용할 필요

■ 접근자와 설정자

- private으로 지정된 필드에 값을 반환하는 접근자와 값을 변경하는 설정자는 공개된 메서드
- 일반적으로 접근자는 get, 설정자는 set으로 시작하는 이름을 사용
- 필드 이름을 외부와 차단해서 독립시키기 때문에 필드 이름 변경이나 데이터 검증도 가능



접근자(get)와 설정자(set)

■ 캡슐화 private

- 클래스 멤버를 은닉
- 외부 클래스 접근 제어

```

HelloWorld.java  CastDemo.java  Test.java  *CircleDemo...  »s  □
1  // 클래스 객체 생성
2  public class CircleDemo {
3      public static void main(String[] args) {
4          Circle myCircle = new Circle();
5
6          myCircle.radius = 10.0;
7
8          myCircle.show(myCircle.radius, myCircle.findArea());
9      }
10 }
11
12 /* 클래스 선언 */
13 class Circle {
14     private Double radius;
15
16     double findArea() {
17         return 3.14 * radius * radius;
18     }
19
20     void show(double x, double y) {
21         System.out.printf("반지름 = %.1f, 넓이 = %.1f\n", x, y);
22     }
23 }
```



접근자(get)와 설정자(set)

```
1 public class CircleDemo {
2     public static void main(String[] args) {
3         Circle myCircle = new Circle();
4
5         myCircle.setRadius(10.0);
6
7         myCircle.show(myCircle.getRadius(), myCircle.findArea());
8     }
9 }
10
11 class Circle {
12     private double radius;
13
14     public double getRadius() {
15         return radius;
16     }
17
18     public void setRadius(double r) {
19         this.radius = r;
20     }
21
22     double findArea() {
23         return 3.14 * radius * radius;
24     }
25
26     void show(double x, double y) {
27         System.out.printf("반지름 = %f, 넓이 = %f\n", x,y);
28     }
29 }
```

<terminated> CircleDemo [Java Applet]

반지름 = 10.000000, 넓이 = 314.000000

Outline

CircleDemo

main(String[]) : void

Circle

radius : double

getRadius() : double

setRadius(double) : void

findArea() : double

show(double, double) : void

