Constructor Program

Reference: Liang, Y., 2015. Intro To Java Programming. 10th ed. Pearson Education Limited.

Create a class named 'Student' with String variable 'name' and integer variable 'roll_no'. Assign the value of roll_no as '2' and that of name as "John" by creating an object of the class Student.

```
class Student
{
String name;
  int id;
}
class Test{
public static void main(String[] args)
{
Student s = new Student();
s.name = "John";
s.id = 12;
System.out.println("Name is "+ s.name+ " and roll number is "+ s.id);
  }
}
```

Name is John and roll number is 2

```
class Student{
 int id;
 String name;
}
class Test{
 public static void main(String[] args){
  //Creating objects
  Student s1=new Student();
  Student s2=new Student();
  //Initializing objects
  s1.id=10;
  s1.name="Mike";
  s2.id=11;
  s2.name="Luke";
  //Printing data
  System.out.println("Name is "+ s1.name+ " and roll number is "+ s1.id);
  System.out.println("Name is "+ s2.name+ " and roll number is "+ s2.id);
 }
}
```

```
Name is Mike and roll number is 11
Name is Luke and roll number is 12
```

```
class Employee{
   int id;
   String name;
   float salary;
   void insert(int i, String n, float s)
{     id=i;
       name=n;
       salary=s;   }
   void display(){System.out.println(id+" "+name+" "+salary);}
}
public class TestEmployee {
public static void main(String[] args) {
   Employee e1=new Employee();
   Employee e2=new Employee();
   Employee e3=new Employee();
   e1.insert(101,"ajeet",45000);
   e2.insert(102,"irfan",25000);
   e3.insert(103,"nakul",55000);
   e1.display();
   e2.display();
   e3.display();
}
}
```

```
// Define the circle class with one variable and two methods

class Circle
{
  double radius=10;

  double getArea()
   {
     return radius * radius * Math.PI;
   }
  double getPerimeter()
   {
     return 2 * radius * Math.PI;
   }

 }
```

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()

Calling Circle class via object:

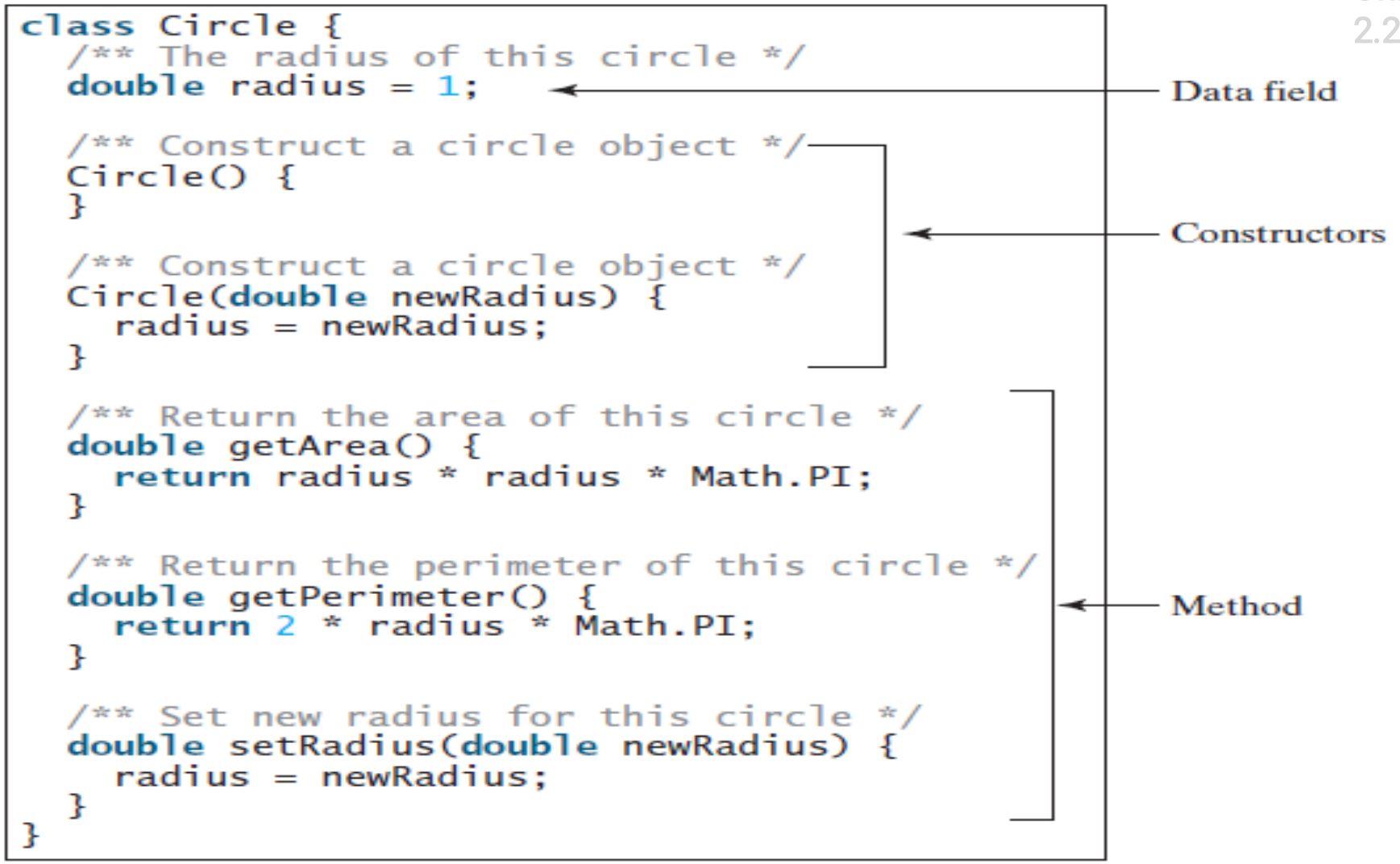

```
public class mainCircle
{

     public static void main(String[] args)
     {
          // Circle with default radius
          Circle c = new Circle();
          double area = c.getArea();
          double perimeter = c.getPerimeter();

          System.out.println("Circle Area is " +  area);
          System.out.println("Circle Perimeter is "+ perimeter);

     }
}
```

Implementing the circle class:



Implementing the circle class:


```
class Circle {
  /** The radius of this circle */
  double radius = 1;

  /** Construct a circle object */
  Circle() {
  }

  /** Construct a circle object */
  Circle(double newRadius) {
    radius = newRadius;
  }

  /** Return the area of this circle */
  double getArea() {
    return radius * radius * Math.PI;
  }

  /** Return the perimeter of this circle */
  double getPerimeter() {
    return 2 * radius * Math.PI;
  }

  /** Set new radius for this circle */
  double setRadius(double newRadius) {
    radius = newRadius;
  }
}
```

Data field

Constructors

Method

What's missing?

Implementing the circle class:


```
class Circle {
  /** The radius of this circle */
  double radius = 1;

  /** Construct a circle object */
  Circle() {
  }

  /** Construct a circle object */
  Circle(double newRadius) {
    radius = newRadius;
  }

  /** Return the area of this circle */
  double getArea() {
    return radius * radius * Math.PI;
  }

  /** Return the perimeter of this circle */
  double getPerimeter() {
    return 2 * radius * Math.PI;
  }

  /** Set new radius for this circle */
  double setRadius(double newRadius) {
    radius = newRadius;
  }
}
```

Data field

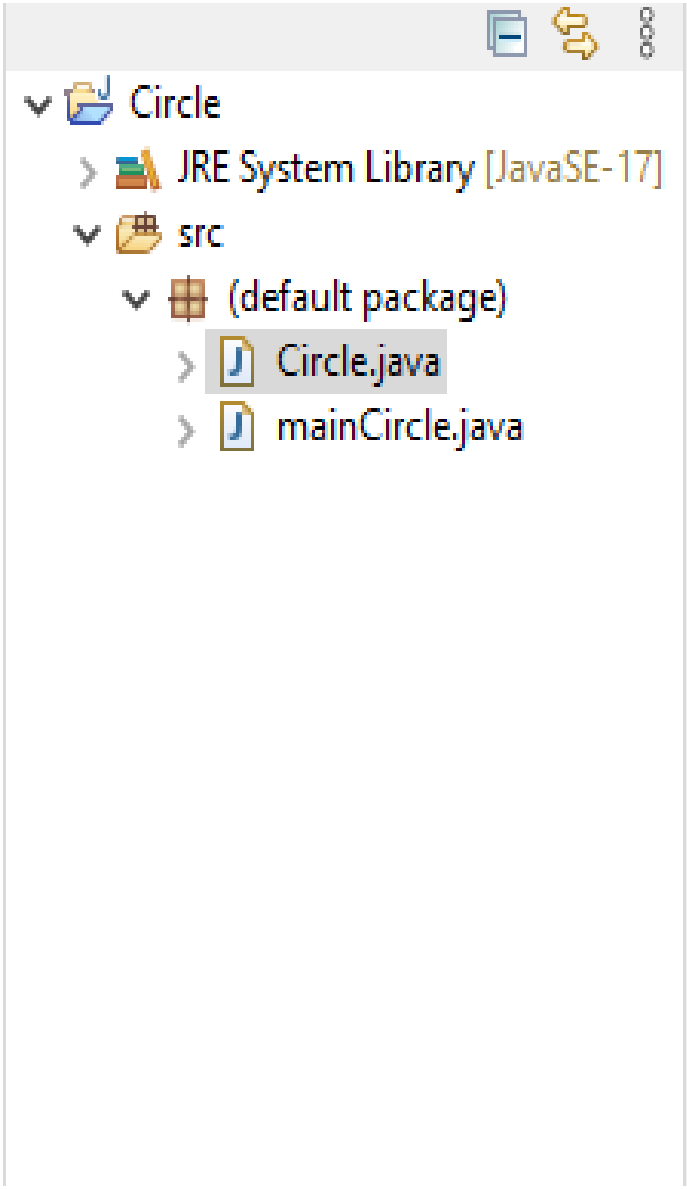
Constructors

Method

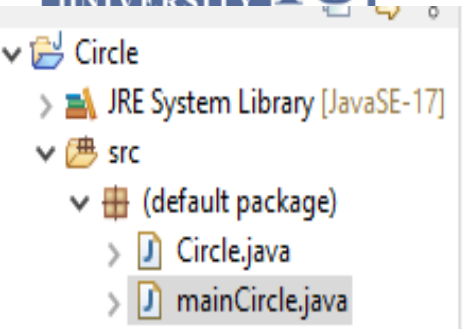
It does not have a **main method.**

Constructor vs. Method

Constructor	Method
Constructor is used to initialize the state of an object.	Method is used to expose behaviour of an object.
Constructor must not have return type.	Method must have return type.
Constructor is invoked implicitly.	Method is invoked explicitly.
The java compiler provides a default constructor if you don't have any constructor.	Method is not provided by compiler in any case.
Constructor name must be same as the class name.	Method name may or may not be same as class name.




```
1
2  public class Circle {
3          double radius;
4          // Non-parameterized Constructor
5          Circle() {
6              radius = 1;
7          }
8          // Parameterized Constructor
9          Circle(double newRadius) {
10             radius = newRadius;
11         }
12         // Defining Methods
13         double getArea()          {
14             return radius * radius * Math.PI;
15         }
16         double getPerimeter()         {
17             return 2 * radius * Math.PI;
18         }
19         void setRadius(double newRadius)     {
20             radius = newRadius;
21         }
22     }
```




```java
public class mainCircle {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        // Circle with default radius 10
        Circle c1 = new Circle();
        double area = c1.getArea();
        double perimeter = c1.getPerimeter();
        System.out.println("The Area of the circle of radius (" + c1.radius + ") is " +  area);
        System.out.println("The Perimeter of the circle of radius " + c1.radius + " is " +  perimeter);
        // Circle with radius 20
        Circle c2 = new Circle(20);
        area = c2.getArea();
        perimeter = c2.getPerimeter();
        System.out.println("The Area of the circle of radius (" + c2.radius + ") is " +  area);
        System.out.println("The Perimeter of the circle of radius " + c2.radius + " is " +  perimeter);
        // Circle with radius 30
        c2.radius = 30;
        area = c2.getArea();
        perimeter = c2.getPerimeter();
        System.out.println("The Area of the circle of radius (" + c2.radius + ") is " +  area);
        System.out.println("The Perimeter of the circle of radius " + c2.radius + " is " +  perimeter);
    }
}
```

Problems @ Javadoc Declaration Console

<terminated> mainCircle [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (06-Mar-2022, 12:04:42 pm – 12:04:48 pm)

```
The Area of the circle of radius (1.0) is 3.141592653589793
The Perimeter of the circle of radius 1.0 is 6.283185307179586
The Area of the circle of radius (20.0) is 1256.6370614359173
The Perimeter of the circle of radius 20.0 is 125.66370614359172
The Area of the circle of radius (30.0) is 2827.4333882308138
The Perimeter of the circle of radius 30.0 is 188.49555921538757
```


THANK YOU