Classes

- Classes are templates that define objects of the same type.
- A Java class uses:
  - variables to define data fields and
  - methods to define behaviors
- A class provides a special type of methods called constructors which are invoked to construct objects from the class

Classes

```
class Circle {
/** The radius of this circle */
private double radius = 1.0;

/** Construct a circle object */
public Circle() {
}
/** Construct a circle object */
public Circle(double newRadius)
{
 radius = newRadius;
}
/** Return the area of this circle
*/
public double getArea() {
 return radius * radius * 3.14159;
```

Data field

Constructors

Method

Classes

```
public class TestCircle {
public static void main(String[] args) {
 Circle c1 = new Circle();
Circle c2 = new Circle(5.0);
 System.out.println( c1.getArea() );
 System.out.println( c2.getArea() );
 System.out.println( c1.radius );
 System.out.println( c2.radius );
```

Constructors

- Constructors must have the same name as the class itself.
- Constructors do not have a return type—not even void.
- Constructors are invoked using the new operator when an object is created – they initialize objects to reference

  variables:

  ClassName o = new ClassName();

- Example:

  Circle myCircle = new Circle(5.0);

- A class may be declared without constructors: a no-arg default constructor with an empty body is implicitly declared in the class

Accessing Objects

- Referencing the object's data:
  objectRefVar.data
  - Example: myCircle.radius
- Invoking the object's method:
  objectRefVar.methodName(arguments)
- Example: myCircle.getArea()

Using classes

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

SCircle yourCircle = new Circle();

yourCircle.radius = 100;
```

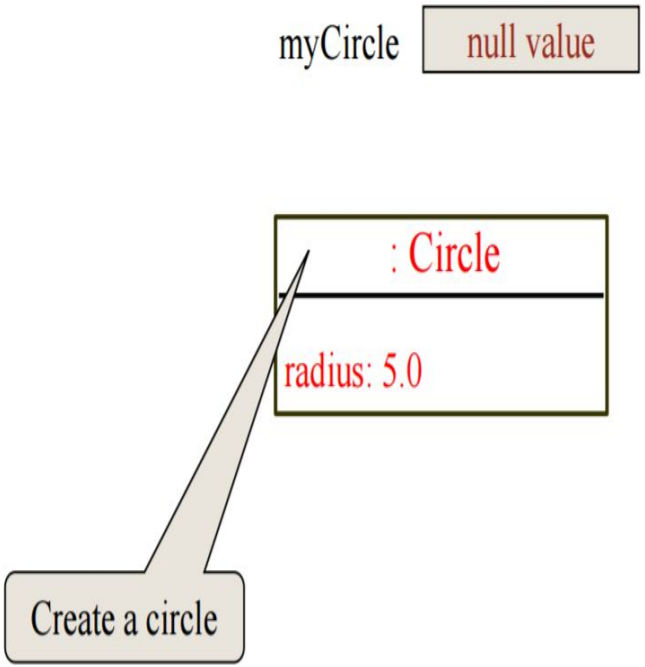
Declare myCircle

myCircle

null value

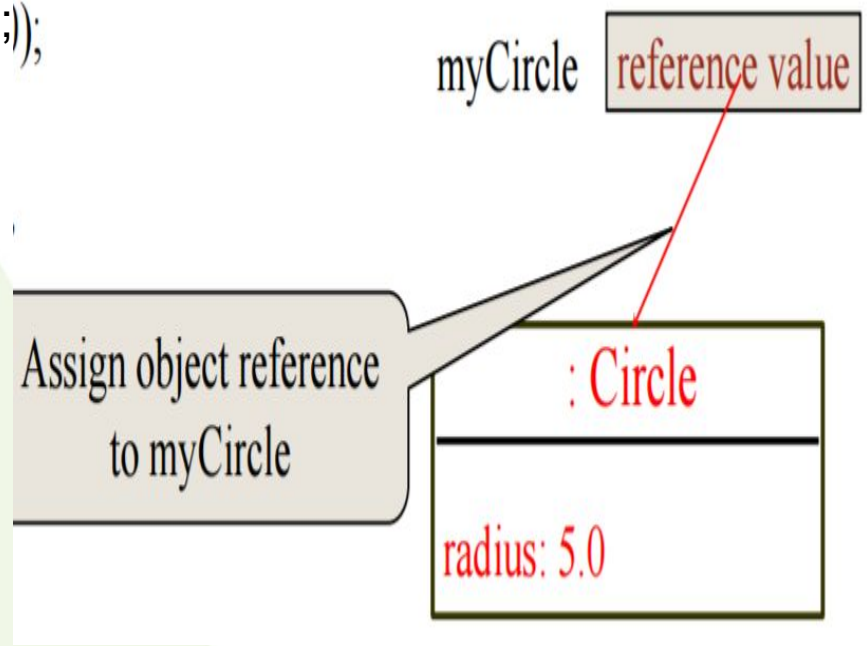
Using classes

```
Circle myCircle = new Circle(5.0);
Circle yourCircle = new Circle();
yourCircle.radius = 100;
```



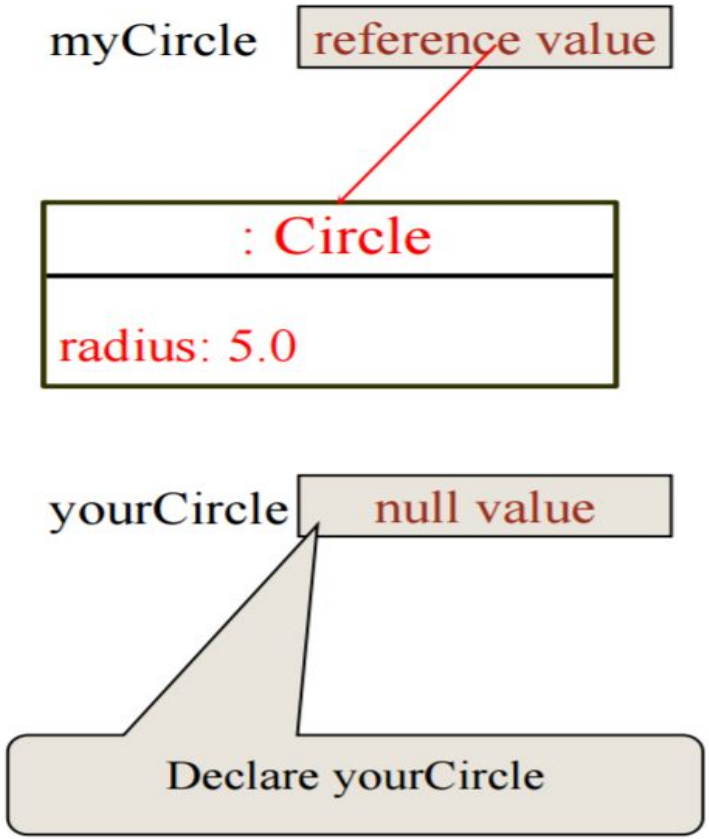
Using classes

```
Circle myCircle = new Circle(5.0));
Circle yourCircle = new Circle();
yourCircle.radius = 100;
```



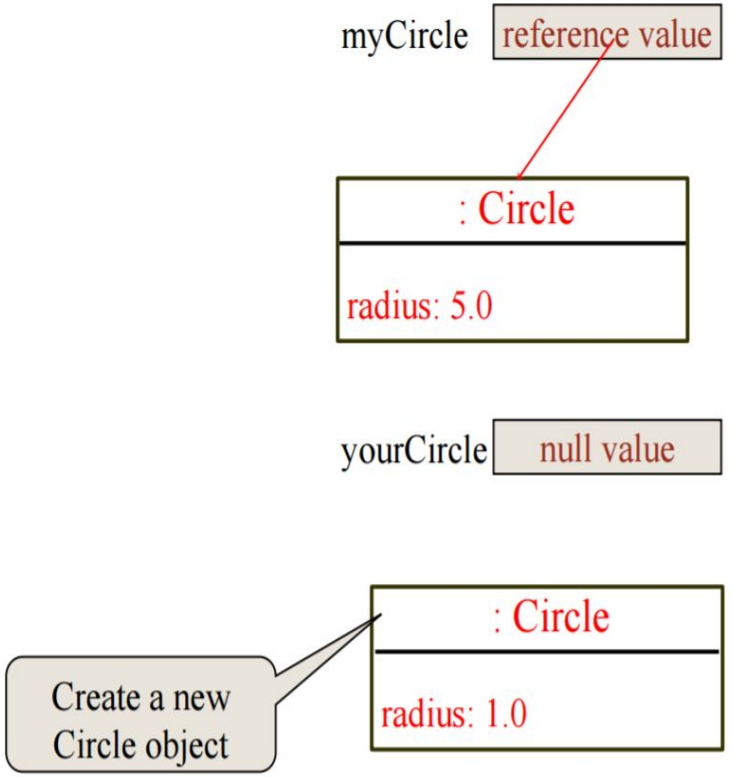
Using classes

```
Circle myCircle = new Circle(5.0);
Circle yourCircle = new Circle();
yourCircle.radius = 100;
```



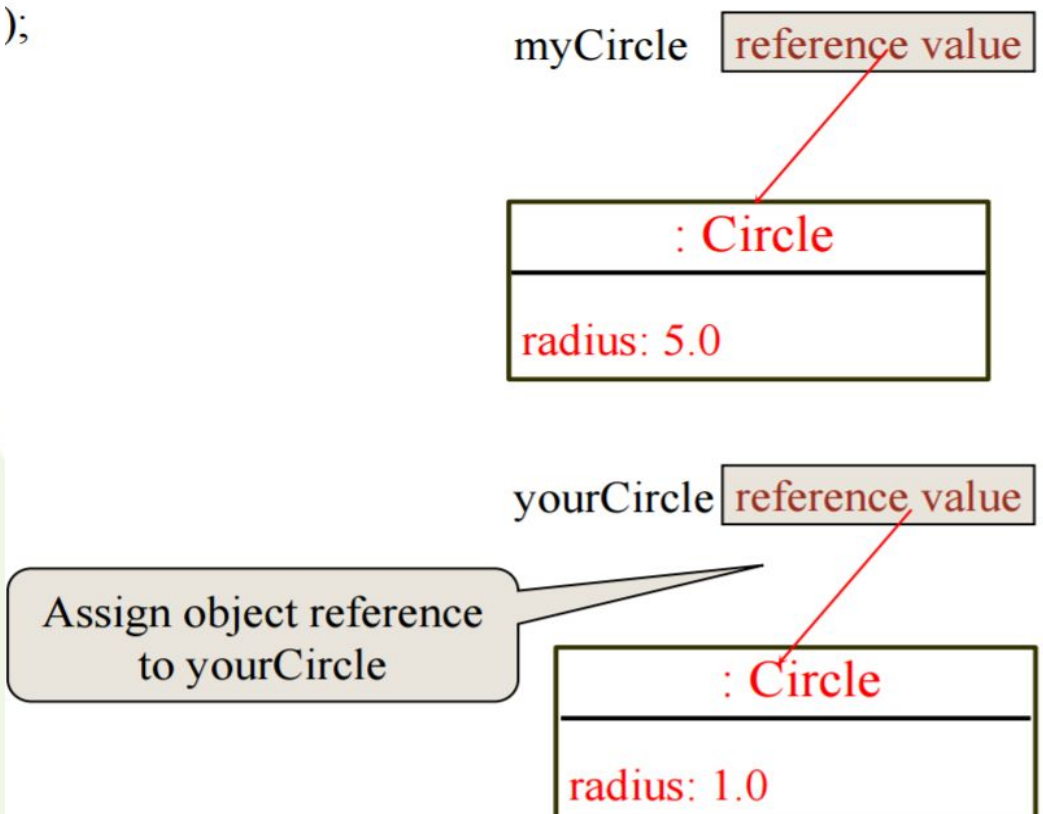
Using classes

```
Circle myCircle = new Circle(5.0);
Circle yourCircle = new Circle();
yourCircle.radius = 100;
```



Using classes

```
Circle myCircle = new Circle(5.0);
Circle yourCircle = new Circle();
yourCircle.radius = 100;
```



Using classes

```
Circle myCircle = new Circle(5.0);
Circle yourCircle = new Circle();
yourCircle.radius = 100;
```