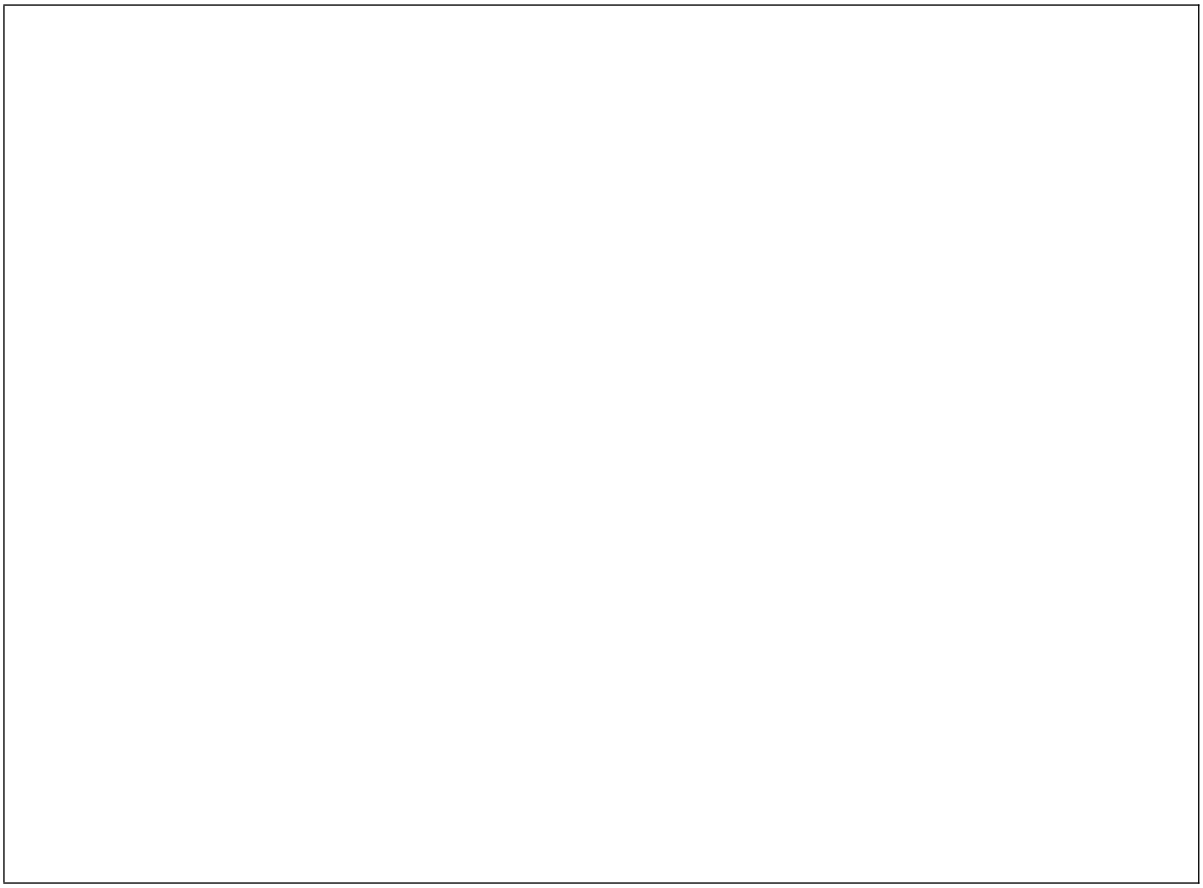
Worksheet: `BoundingBox` Class

©2026 Chris Nielsen – *www.nielsenedu.com*

Write `class BoundingBox` that has:

- Four private fields: `left`, `bottom`, `right`, and `top`, each of which stores a real number, and which together represents the bottom-left and top-right corners on a coordinate system.
- A constructor that takes four real parameters to set the four fields in the order given above.
- A getter for each of the fields (...but actually, just write one on your worksheet).
- An instance method named `width` that takes no parameters and returns the width of the `BoudingBox` object.
- An instance method named `height` that takes in no parameters and returns the height of the `BoudingBox` object.
- An instance method named `area` that returns the area of the `BoudingBox` object.
- An instance method named `encloses` that takes two parameters, `x` and `y`, which store real numbers and represents two coordinates in the coordinate system, and returns `true` if the coordinates are within the `BoudingBox` object, otherwise returns `false`.
- An instance method named `equals` that takes one `BoundingBox` object parameter and returns `true` if all four coordinates are the same in both objects, otherwise returns `false`.
- An instance method named toString that prints out the bounding box as two points on a coordinate system with one point being the bottom left and the other being the top right, for example: `(1.00, 1.00), (3.00, 5.00)`.

You do not need to perform error checking for any real value being “not a number” or “infinity”, however, the `BoundingBox` class must ensure `left` is more left than `right` on a horizontal number line and `bottom` is lower than the `top` on a vertical number line. Write the class in the space below and on the back of this page.



Worksheet: `BoundingBox` Class

©2026 Chris Nielsen – *www.nielsenedu.com*

```
1  public class BoundingBox {
2     private double left, bottom, right, top;
3     public BoundingBox(double left, double bottom,
4                        double right, double top) {
5        this.left = Math.min(left, right);
6        this.right = Math.max(left, right);
7        this.bottom = Math.min(bottom, top);
8        this.top = Math.max(bottom, top);
9     }
10    public double width() {
11       return right-left;
12    }
13    public double height() {
14       return top-bottom;
15    }
16    public double area() {
17       return this.width() * this.height();
18    }
19    public boolean encloses(double x, double y) {
20       return left <= x && x <= right &&
21            bottom <= y && y <= top;
22    }
23    public boolean equals(BoundingBox b) {
24       return this.left == b.left &&
25              this.right == b.right &&
26              this.bottom == b.bottom &&
27              this.top == b.top;
28    }
29    @Override
30    public String toString() {
31       return String.format("(%.2f, %.2f), (%.2f, %.2f)",
32                            left, bottom, right, top);
33    }
34    public double getLeft() {
35       return left;
36    }
37    public double getRight() {
38       return right;
39    }
40    public double getBottom() {
41       return bottom;
42    }
43    public double getTop() {
44       return top;
45    }
46 }
```