

Java Classes

CSCI 2300

In your assigned reading you learned about

- Java Classes and Objects
- Java Class Attributes
- Java Class Methods
- Java Constructors
- Java Modifiers

•Any Questions?

Class Attributes

```
public class Point2D
{
    public int x = 0;
    public int y = 0;
}
```

Point2D.java

What are the (x, y) values of p1 and p2?

```
public class CreatePoints
{
    public static void main(String []args)
    {
        Point2D p1 = new Point2D();
        Point2D p2 = new Point2D();
        p2.x = 10;
    }
}
```

CreatePoints.java

- A. p1: (0, 0), p2: (0, 0)
- B. p1: (10, 0), p2: (0, 0)
- C. p1: (0, 0), p2: (10, 0)
- D. None of the above, the code will not compile.

Point2D.java

```
public class Point2D
{
    private int x = 0;
    private int y = 0;
}
```

What are the (x, y) values of p1 and p2?

CreatePoints.java

```
public class CreatePoints
{
    public static void main(String []args)
    {
        Point2D p1 = new Point2D();
        Point2D p2 = new Point2D();
        p2.x = 10;
    }
}
```

A. p1: (0, 0), p2: (0, 0)

B. p1: (10, 0), p2: (0, 0)

C. p1: (0, 0), p2: (10, 0)

D. None of the above, the code will not compile.

Class Methods

```
public class Point2D
{
    public int x = 0;
    public int y = 0;
    public void setX(int v){x = v;}
    public void setY(int v){y = v;}
}
```

Point2D.java

How can we set the (x, y) value of object p to be (10, 10)?

```
public class CreatePoints
{
    public static void main(String []args)
    {
        Point2D p = new Point2D();
    }
}
```

CreatePoints.java

- A. p.setX(10); p.setY(10);
- B. p.x = 10; p.y = 10;
- C. A & B
- D. We need to modify Point2D class definition.

Constructors

- Special methods used to initialize objects
- If a custom constructor is not defined, Java provides a default constructor.
- Example: `Point2D p = new Point2D();`
- Can define our own constructors:

```
public Point2D(int x, int y)
{
    this.x = x;
    this.y = y;
}
```

this: keyword used to refer to the calling object.

```

public class Point2D
{
    public int x = 0;
    public int y = 0;
    public Point2D(int x, int y)
    {
        this.x = x;
        this.y = y;
    }
}

```

Point2D.java

```

public class CreatePoints
{
    public static void main(String []args)
    {
        Point2D p = new Point2D(10, 10);
    }
}

```

CreatePoints.java

- If you provide a custom constructor, the default (no argument) constructor is no longer available automatically
- Define your own default constructor, if you need it

```

public class Point2D
{
    public int x;
    public int y;
    public Point2D(int x, int y){...}
    public Point2D()
    {
        this.x = 0;
        this.y = 0;
    }
}

```

Point2D.java

Constructors

When we create (instantiate) a Point2D object as shown below, which constructor are we using?

```
Point2D p = new Point2D(1, 2);
```

```
1 public class Point2D
2 {
3     public int x;
4     public int y;
5     public Point2D(int x, int y){...}
6     public Point2D()
7     {
8         this.x = 0;
9         this.y = 0;
10    }
11}
```

Point2D.java

- A. Constructor on line 5
- B. Constructor on lines 6-10
- C. We cannot instantiate a Point2D object this way
- D. This call is ambiguous: Java will not “know” which constructor to use

When we create (instantiate) a Point2D object as shown below, which constructor are we using?

```
Point2D p = new Point2D();
```

```
1 public class Point2D
2 {
3     public int x;
4     public int y;
5     public Point2D(int x, int y){...}
6     public Point2D()
7     {
8         this.x = 0;
9         this.y = 0;
10    }
11}
```

Point2D.java

- A. Constructor on line 5
- B. Constructor on lines 6-10
- C. We cannot instantiate a Point2D object this way
- D. This call is ambiguous: Java will not “know” which constructor to use

When we create (instantiate) a Point2D object as shown below, which constructor are we using?

```
Point2D p = new Point2D(1);
```

```
1 public class Point2D
2 {
3     public int x;
4     public int y;
5     public Point2D(int x, int y){...}
6     public Point2D()
7     {
8         this.x = 0;
9         this.y = 0;
10    }
11}
```

Point2D.java

- A. Constructor on line 5
- B. Constructor on lines 6-10
- C. We cannot instantiate a Point2D object this way
- D. This call is ambiguous: Java will not “know” which constructor to use

Using classes to define other classes

- Class – abstract data type (ADT)
- We can use a class to define attributes of other classes

```
public class Line2D
{
    private Point2D start;
    private Point2D end;
    public Line2D(Point2D s, Point2D e)
    {
        start = new Point2D(s.getX(), s.getY());
        end = new Point2D(e.getX(), e.getY());
    }
}
```

Line2D.java

Useful methods to include:

- Default constructor
- `public String toString(){ ...}` method
 - Allows objects to be used as “Strings”

```
public class Point2D
{
    public int x = 0;
    public int y = 0;
    public String toString()
    {
        return "(" + x + ", " + y + ")";
    }
}
```

Line2D example on course web site

Lab Assignment 3

- Create lab3 sub-directory in your csci2300 directory.
- Follow Line2D example to:
 - Design and implement a Circle2D class. Consider the following questions:
 - What are the attributes of a Circle in 2D?
 - What methods (behaviors) are appropriate for a Circle in 2D?
 - Write a "MakeCircle2D" class with the main method that:
 - Creates two circles in 2D
 - Outputs attribute values of the circles.
- Compile and run your code:

```
javac *.java  
java MakeCircle2D
```