Class Bumper


```
1  import java.awt.Color;
2  import java.awt.Graphics;
3
4  /**
5   * @author      Nathan Chen
6   * @author      Benjamin Tu
7   * @period      2
8   * @teacher     Coglianese
9   * @version     10-25-18
10  *
11  * Class Bumper is a rectangle that can check collision and jump
12  */
13 public class Bumper
14 {
15     private int myX;
16     private int myY;
17     private int myXWidth;
18     private int myYWidth;
19     private Color myColor;
20
21     /**
22      * Initializes bumper with default values
23      */
24     public Bumper(){
25         myX = 200;
26         myY = 200;
27         myXWidth = 25;
28         myYWidth = 50;
29         myColor = Color.BLUE;
30     }
31
32     /**
33      * Makes a bumper with the specified arguments
34      *
35      * @params      x       x position
36      * @params      y       y position
37      * @params      xWidth  Width of bumper
38      * @params      yWidth  Height of bumper
39      * @params      c       Color of bumper
40      */
41     public Bumper(int x, int y, int xWidth, int yWidth, Color c){
42         myX = x;
43         myY = y;
44         myXWidth = xWidth;
45         myYWidth = yWidth;
46         myColor = c;
47     }
48
49     //Accessor methods
```

```
50      public int getX(){return myX;}
51      public int getY(){return myY;}
52      public int getXWidth(){return myXWidth;}
53      public int getYWidth(){return myYWidth;}
54      public Color getColor(){return myColor;}
55
56      //Modifier methods
57      public void setX(int x){myX = x;}
58      public void setY(int y){myY = y;}
59      public void setXWidth(int xWidth){myXWidth = xWidth;}
60      public void setYWidth(int yWidth){myYWidth = yWidth;}
61      public void setColor(Color c){myColor = c;}
62
63      /**
64       * Bumper moves to a location within the right and bottom edges
65       *
66       * @params      rightEdge       Right edge that bumper cannot go beyo
    nd
67       * @params      bottomEdge      Bottom edge that bumper cannot go bey
    ond
68       */
69      public void jump(int rightEdge, int bottomEdge)
70      {
71          myX = ((int)(Math.random() * (rightEdge - myXWidth)));
72          myY = ((int)(Math.random() * (bottomEdge - myYWidth)));
73      }
74
75      /**
76       * Draws the bumper on a graphics class
77       *
78       * @params      myBuffer        Graphics class where bumper is drawn
79       */
80      public void draw(Graphics myBuffer)
81      {
82          myBuffer.setColor(getColor());
83          myBuffer.fillRect(getX(), getY(), getXWidth(), getYWidth());
84      }
85
86      /**
87       * Checks whether or not a Polkadot and any subclass is in the bumper
    , returns boolean
88       *
89       * @params      dot     Polkadot being checked
90       */
91      public boolean inBumper(Polkadot dot)
92      {
93          for (int x = getX(); x <= getX() + getXWidth(); x++)
94              for (int y = getY(); y <= getY() + getYWidth(); y++)
95                  if (distance(x, y, dot.getX(), dot.getY()) <= dot.getRadi
```

```
us())
                    return true;
        return false;
    }

    /**
     * Distance calculator between two objects, returns a double
     *
     * @params      x1      x of object 1
     * @params      y1      y of object 1
     * @params      x2      x of object 2
     * @params      y2      y of object 2
     */
    private double distance(double x1, double y1, double x2, double y2)
    {
        return Math.sqrt(Math.pow(x1 - x2, 2.0) + Math.pow(y1 - y2, 2.0))
;
    }
}
```