Concepts of Classes and Objects

Prepared by

Dr. Rajesh Kumar Ojha
Asst. Prof., CSE, Silicon University

Class?

- A *class* is the blueprint from which individual objects are created.
- It is a logical entity.

Class by Example

- In the real world, you'll often find many individual objects all of the same kind.
- There may be thousands of other bicycles in existence, all of the same make and model.
- Each bicycle was built from the same set of blueprints and therefore contains the same components.
- In object-oriented terms, we say that your bicycle is an *instance* of the *class of objects* known as bicycles

Class by Example

```
class Bicycle
{
int cadence = 0;
int speed = 0;
 int gear = 1;
void changeCadence(int newValue) {
cadence = newValue;
}
void changeGear(int newValue) {
 gear = newValue;
}
void speedUp(int increment) {
speed = speed + increment;
 }
void applyBrakes(int decrement) {
speed = speed - decrement;
}
void printStates() {
System.out.println("cadence:" +
cadence + " speed:" + speed + "
gear:" + gear);
}
}
```

```
class Box {
   double width;
   double height;
   double depth;
}
```

```
Box mybox = new Box(); // create a Box object called mybox
```

```
class Box {
  double width;
  double height;
  double depth;
}

// This class declares an object of type Box.
class BoxDemo {
  public static void main(String args[]) {
    Box mybox = new Box();
    double vol;

    // assign values to mybox's instance variables
    mybox.width = 10;
```

```
    mybox.height = 20;
    mybox.depth = 15;

    // compute volume of box
    vol = mybox.width * mybox.height * mybox.depth;

    System.out.println("Volume is " + vol);
  }
}
```

```
Volume is 3000.0
```

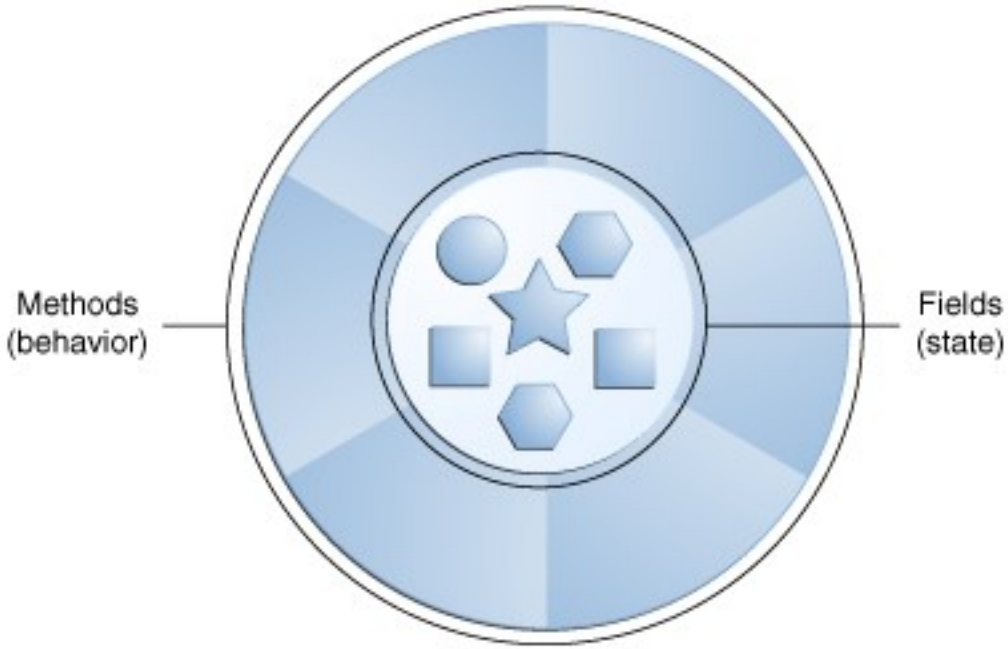
Object?

- Objects are key to understanding *object-oriented* technology. Look around right now and you'll find many examples of real-world objects: your dog, your desk, your television set, your bicycle.
- Real-world objects share two characteristics: They all have *state* and *behavior*.
- Identifying the state and behavior for real-world objects is a great way to begin thinking in terms of object-oriented programming.
- Dogs have state (name, color, breed, hungry) and behavior (barking, fetching, wagging tail). Bicycles also have state (current gear, current pedal cadence, current speed) and behavior (changing gear, changing pedal cadence, applying brakes).

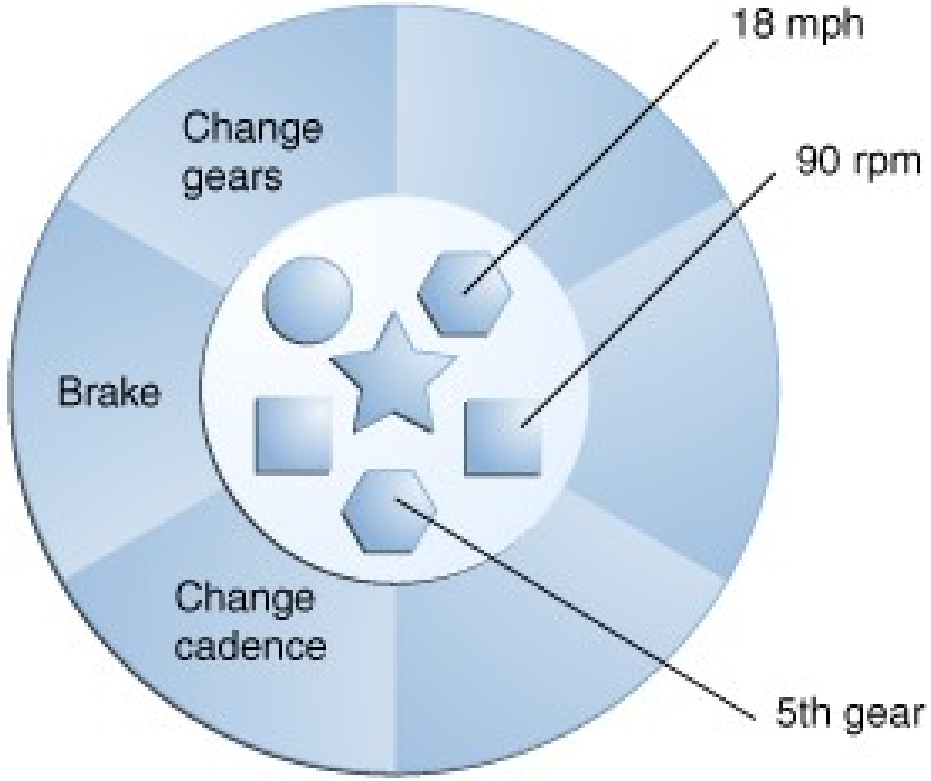
Object?

- Software objects are conceptually similar to real-world objects: they too consist of state and related behavior.
- An object stores its state in *fields* (variables in some programming languages) and exposes its behavior through *methods* (functions in some programming languages).
- Methods operate on an object's internal state and serve as the primary mechanism for object-to-object communication.

Object?

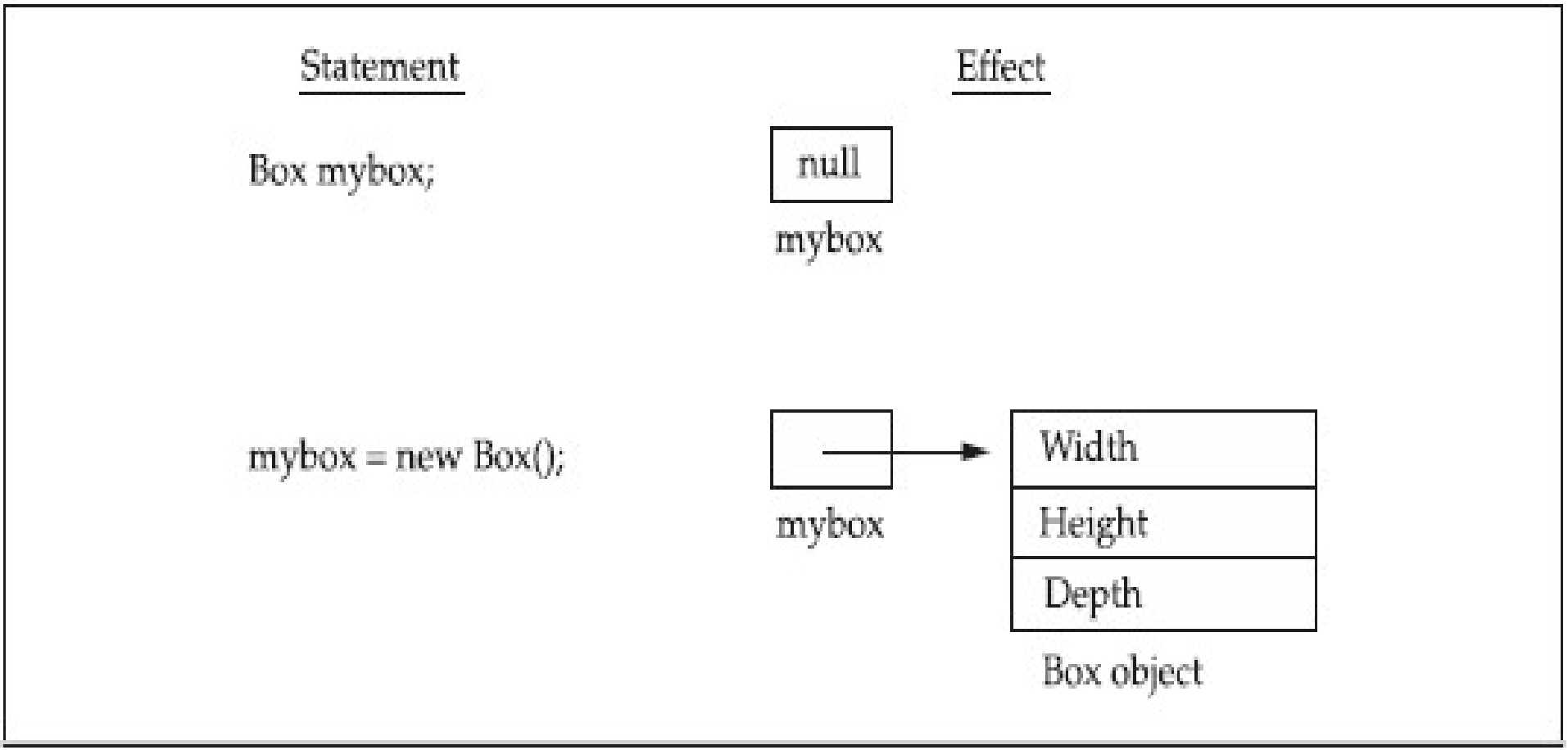


A software object.



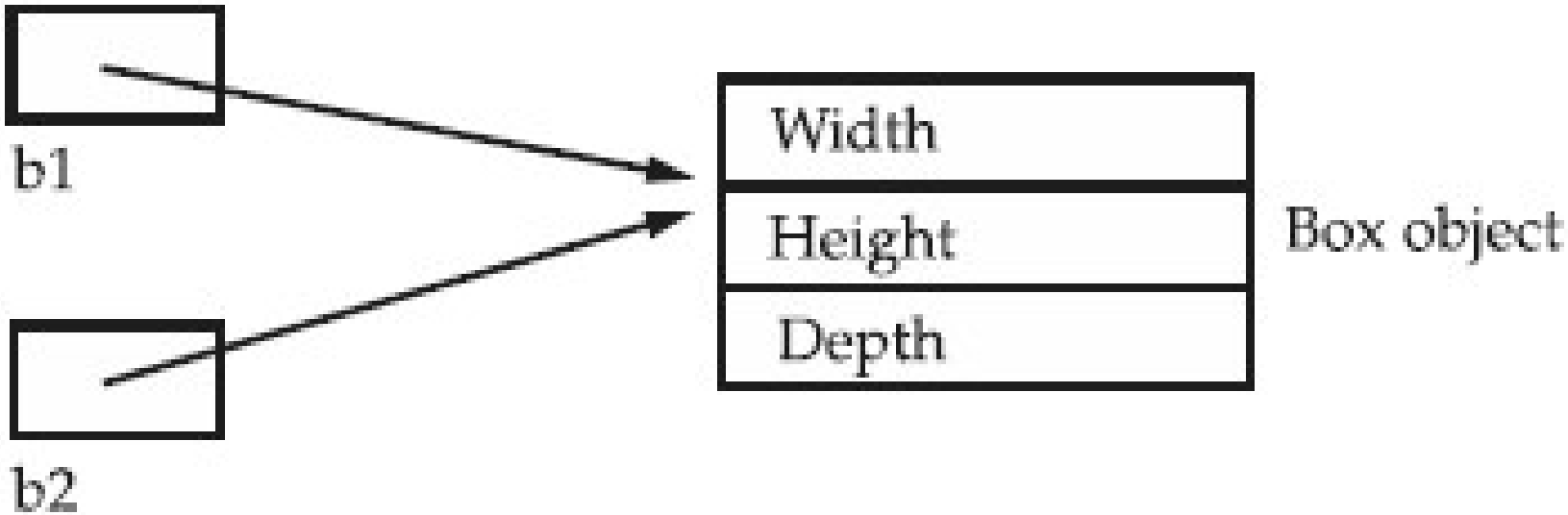
A bicycle modeled as a software object.

```
Box mybox; // declare reference to object
mybox = new Box(); // allocate a Box object
```



Object Reference Variables

```
Box b1 = new Box();
Box b2 = b1;
```



Declaring/Creating Objects in a Single Step

```
ClassName objectRefVar = new ClassName();
```

Example:

Assign object reference

Create an object

```
Circle myCircle = new Circle();
```

Accessing Objects

- Referencing the object's data:

  `objectRefVar.data`

  e.g., `myCircle.radius`

- Invoking the object's method:

  `objectRefVar.methodName(arguments)`

  e.g., `myCircle.getArea()`

A Simple Circle Class

- Objective: Demonstrate creating objects, accessing data, and using methods.

TestCircle1

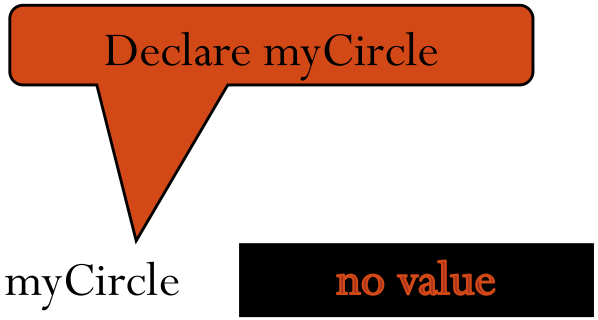
Run

Trace Code

```
Circle myCircle = new Circle(5.0);

SCircle yourCircle = new Circle();

yourCircle.radius = 100;
```

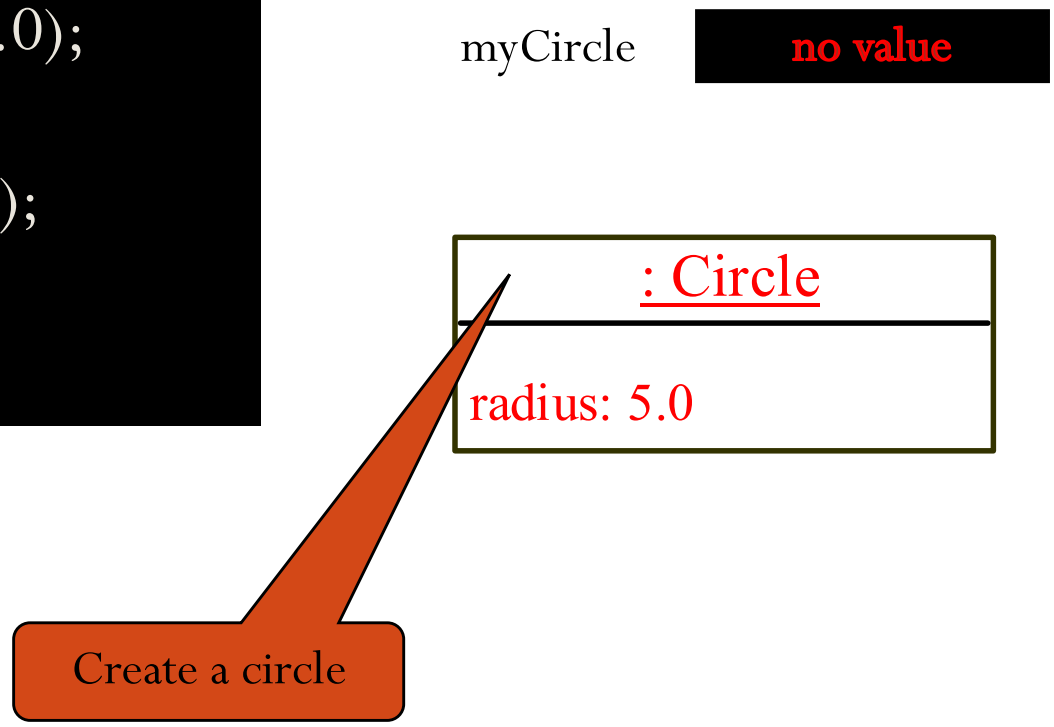


Trace Code, cont.

```
Circle myCircle = new Circle(5.0);

Circle yourCircle = new Circle();

yourCircle.radius = 100;
```



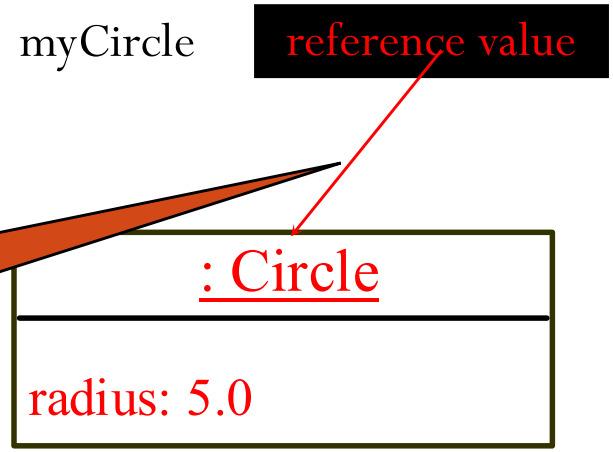
Trace Code, cont.

```
Circle myCircle = new Circle(5.0);

Circle yourCircle = new Circle();

yourCircle.radius = 100;
```

Assign object reference to myCircle

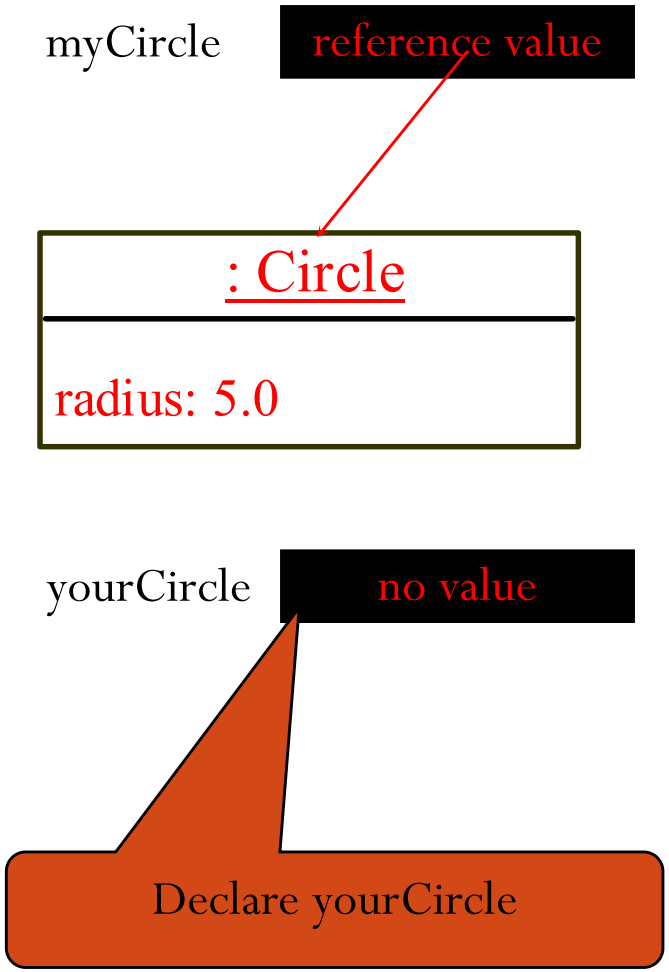


Trace Code, cont.

```
Circle myCircle = new Circle(5.0);

Circle yourCircle = new Circle();

yourCircle.radius = 100;
```

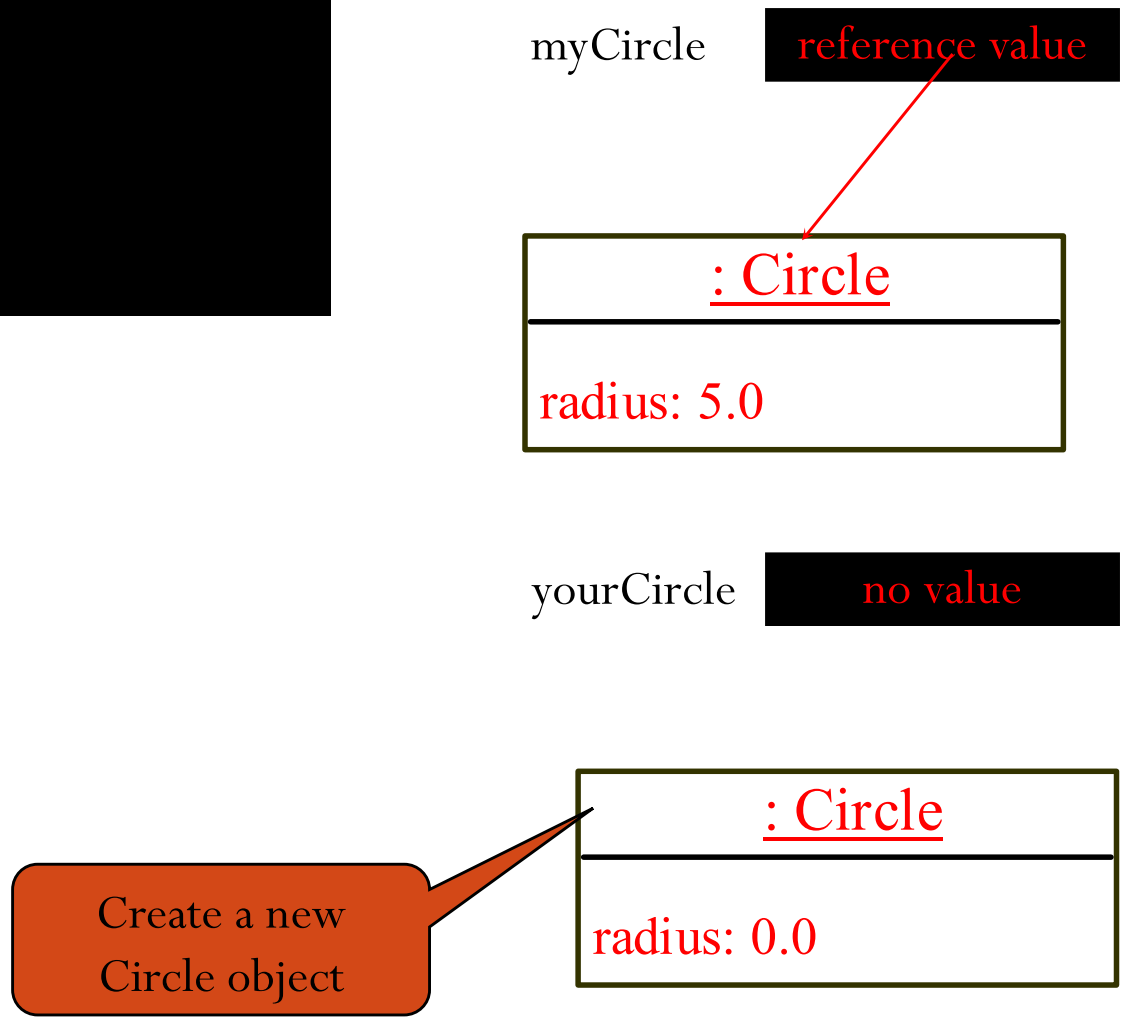


Trace Code, cont.

```
Circle myCircle = new Circle(5.0);

Circle yourCircle = new Circle();

yourCircle.radius = 100;
```

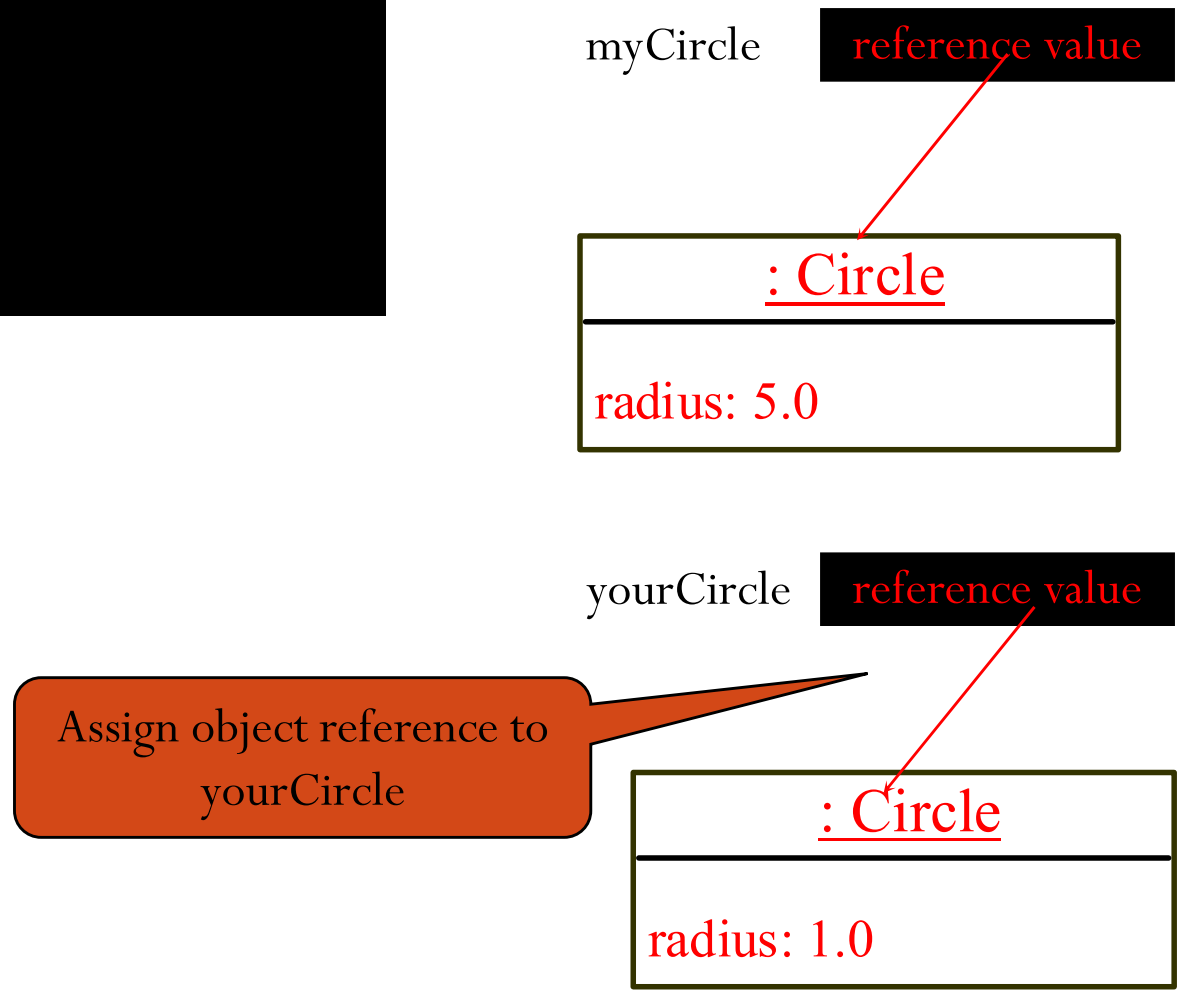


Trace Code, cont.

```
Circle myCircle = new Circle(5.0);

Circle yourCircle = new Circle();

yourCircle.radius = 100;
```

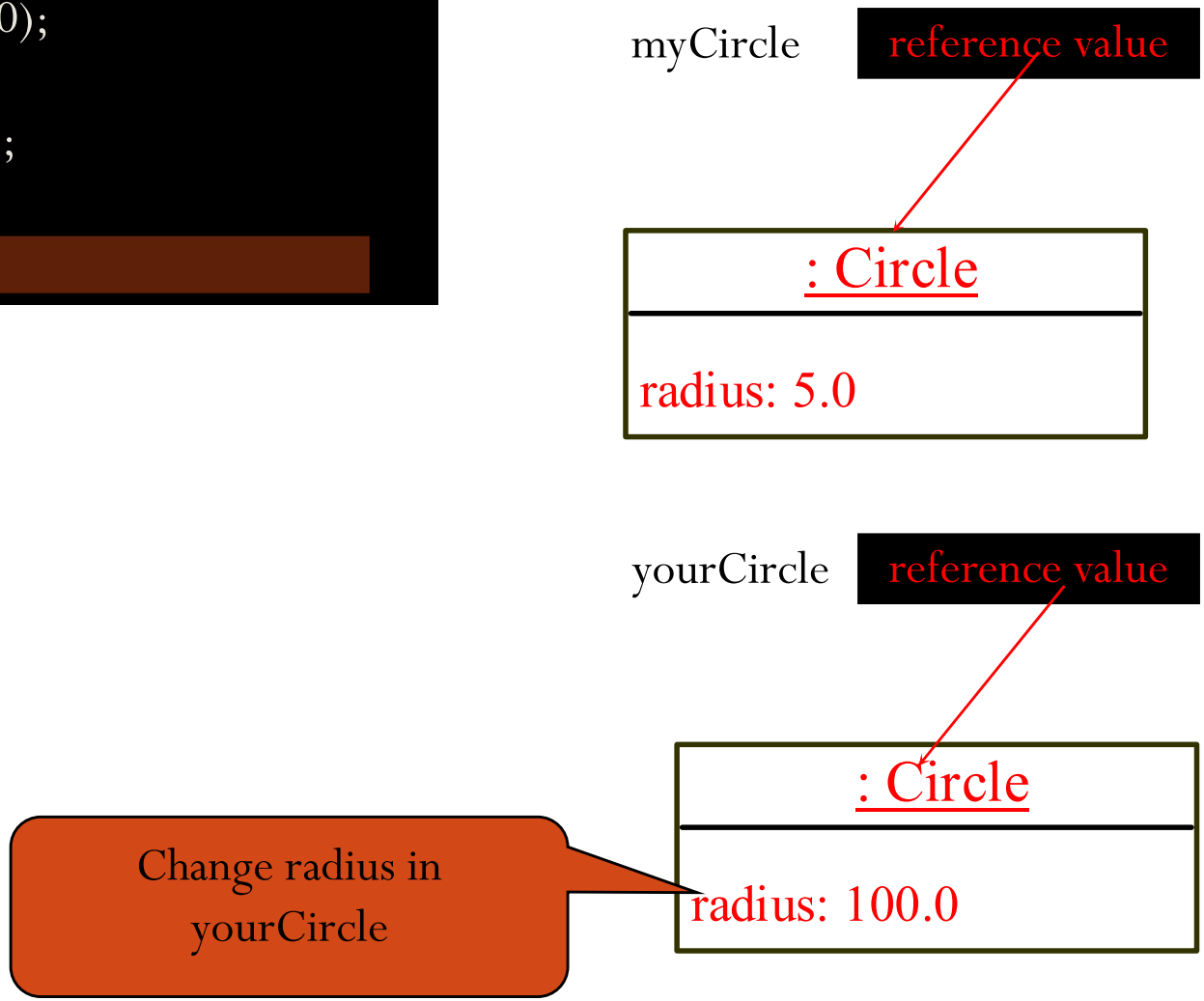


Trace Code, cont.

```
Circle myCircle = new Circle(5.0);

Circle yourCircle = new Circle();

yourCircle.radius = 100;
```



Thank you