Morgan Visnesky

Chapter 8 AW #1

Consider the following class declaration:

public class Circle

{

private double radius;

public Circle(double r)

{

radius = r;

}

public double getArea()

{

return Math.PI \* radius \* radius;

}

public double getRadius()

{

return radius;

}

}

(a.)

public String toString()

{

// Create a string describing the circle.

String str = "Radius: " + radius +

"\nArea: " + getArea();

// Return the string.

return str;

}

(b.)

public boolean equals(Circle object2)

{

boolean status;

// Determine whether this object's Radius and

// Area are equal to object2's

// Radius and Area.

if (radius.equals(object2.radius) && getArea() == object2.getArea())

status = true; // Yes, the objects are equal.

else status = false; // No, the objects are not equal.

// Return the value in status.

return status;

}

\*\*so it doesnt specify to radius and area in part (b) of this problem, i know that technically you dont need to include the part of the code above " && getArea() == object2.getArea())" because if the radius is the same both the circles will be the same, i guess my question is will what i wrote in the code above work? comparing to methods with == ? Below if the code just comparing the radius:

(b.)

public boolean equals(Circle object2)

{

boolean status;

// Determine whether this object's Radius and

// Area are equal to object2's

// Radius and Area.

if (radius.equals(object2.radius))

status = true; // Yes, the objects are equal.

else status = false; // No, the objects are not equal.

// Return the value in status.

return status;

}

(c.)

public boolean greaterThan(Circle object2)

{

boolean status;

if (object2.getArea() > getArea())

status = true; // Yes, object2 is greater.

else status = false; // No, object2 is not greater

// Return the value in status.

return status;

}