

J Demo.java > Demo > main(String[])

```
1 public class Demo {
    Run | Debug
2     public static void main(String[] args) {
3         // Create a Person instance
4         Person bob = new Person(name:"Coach Bob", age:27, gender:"M");
5         System.out.println();
6         System.out.println(bob);
7
8         // Create a Student instance
9         Student lynne = new Student(name:"Lynne Brooke", age:16, gender:"F", id:"HS95129", gpa:3.5);
10        System.out.println(lynne);
11
12        // Create a Teacher instance
13        Teacher mrJava = new Teacher(name:"Duke Java", age:34, gender:"M", subject:"Computer Science",
14        salary:50000);
15        System.out.println(mrJava);
16
17        // Create a CollegeStudent instance
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Run: Demo + v [] [] ...

```
● PS C:\Users\jerja\OneDrive\Desktop\Java_Course\QAP3\Problem1> c:: cd 'c:\Users\jerja\OneDrive\Desktop\Java_Course\QAP3\Problem1'; & 'C:\Pro
iles\Eclipse Adoptium\jdk-21.0.5.11-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\jerja\AppData\Roaming\Co
r\workspaceStorage\329d725d608df3e5fcfd2c03145098d\redhat.java\jdt_ws\Problem1_e43e6719\bin' 'Demo'
```

Coach Bob, age: 27, gender: M

Lynne Brooke, age: 16, gender: F, ID Number: HS95129, GPA: 3.5

Duke Java, age: 34, gender: M, subject: Computer Science, annual salary: \$50000.0

Ima Frosh, age: 18, gender: F, ID Number: UCB123, GPA: 4.0, year: 1, major: English

Screenshot: Problem#1

```
J Demo.java > Demo > main(String[])
1 public class Demo {
2
3     Run | Debug
4     public static void main(String[] args) {
5         // Create the Point object
6         Point point = new Point(x:1.0f, y:2.0f);
7         System.out.println();
8         System.out.println("Point: " + point);
9
10        // Create the MovablePoint object
11        MovablePoint movablePoint = new MovablePoint(x:1.0f, y:2.0f, xSpeed:0.5f, ySpeed:1.5f);
12        System.out.println("MovablePoint before moving: " + movablePoint);
13
14        // Move the MovablePoint for the first time
15        movablePoint.move();
16        System.out.println("MovablePoint after first move: " + movablePoint);
17        // Move the MovablePoint again
18    }
19 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Run: Demo + v [] [] ...

```
PS C:\Users\jerja\OneDrive\Desktop\Java_Course\QAP3\Problem2> c::; cd 'c:\Users\jerja\OneDrive\Desktop\Java_Course\QAP3\Problem2'; & 'C:\Program Files\Eclipse Adoptium\jdk-21.0.5.11-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\jerja\AppData\Roaming\Code\workspaceStorage\65bc263246a73dd814fef5bb2a51a6f5\redhat.java\jdt_ws\Problem2_e43e671a\bin' 'Demo'
```

Point: (1.0,2.0)
MovablePoint before moving: (1.0,2.0), speed=(0.5, 1.5)
MovablePoint after first move: (1.5,3.5), speed=(0.5, 1.5)
MovablePoint after another move: (2.0,5.0), speed=(0.5, 1.5)

Screenshot: Problem#2

J Demo.java > Demo > main(String[])

```
1 public class Demo {
    Run | Debug
2     public static void main(String[] args) {
3         // Create objects from different shape classes
4         Shape[] shapes = new Shape[4];
5         shapes[0] = new Circle(radius:4);
6         shapes[1] = new Ellipse(a:9, b:6);
7         shapes[2] = new Triangle(side1:5, side2:6, side3:7);
8         shapes[3] = new Equilateral(side:8);
9
10        System.out.println(x:"=====");
11        System.out.println(x:"                SHAPE DETAILS REPORT                ");
12        System.out.println(x:"=====");
13        System.out.printf(format:"%-25s %-12s %-12s %-15s\n", ...args:"Shape Type", "Area", "Perimeter",
14        "Additional Info");
15        System.out.println(x:"-----");
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

m3_e43e671b\bin' 'Demo'

```
=====
                SHAPE DETAILS REPORT
=====
Shape Type                Area          Perimeter    Additional Info
-----
Circle                    50.27        25.13        Radius: 4.00
-----
Ellipse                    169.65       47.60        Major Axis: 9.00, Minor Axis: 6.00
-----
Triangle                   14.70        18.00        Sides: (5.00, 6.00, 7.00)
-----
Equilateral Triangle      27.71        24.00        Sides: (8.00, 8.00, 8.00)
-----
```

Screenshot: Problem#3

SHAPE DETAILS BEFORE SCALING

Shape Type	Area	Perimeter	Additional Info
Circle	50.27	25.13	Radius: 4.00
Ellipse	169.65	47.60	Major Axis: 9.00, Minor Axis: 6.00
Triangle	14.70	18.00	Sides: (5.00, 6.00, 7.00)
Equilateral Triangle	27.71	24.00	Sides: (8.00, 8.00, 8.00)

Applying scale factor: 2.0

SHAPE DETAILS AFTER SCALING

Shape Type	Area	Perimeter	Additional Info
Circle	201.06	50.27	Radius: 8.00
Ellipse	678.58	95.19	Major Axis: 18.00, Minor Axis: 12.00
Triangle	58.79	36.00	Sides: (10.00, 12.00, 14.00)
Equilateral Triangle	110.85	48.00	Sides: (16.00, 16.00, 16.00)

Assessment Feedback

1.Q) How many hours did it take you to complete this assessment?

1.A) It took me about 8 hours to complete this assignment

2.Q) What online resources you have used?

2.A) I used Lecture Notes & google(stack overflow,...)

3.Q) Did you need to ask any of your friends in solving the problems?

3.A) I did not ask of my classmates for help during this assignment

4.Q) Did you need to ask questions to any of your instructors?

4.A) No, I didn't need to ask any questions to my instructors during this assignment

5.Q) Rate (subjectively) the difficulty of each question from your own perspective, and whether you feel confident that you can solve a similar but different problem requiring some of the same techniques in the future

5.A)

Problem 1: Difficulty - 1/10. This problem was fairly easy. I did not run into anything to complicated.

Problem 2: Difficulty - 2/10. I ran into a few problems having the code come out complete, otherwise well done.

Problem 3: Difficulty - 2/10. This was relatively easy besides a few parts.

Problem 4: Difficulty - 2/10. Had a few problems with the scaling, otherwise good.