

SD2 - Worksheet 3 - 8%



Student name:	Asralt					
Student number:	Enkhbadral					
Faculty:	Computing Science					
Course:	BSCH/BSCO/EXCH			Stage/year:	2	
Subject:	Software Development 2					
Study Mode:	Full time	<input checked="" type="checkbox"/>		Part-time	<input type="checkbox"/>	
Lecturer Name:	Gemma Deery					
Assignment Title:	Worksheet 3					
Date due:						
Date submitted:						
Plagiarism disclaimer: <p><i>I understand that plagiarism is a serious offence and have read and understood the college policy on plagiarism. I also understand that I may receive a mark of zero if I have not identified and properly attributed sources which have been used, referred to, or have in any way influenced the preparation of this assignment, or if I have knowingly allowed others to plagiarise my work in this way.</i></p> <p><i>I hereby certify that this assignment is my own work, based on my personal study and/or research, and that I have acknowledged all material and sources used in its preparation. I also certify that the assignment has not previously been submitted for assessment and that I have not copied in part or whole or otherwise plagiarised the work of anyone else, including other students.</i></p> <p>Signed: _____ Date: _____</p>						
<p>Please note: Students MUST retain a hard / soft copy of ALL assignments as well as a receipt issued and signed by a member of Faculty as proof of submission.</p>						

Repo Link:

Tasks

Student Name Student Number

Task1.

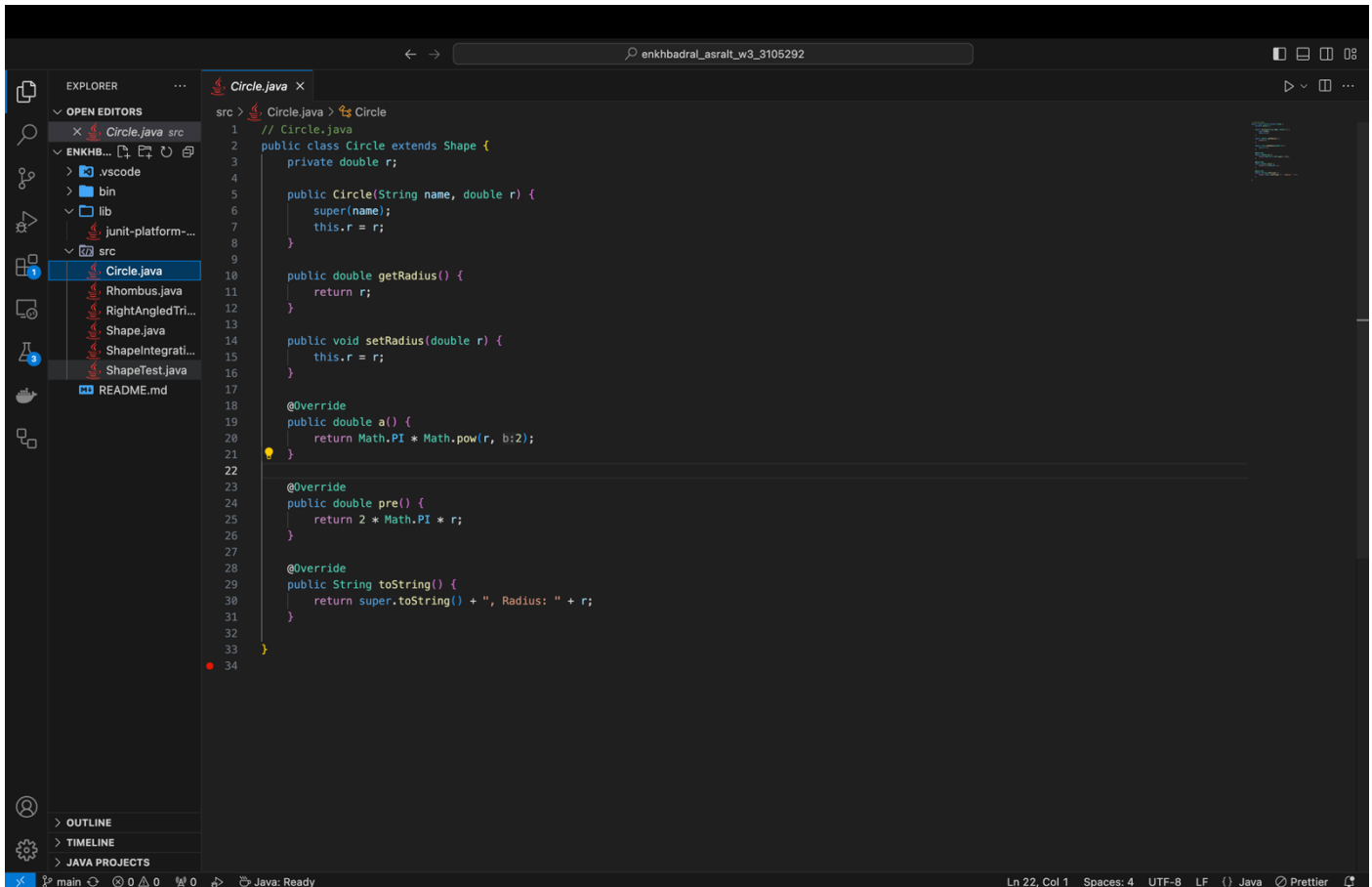
Student Name Student Number

SD2 - Worksheet 3 - 8%

```
src > Shape.java > Shape > setName(String)
1 public abstract class Shape {
2     private String name;
3
4     public Shape(String name) {
5         this.name = name;
6     }
7
8     public String getName() {
9         return name;
10    }
11
12    public void setName(String name) {
13        this.name = name;
14    }
15
16    public abstract double a();
17
18    public abstract double pre();
19
20    public String toString() {
21        return "shape is " + name;
22    }
23
24 }
```

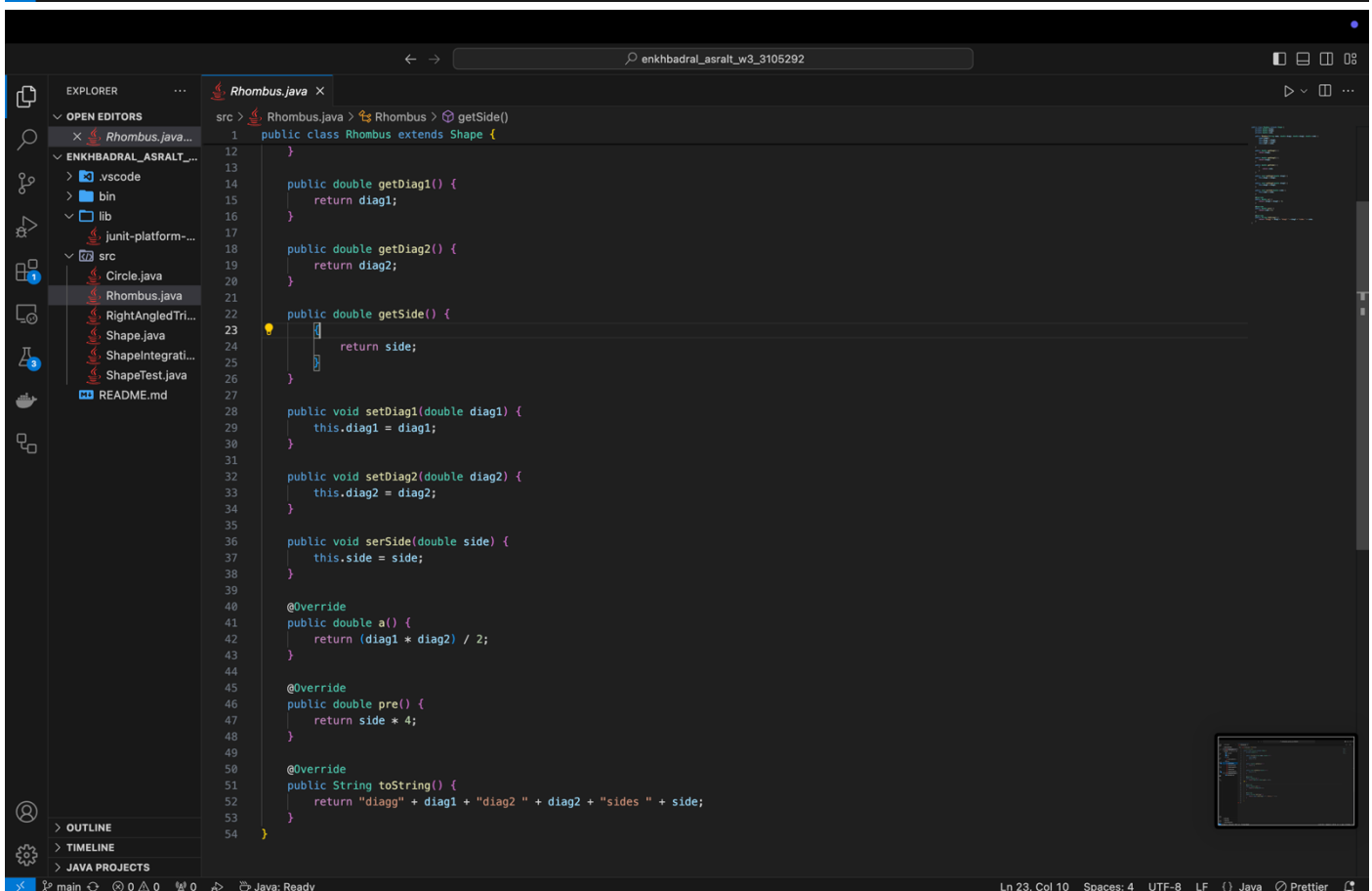
Student Name Student Number

SD2 - Worksheet 3 - 8%



The screenshot shows the Visual Studio Code editor with the file `Circle.java` open. The Explorer sidebar on the left shows the project structure with files like `Circle.java`, `Rhombus.java`, `RightAngledTri...`, `Shape.java`, `ShapeIntegrati...`, `ShapeTest.java`, and `README.md`. The main editor area displays the code for `Circle.java`, which is a class extending `Shape`. The code includes a private field `r`, a constructor `Circle(String name, double r)`, and methods `getRadius()`, `setRadius(double r)`, `a()`, `pre()`, and `toString()`.

```
1 // Circle.java
2 public class Circle extends Shape {
3     private double r;
4
5     public Circle(String name, double r) {
6         super(name);
7         this.r = r;
8     }
9
10    public double getRadius() {
11        return r;
12    }
13
14    public void setRadius(double r) {
15        this.r = r;
16    }
17
18    @Override
19    public double a() {
20        return Math.PI * Math.pow(r, 2);
21    }
22
23    @Override
24    public double pre() {
25        return 2 * Math.PI * r;
26    }
27
28    @Override
29    public String toString() {
30        return super.toString() + ", Radius: " + r;
31    }
32 }
33
34
```

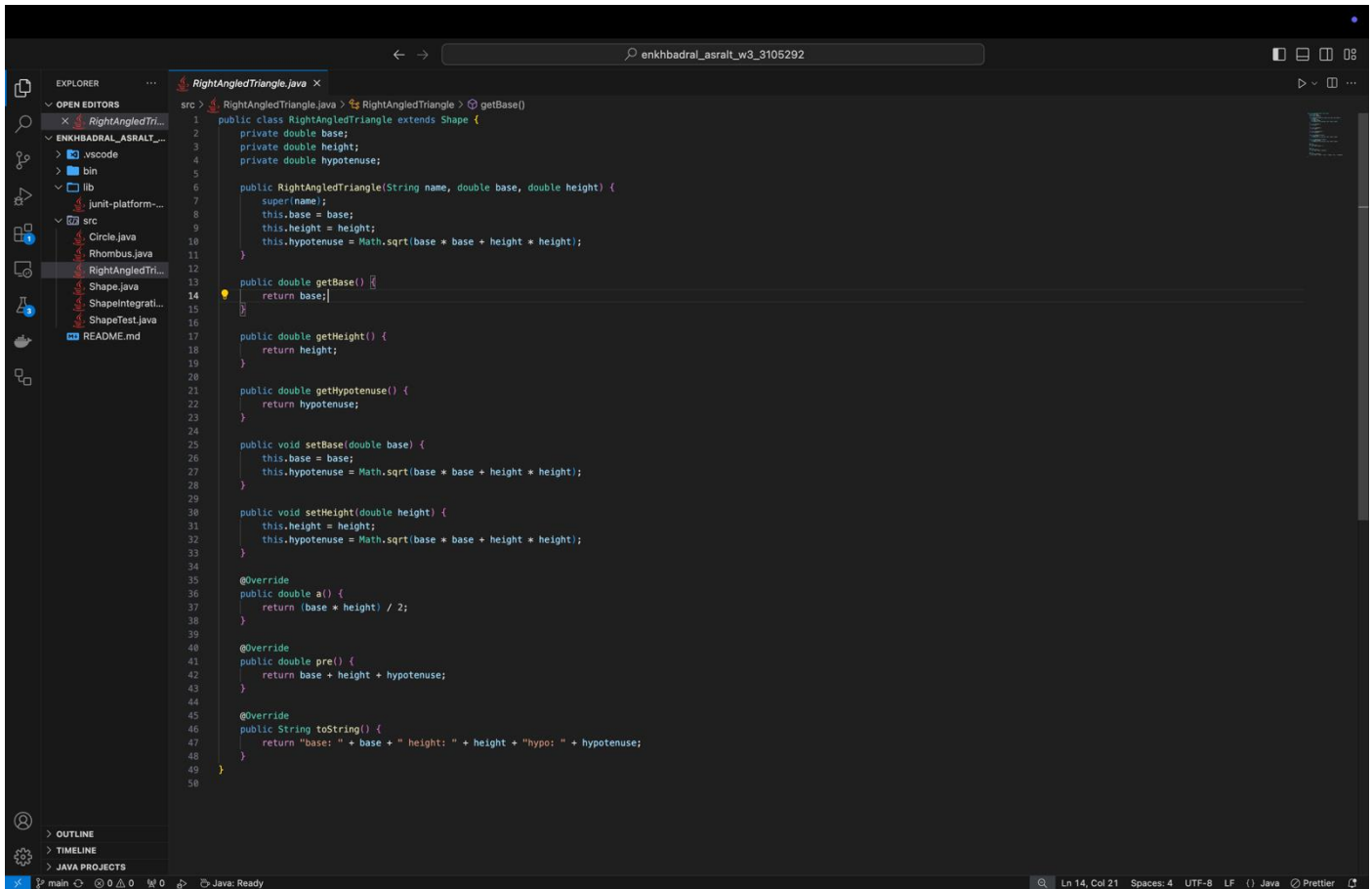


The screenshot shows the Visual Studio Code editor with the file `Rhombus.java` open. The Explorer sidebar on the left shows the project structure with files like `Circle.java`, `Rhombus.java`, `RightAngledTri...`, `Shape.java`, `ShapeIntegrati...`, `ShapeTest.java`, and `README.md`. The main editor area displays the code for `Rhombus.java`, which is a class extending `Shape`. The code includes methods `getDiag1()`, `getDiag2()`, `getSide()`, `setDiag1(double diag1)`, `setDiag2(double diag2)`, `serSide(double side)`, `a()`, `pre()`, and `toString()`.

```
1 public class Rhombus extends Shape {
2
3     public double getDiag1() {
4         return diag1;
5     }
6
7     public double getDiag2() {
8         return diag2;
9     }
10
11    public double getSide() {
12        return side;
13    }
14
15    public void setDiag1(double diag1) {
16        this.diag1 = diag1;
17    }
18
19    public void setDiag2(double diag2) {
20        this.diag2 = diag2;
21    }
22
23    public void serSide(double side) {
24        this.side = side;
25    }
26
27    @Override
28    public double a() {
29        return (diag1 * diag2) / 2;
30    }
31
32    @Override
33    public double pre() {
34        return side * 4;
35    }
36
37    @Override
38    public String toString() {
39        return "diag1 " + diag1 + "diag2 " + diag2 + "sides " + side;
40    }
41 }
42
43
```

Student Name Student Number

SD2 - Worksheet 3 - 8%

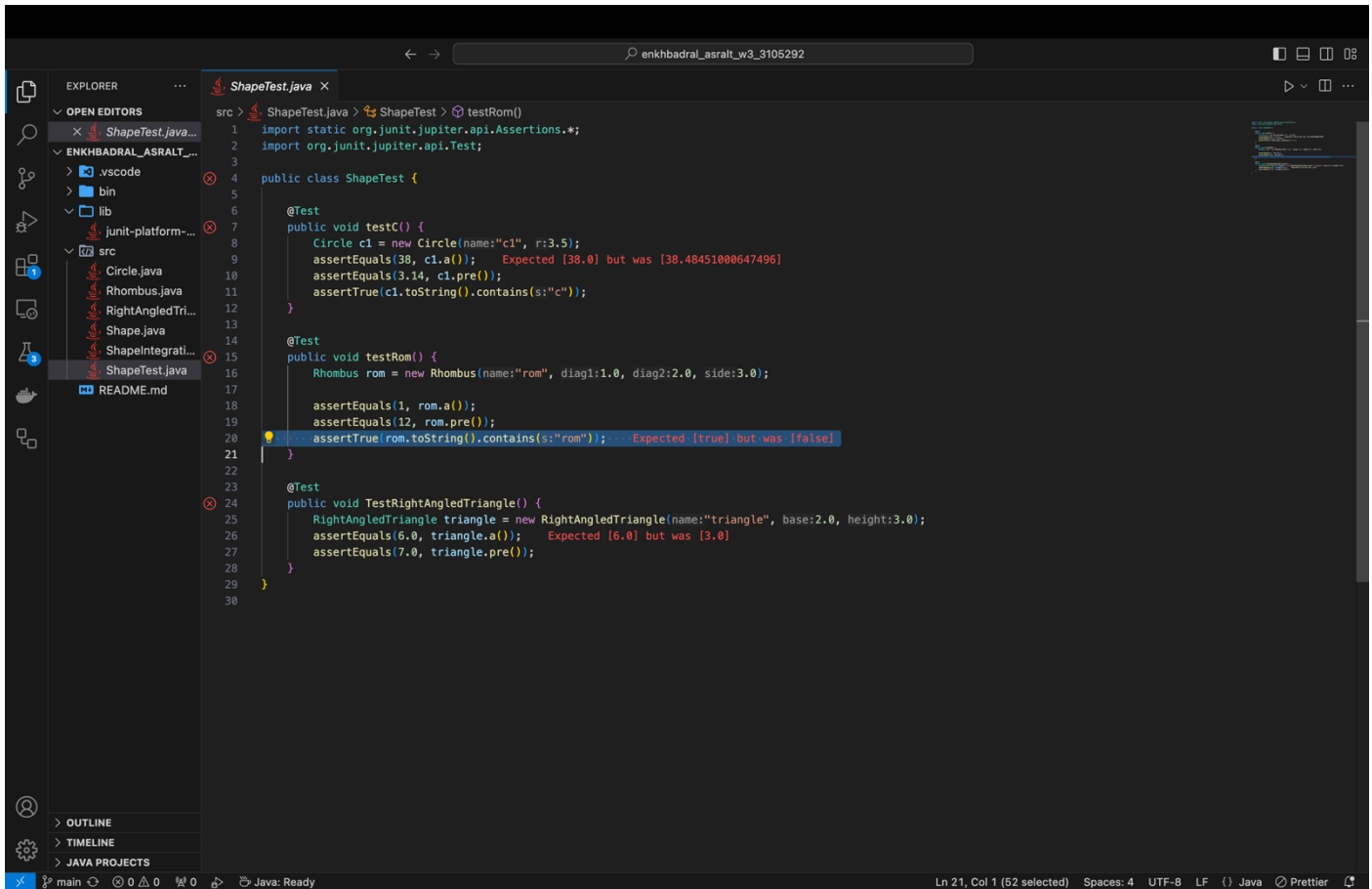


```
src > RightAngledTriangle.java > RightAngledTriangle > getBase()
1  public class RightAngledTriangle extends Shape {
2      private double base;
3      private double height;
4      private double hypotenuse;
5
6      public RightAngledTriangle(String name, double base, double height) {
7          super(name);
8          this.base = base;
9          this.height = height;
10         this.hypotenuse = Math.sqrt(base * base + height * height);
11     }
12
13     public double getBase() {
14         return base;
15     }
16
17     public double getHeight() {
18         return height;
19     }
20
21     public double getHypotenuse() {
22         return hypotenuse;
23     }
24
25     public void setBase(double base) {
26         this.base = base;
27         this.hypotenuse = Math.sqrt(base * base + height * height);
28     }
29
30     public void setHeight(double height) {
31         this.height = height;
32         this.hypotenuse = Math.sqrt(base * base + height * height);
33     }
34
35     @Override
36     public double a() {
37         return (base * height) / 2;
38     }
39
40     @Override
41     public double pre() {
42         return base + height + hypotenuse;
43     }
44
45     @Override
46     public String toString() {
47         return "base: " + base + " height: " + height + " hypo: " + hypotenuse;
48     }
49 }
50
```

Task2.

Student Name Student Number

SD2 - Worksheet 3 - 8%

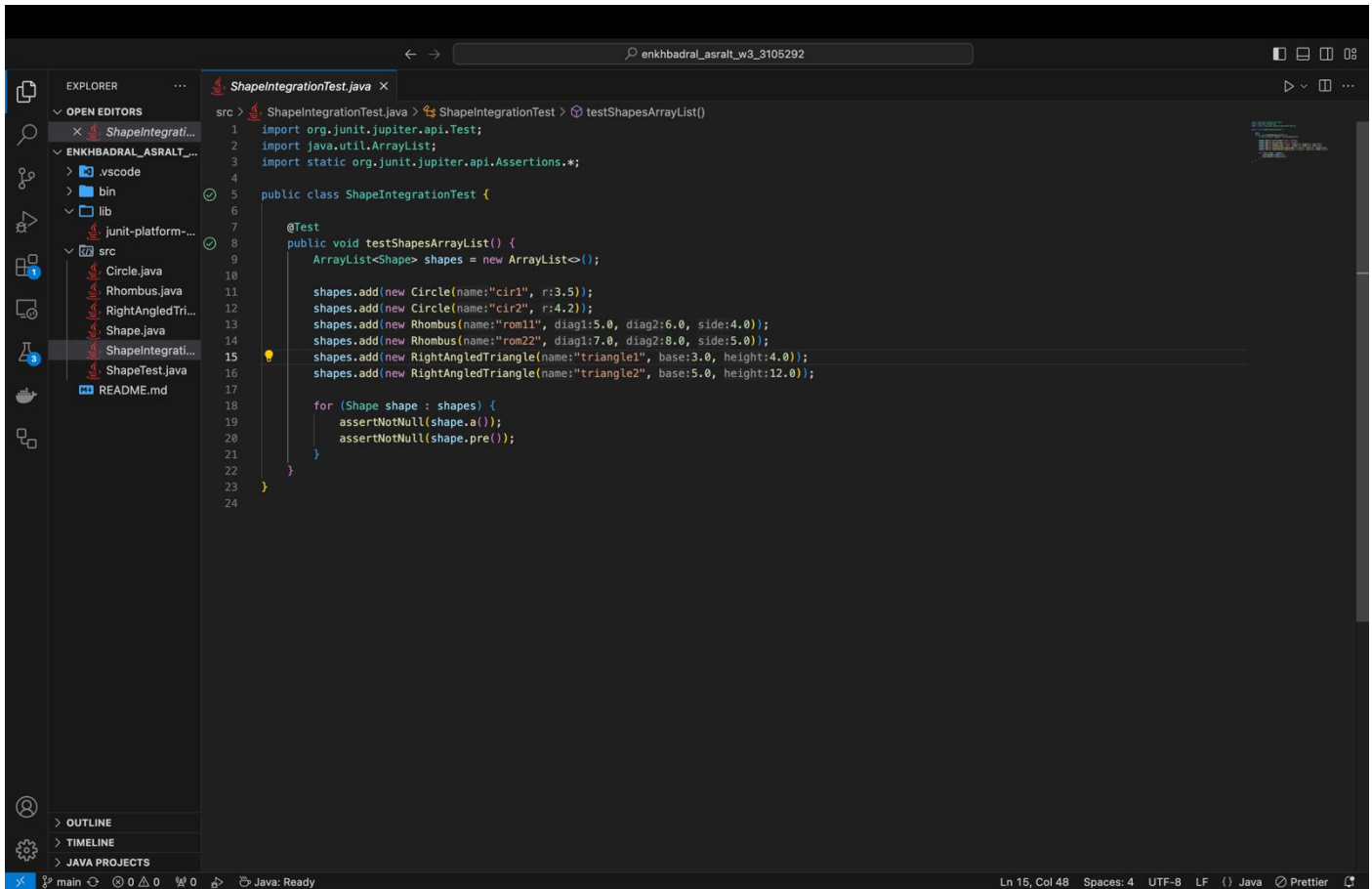


```
src > ShapeTest.java > ShapeTest > testRom()
1  import static org.junit.jupiter.api.Assertions.*;
2  import org.junit.jupiter.api.Test;
3
4  public class ShapeTest {
5
6
7      @Test
8      public void testC() {
9          Circle c1 = new Circle(name:"c1", r:3.5);
10         assertEquals(38, c1.a()); Expected [38.0] but was [38.48451000647496]
11         assertEquals(3.14, c1.pre());
12         assertTrue(c1.toString().contains(s:"c"));
13     }
14
15     @Test
16     public void testRom() {
17         Rhombus rom = new Rhombus(name:"rom", diag1:1.0, diag2:2.0, side:3.0);
18
19         assertEquals(1, rom.a());
20         assertEquals(12, rom.pre());
21         assertTrue(rom.toString().contains(s:"rom")); Expected [true] but was [false]
22     }
23
24     @Test
25     public void TestRightAngledTriangle() {
26         RightAngledTriangle triangle = new RightAngledTriangle(name:"triangle", base:2.0, height:3.0);
27         assertEquals(6.0, triangle.a()); Expected [6.0] but was [3.0]
28         assertEquals(7.0, triangle.pre());
29     }
30 }
```

Task3.

Student Name Student Number

SD2 - Worksheet 3 - 8%



The screenshot shows a Visual Studio Code editor window with a dark theme. The Explorer sidebar on the left shows a project structure with folders like .vscode, bin, lib, and src. The src folder is expanded, showing files like Circle.java, Rhombus.java, RightAngledTri..., Shape.java, ShapeIntegrati..., ShapeTest.java, and README.md. The main editor area displays the file ShapeIntegrationTest.java. The code is a Java test class using JUnit 5. It imports org.junit.jupiter.api.Test, java.util.ArrayList, and org.junit.jupiter.api.Assertions.*. The class ShapeIntegrationTest contains a single test method testShapesArrayList() annotated with @Test. This method creates an ArrayList of Shape objects, adds five shapes (two Circles, two Rhombuses, and one RightAngledTriangle), and then iterates over the list to assert that each shape is not null and has a pre() method.

```
src > ShapeIntegrationTest.java > ShapeIntegrationTest > testShapesArrayList()
1  import org.junit.jupiter.api.Test;
2  import java.util.ArrayList;
3  import static org.junit.jupiter.api.Assertions.*;
4
5  public class ShapeIntegrationTest {
6
7      @Test
8      public void testShapesArrayList() {
9          ArrayList<Shape> shapes = new ArrayList<>();
10
11          shapes.add(new Circle(name:"cir1", r:3.5));
12          shapes.add(new Circle(name:"cir2", r:4.2));
13          shapes.add(new Rhombus(name:"rom1", diag1:5.0, diag2:6.0, side:4.0));
14          shapes.add(new Rhombus(name:"rom2", diag1:7.0, diag2:8.0, side:5.0));
15          shapes.add(new RightAngledTriangle(name:"triangle1", base:3.0, height:4.0));
16          shapes.add(new RightAngledTriangle(name:"triangle2", base:5.0, height:12.0));
17
18          for (Shape shape : shapes) {
19              assertNotNull(shape.a());
20              assertNotNull(shape.pre());
21          }
22      }
23  }
24
```

Ln 15, Col 48 Spaces: 4 UTF-8 LF () Java Prettier

Student Name Student Number