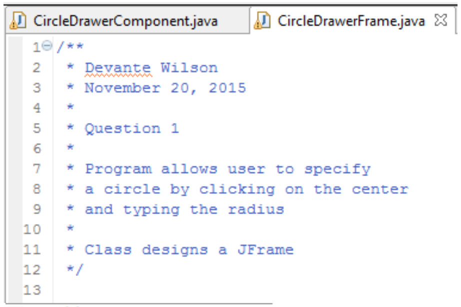
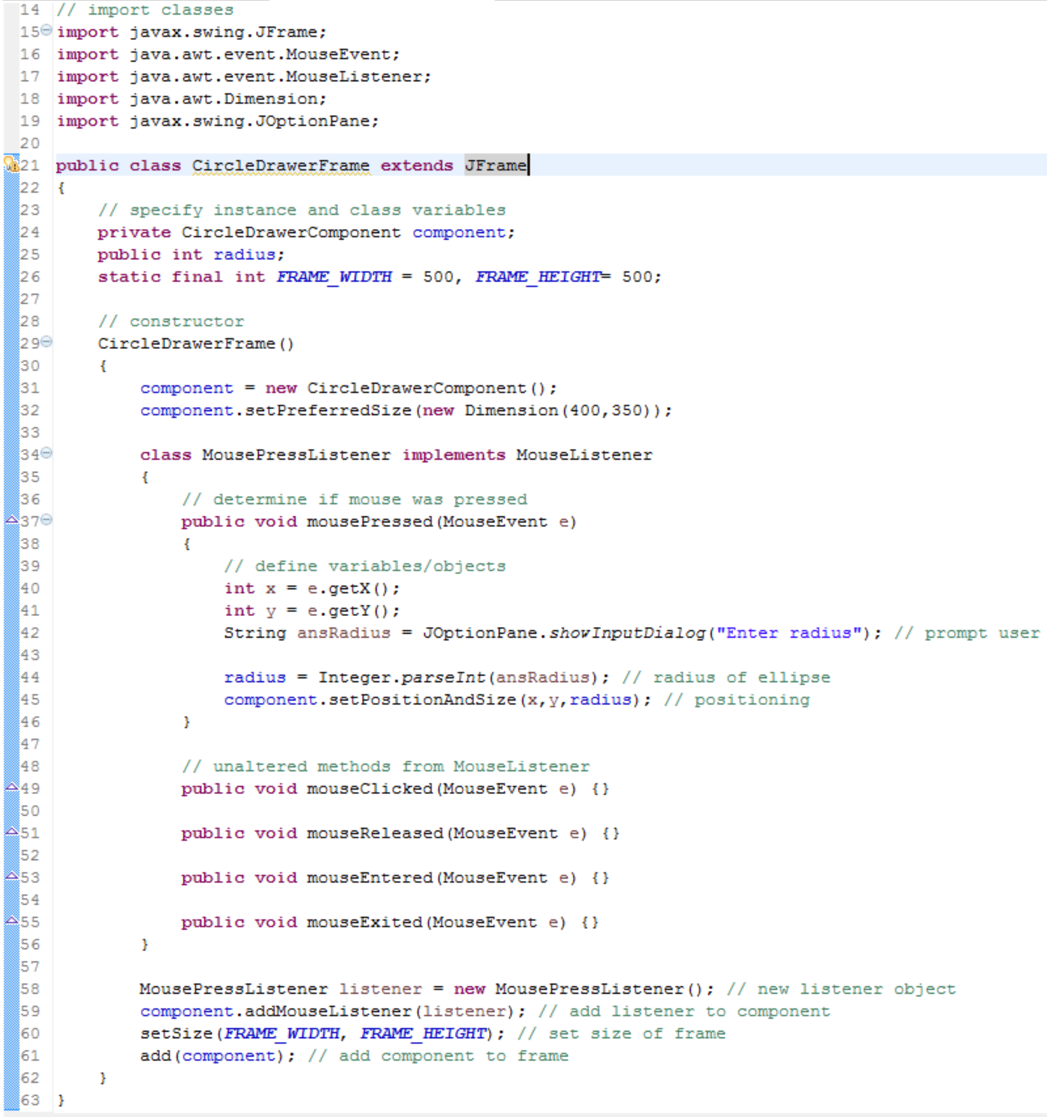
Devante Wilson
Group 7

Assignment 4: Object Oriented Design

1.



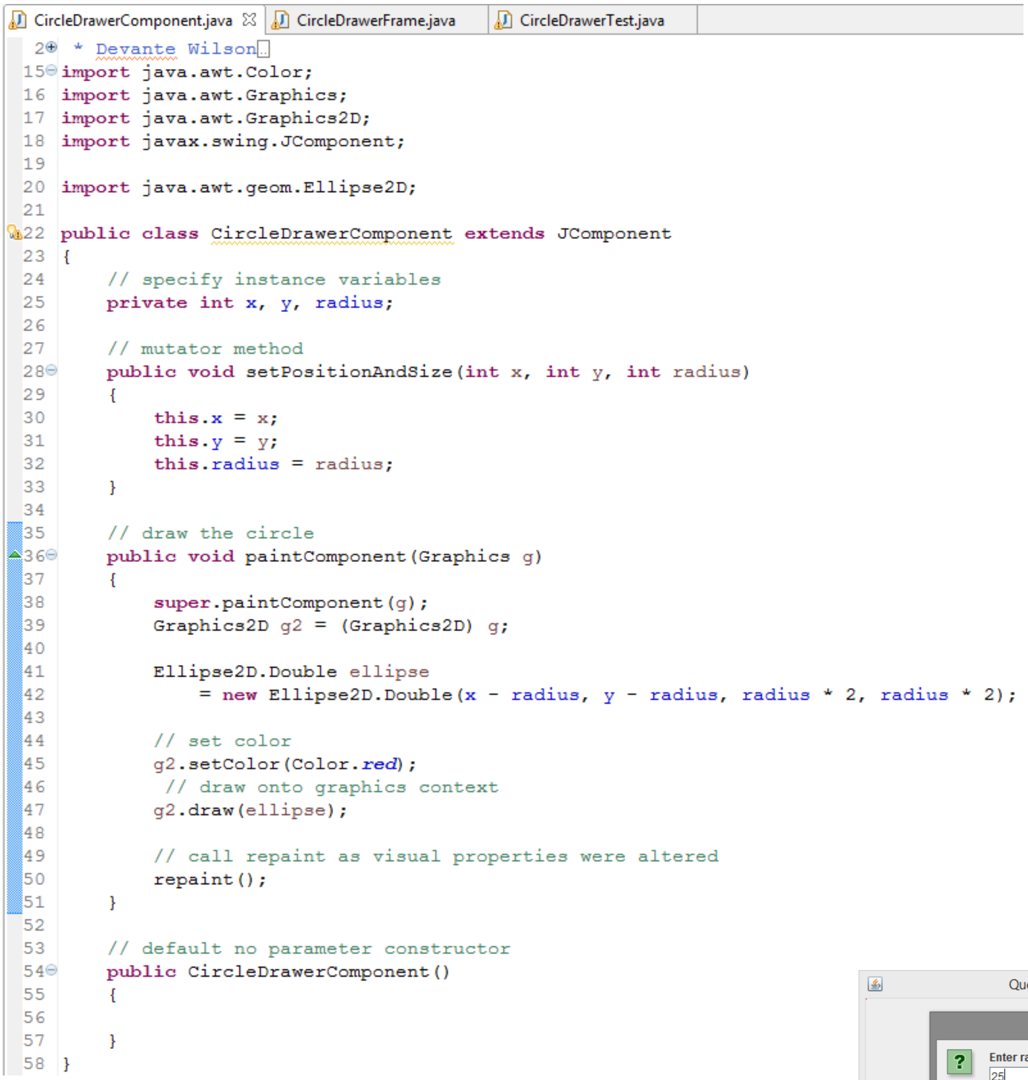
```
/**
 * Devante Wilson
 * November 20, 2015
 *
 * Question 1
 *
 * Program allows user to specify
 * a circle by clicking on the center
 * and typing the radius
 *
 * Class designs a JFrame
 */
```



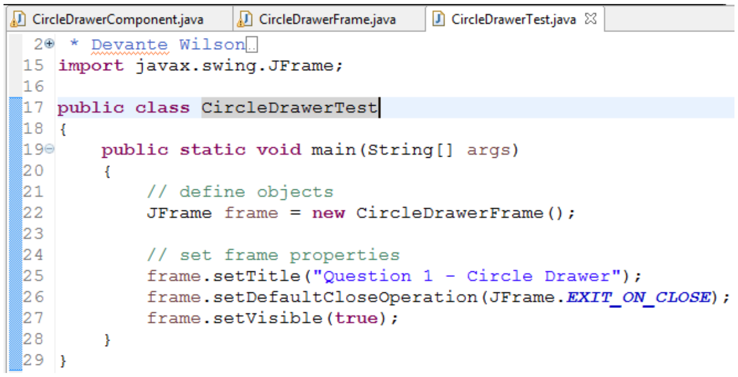
```

// import classes
import javax.swing.JFrame;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;
import java.awt.Dimension;
import javax.swing.JOptionPane;

public class CircleDrawerFrame extends JFrame
{
    // specify instance and class variables
    private CircleDrawerComponent component;
    public int radius;
    static final int FRAME_WIDTH = 500, FRAME_HEIGHT= 500;

    // constructor
    CircleDrawerFrame()
    {
        component = new CircleDrawerComponent();
        component.setPreferredSize(new Dimension(400,350));

        class MousePressListener implements MouseListener
        {
            // determine if mouse was pressed
            public void mousePressed(MouseEvent e)
            {
                // define variables/objects
                int x = e.getX();
                int y = e.getY();
                String ansRadius = JOptionPane.showInputDialog("Enter radius"); // prompt user

                radius = Integer.parseInt(ansRadius); // radius of ellipse
                component.setPositionAndSize(x,y,radius); // positioning
            }

            // unaltered methods from MouseListener
            public void mouseClicked(MouseEvent e) {}

            public void mouseReleased(MouseEvent e) {}

            public void mouseEntered(MouseEvent e) {}

            public void mouseExited(MouseEvent e) {}
        }

        MousePressListener listener = new MousePressListener(); // new listener object
        component.addMouseListener(listener); // add listener to component
        setSize(FRAME_WIDTH, FRAME_HEIGHT); // set size of frame
        add(component); // add component to frame
    }
}
```

Devante Wilson
Group 7

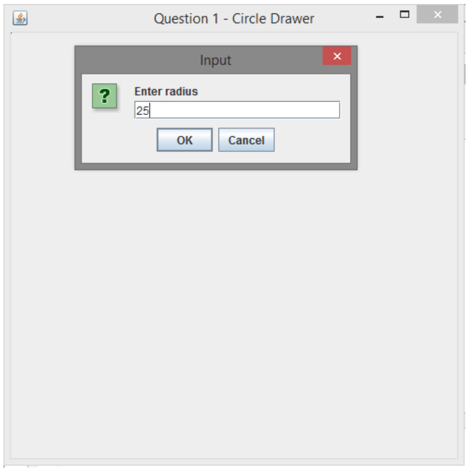
Assignment 4: Object Oriented Design



```java
 * Devante Wilson
import java.awt.Color;
import java.awt.Graphics;
import java.awt.Graphics2D;
import javax.swing.JComponent;

import java.awt.geom.Ellipse2D;

public class CircleDrawerComponent extends JComponent
{
    // specify instance variables
    private int x, y, radius;

    // mutator method
    public void setPositionAndSize(int x, int y, int radius)
    {
        this.x = x;
        this.y = y;
        this.radius = radius;
    }

    // draw the circle
    public void paintComponent(Graphics g)
    {
        super.paintComponent(g);
        Graphics2D g2 = (Graphics2D) g;

        Ellipse2D.Double ellipse
            = new Ellipse2D.Double(x - radius, y - radius, radius * 2, radius * 2);

        // set color
        g2.setColor(Color.red);
         // draw onto graphics context
        g2.draw(ellipse);

        // call repaint as visual properties were altered
        repaint();
    }

    // default no parameter constructor
    public CircleDrawerComponent()
    {

    }
}
```

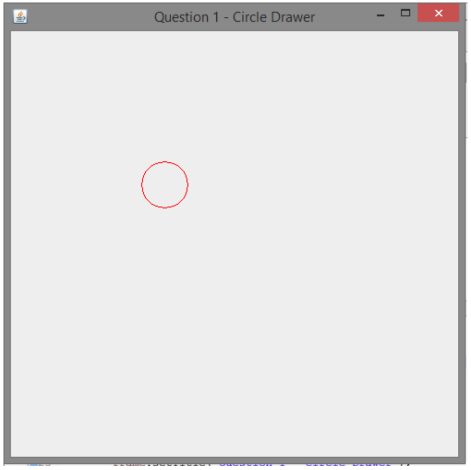


```java
 * Devante Wilson
import javax.swing.JFrame;

public class CircleDrawerTest
{
    public static void main(String[] args)
    {
        // define objects
        JFrame frame = new CircleDrawerFrame();

        // set frame properties
        frame.setTitle("Question 1 - Circle Drawer");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}
```



Enter radius

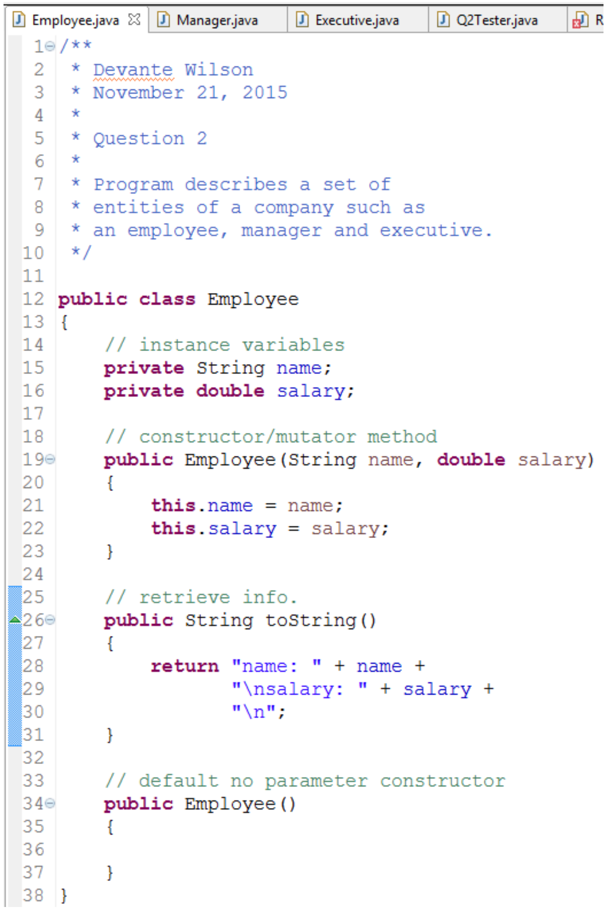
25



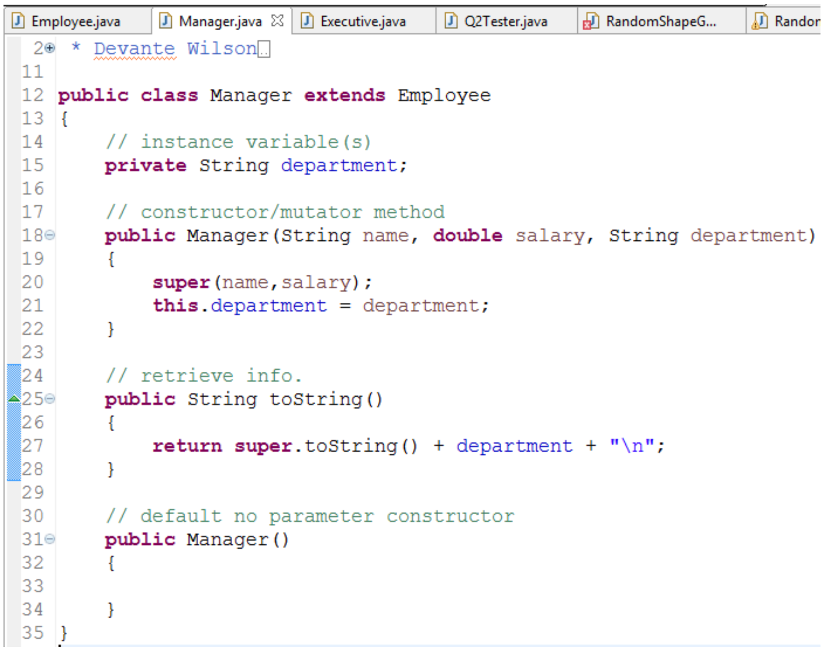
Devante Wilson
Group 7

Assignment 4: Object Oriented Design

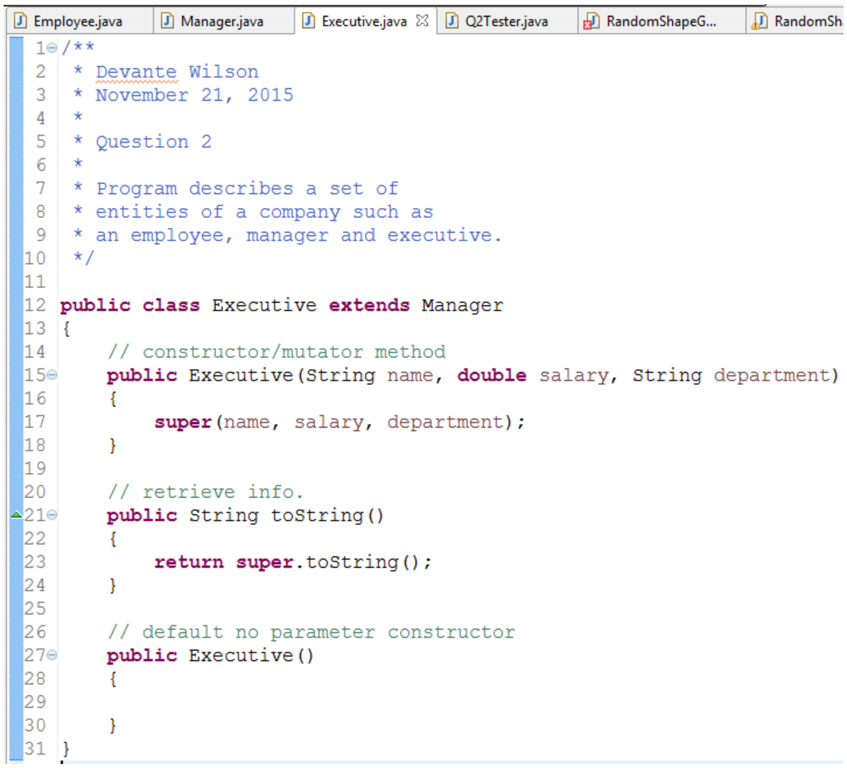
2.



```java
/**
 * Devante Wilson
 * November 21, 2015
 *
 * Question 2
 *
 * Program describes a set of
 * entities of a company such as
 * an employee, manager and executive.
 */

public class Employee
{
    // instance variables
    private String name;
    private double salary;

    // constructor/mutator method
    public Employee(String name, double salary)
    {
        this.name = name;
        this.salary = salary;
    }

    // retrieve info.
    public String toString()
    {
        return "name: " + name +
               "\nsalary: " + salary +
               "\n";
    }

    // default no parameter constructor
    public Employee()
    {

    }
}
```



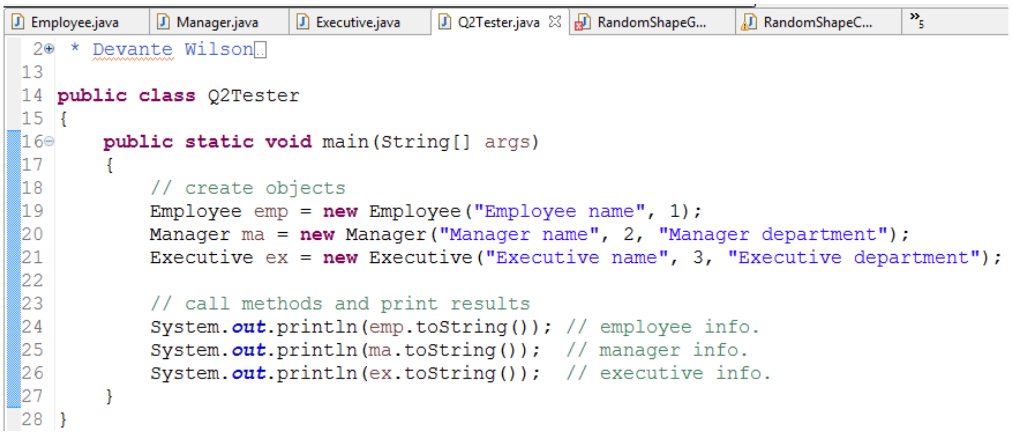
```java
 * Devante Wilson

public class Manager extends Employee
{
    // instance variable(s)
    private String department;

    // constructor/mutator method
    public Manager(String name, double salary, String department)
    {
        super(name,salary);
        this.department = department;
    }

    // retrieve info.
    public String toString()
    {
        return super.toString() + department + "\n";
    }

    // default no parameter constructor
    public Manager()
    {

    }
}
```



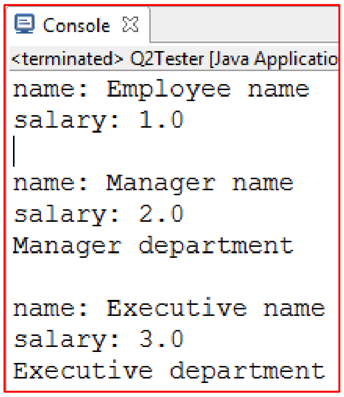
```java
/**
 * Devante Wilson
 * November 21, 2015
 *
 * Question 2
 *
 * Program describes a set of
 * entities of a company such as
 * an employee, manager and executive.
 */

public class Executive extends Manager
{
    // constructor/mutator method
    public Executive(String name, double salary, String department)
    {
        super(name, salary, department);
    }

    // retrieve info.
    public String toString()
    {
        return super.toString();
    }

    // default no parameter constructor
    public Executive()
    {

    }
}
```

Devante Wilson
Group 7

Assignment 4: Object Oriented Design

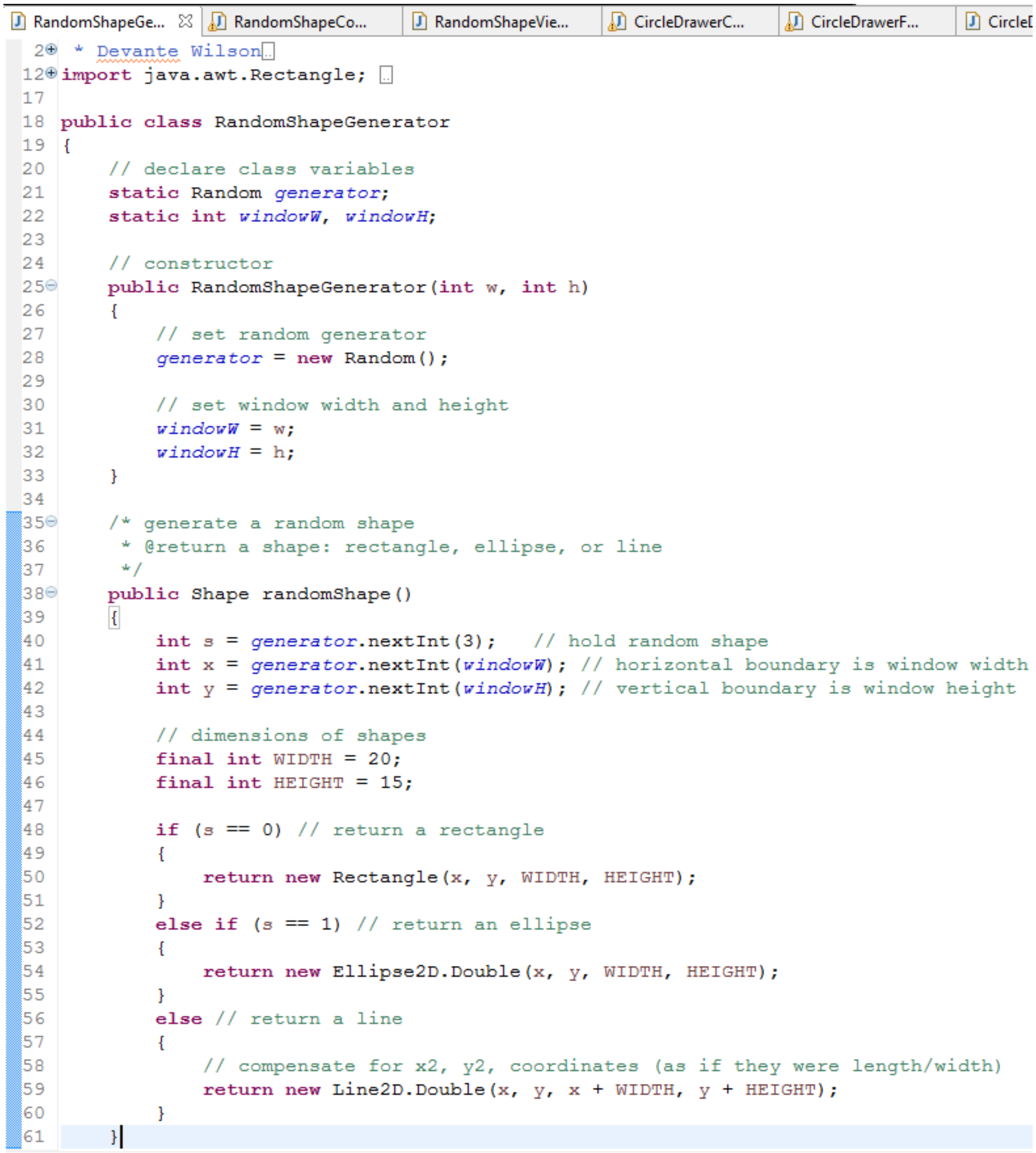


```
 * Devante Wilson

public class Q2Tester
{
    public static void main(String[] args)
    {
        // create objects
        Employee emp = new Employee("Employee name", 1);
        Manager ma = new Manager("Manager name", 2, "Manager department");
        Executive ex = new Executive("Executive name", 3, "Executive department");

        // call methods and print results
        System.out.println(emp.toString()); // employee info.
        System.out.println(ma.toString());  // manager info.
        System.out.println(ex.toString());  // executive info.
    }
}
```



```
name: Employee name
salary: 1.0

name: Manager name
salary: 2.0
Manager department

name: Executive name
salary: 3.0
Executive department
```

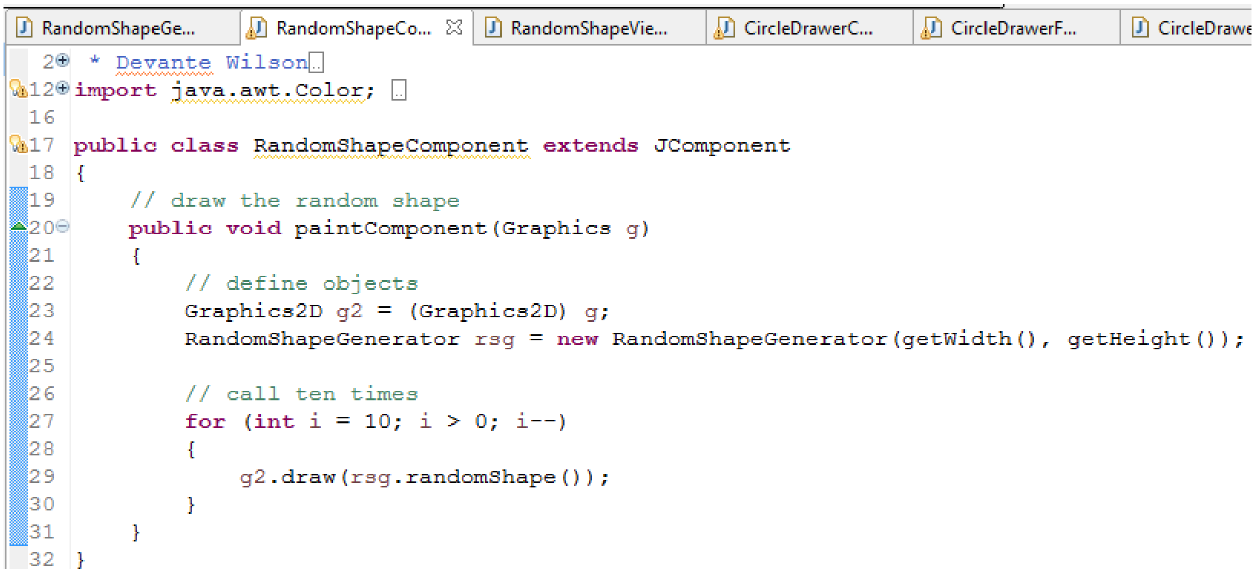
3.



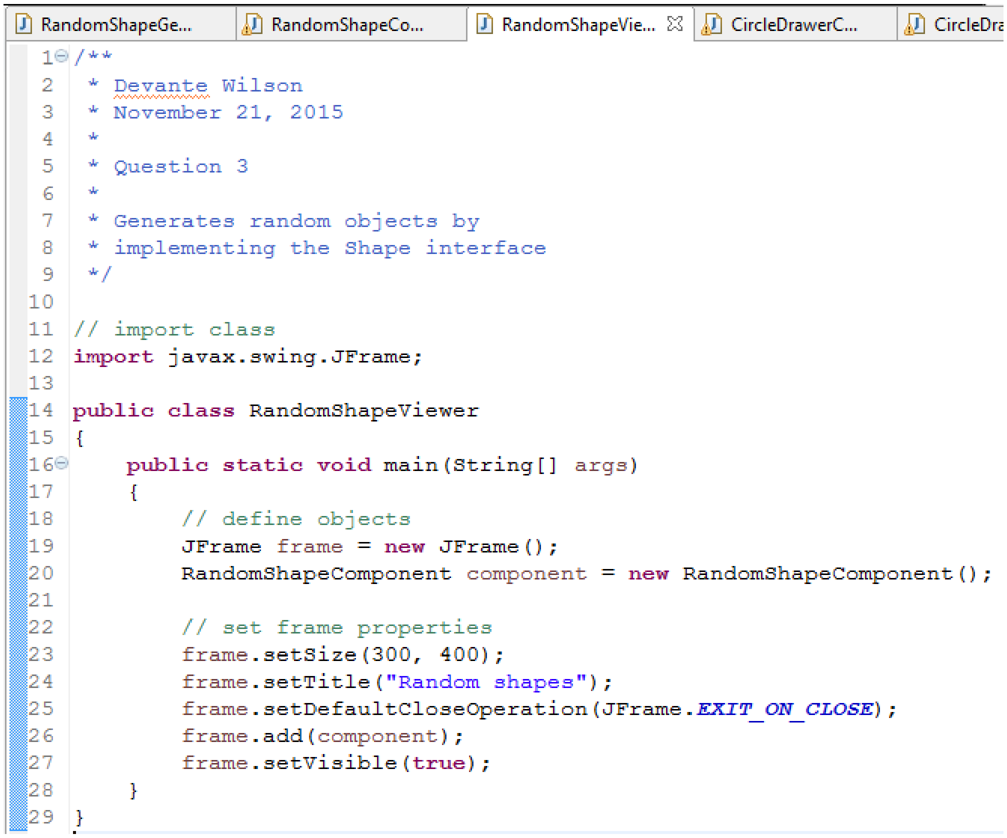
```
 * Devante Wilson
import java.awt.Rectangle;

public class RandomShapeGenerator
{
    // declare class variables
    static Random generator;
    static int windowW, windowH;

    // constructor
    public RandomShapeGenerator(int w, int h)
    {
        // set random generator
        generator = new Random();

        // set window width and height
        windowW = w;
        windowH = h;
    }

    /* generate a random shape
     * @return a shape: rectangle, ellipse, or line
     */
    public Shape randomShape()
    {
        int s = generator.nextInt(3);    // hold random shape
        int x = generator.nextInt(windowW); // horizontal boundary is window width
        int y = generator.nextInt(windowH); // vertical boundary is window height

        // dimensions of shapes
        final int WIDTH = 20;
        final int HEIGHT = 15;

        if (s == 0) // return a rectangle
        {
            return new Rectangle(x, y, WIDTH, HEIGHT);
        }
        else if (s == 1) // return an ellipse
        {
            return new Ellipse2D.Double(x, y, WIDTH, HEIGHT);
        }
        else // return a line
        {
            // compensate for x2, y2, coordinates (as if they were length/width)
            return new Line2D.Double(x, y, x + WIDTH, y + HEIGHT);
        }
    }
```

Devante Wilson
Group 7

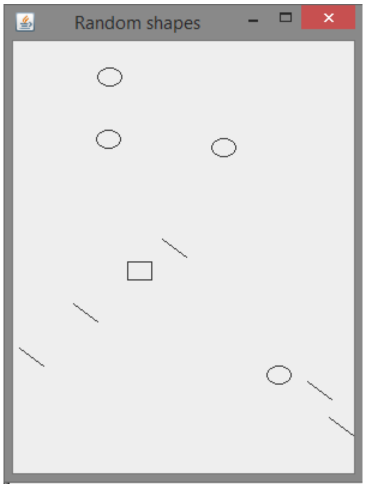
Assignment 4: Object Oriented Design



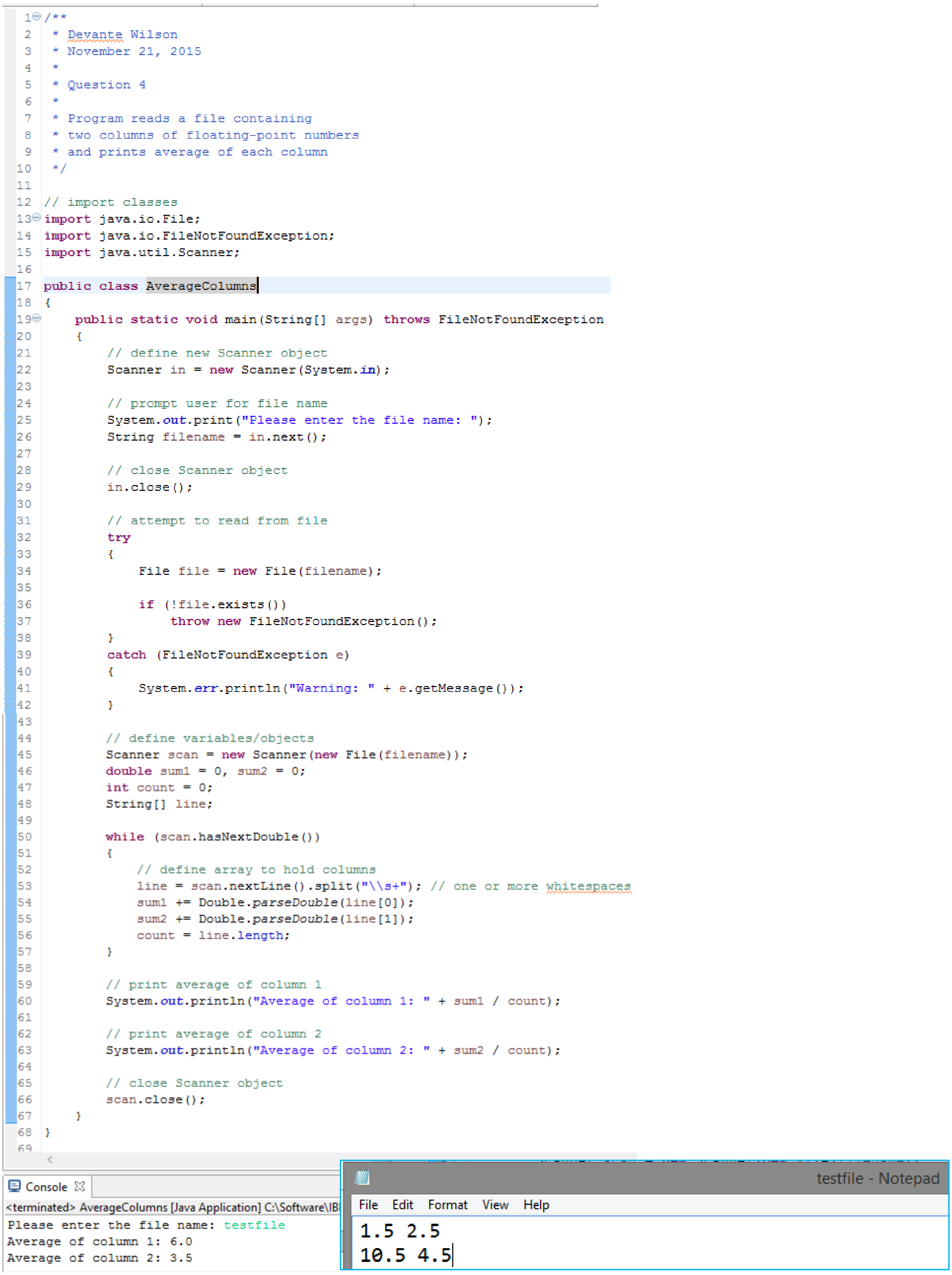
```
 * Devante Wilson
import java.awt.Color;

public class RandomShapeComponent extends JComponent
{
    // draw the random shape
    public void paintComponent(Graphics g)
    {
        // define objects
        Graphics2D g2 = (Graphics2D) g;
        RandomShapeGenerator rsg = new RandomShapeGenerator(getWidth(), getHeight());

        // call ten times
        for (int i = 10; i > 0; i--)
        {
            g2.draw(rsg.randomShape());
        }
    }
}
```



```
/**
 * Devante Wilson
 * November 21, 2015
 *
 * Question 3
 *
 * Generates random objects by
 * implementing the Shape interface
 */

// import class
import javax.swing.JFrame;

public class RandomShapeViewer
{
    public static void main(String[] args)
    {
        // define objects
        JFrame frame = new JFrame();
        RandomShapeComponent component = new RandomShapeComponent();

        // set frame properties
        frame.setSize(300, 400);
        frame.setTitle("Random shapes");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.add(component);
        frame.setVisible(true);
    }
}
```



4.

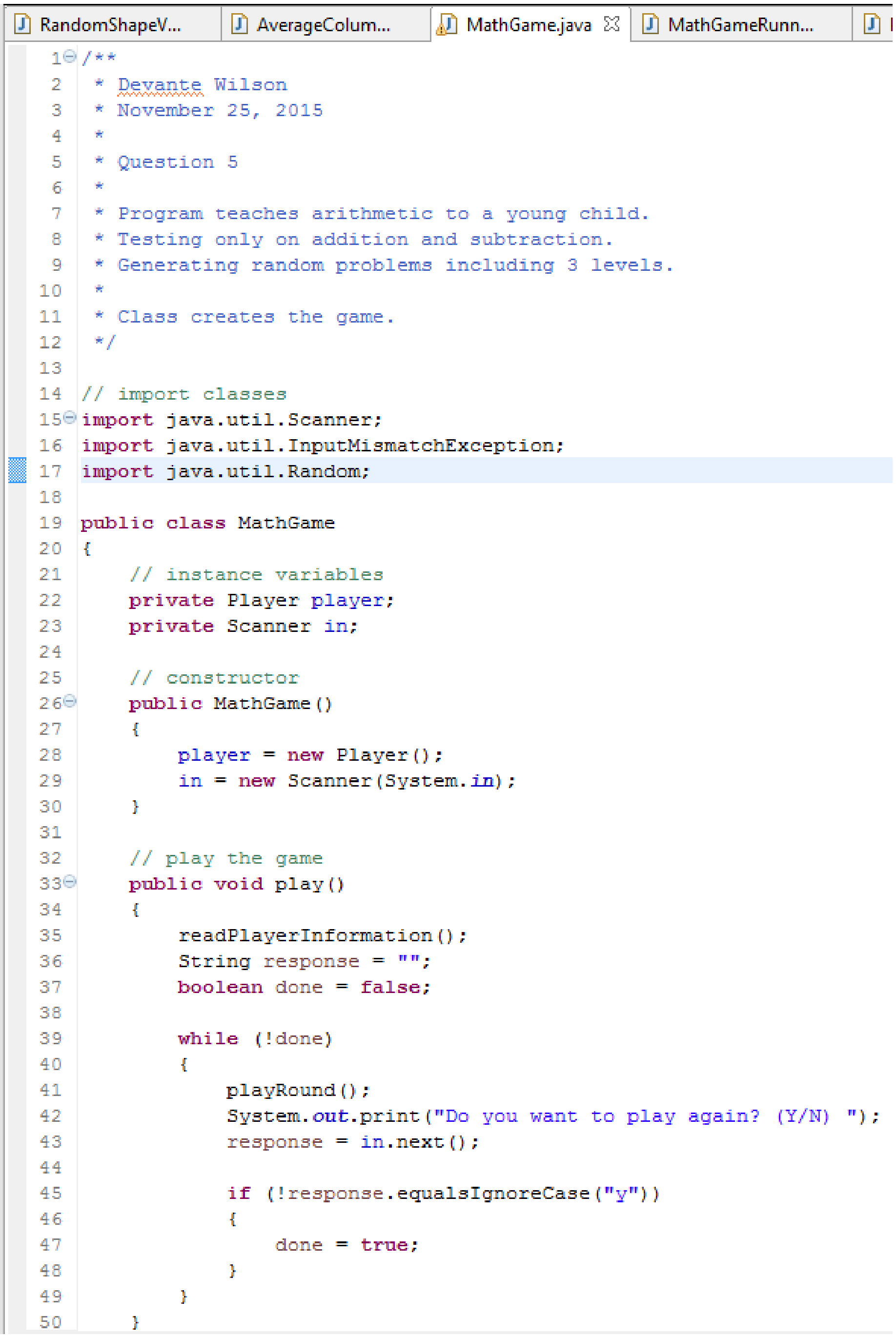


```java
/**
 * Devante Wilson
 * November 21, 2015
 *
 * Question 4
 *
 * Program reads a file containing
 * two columns of floating-point numbers
 * and prints average of each column
 */

// import classes
import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;

public class AverageColumns
{
    public static void main(String[] args) throws FileNotFoundException
    {
        // define new Scanner object
        Scanner in = new Scanner(System.in);

        // prompt user for file name
        System.out.print("Please enter the file name: ");
        String filename = in.next();

        // close Scanner object
        in.close();

        // attempt to read from file
        try
        {
            File file = new File(filename);

            if (!file.exists())
                throw new FileNotFoundException();
        }
        catch (FileNotFoundException e)
        {
            System.err.println("Warning: " + e.getMessage());
        }

        // define variables/objects
        Scanner scan = new Scanner(new File(filename));
        double sum1 = 0, sum2 = 0;
        int count = 0;
        String[] line;

        while (scan.hasNextDouble())
        {
            // define array to hold columns
            line = scan.nextLine().split("\\s+"); // one or more whitespaces
            sum1 += Double.parseDouble(line[0]);
            sum2 += Double.parseDouble(line[1]);
            count = line.length;
        }

        // print average of column 1
        System.out.println("Average of column 1: " + sum1 / count);

        // print average of column 2
        System.out.println("Average of column 2: " + sum2 / count);

        // close Scanner object
        scan.close();
    }
}
```

```
<terminated> AverageColumns [Java Application] C:\Software\IB
Please enter the file name: testfile
Average of column 1: 6.0
Average of column 2: 3.5
```

```
1.5 2.5
10.5 4.5
```

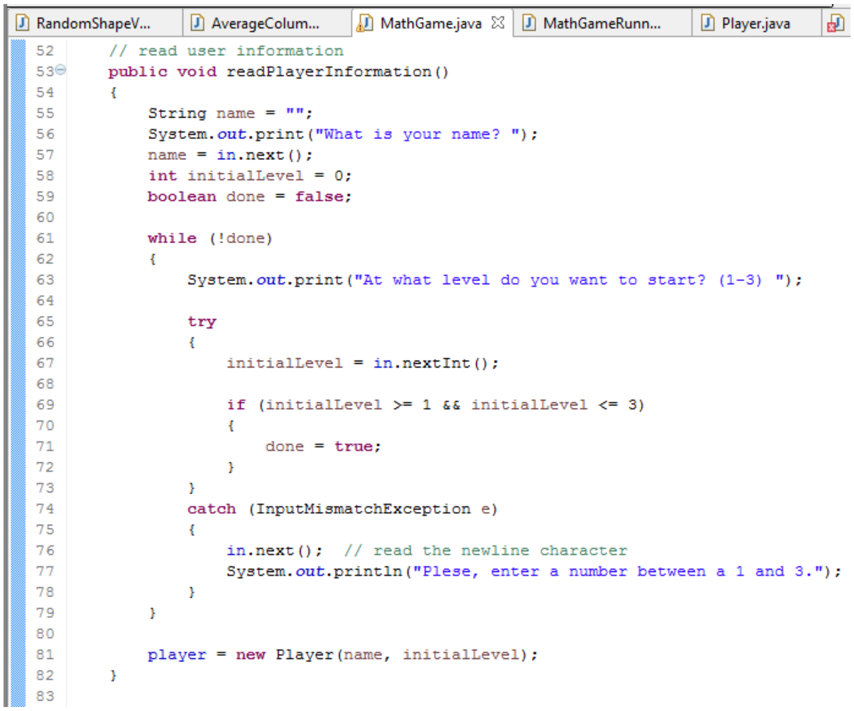
5.



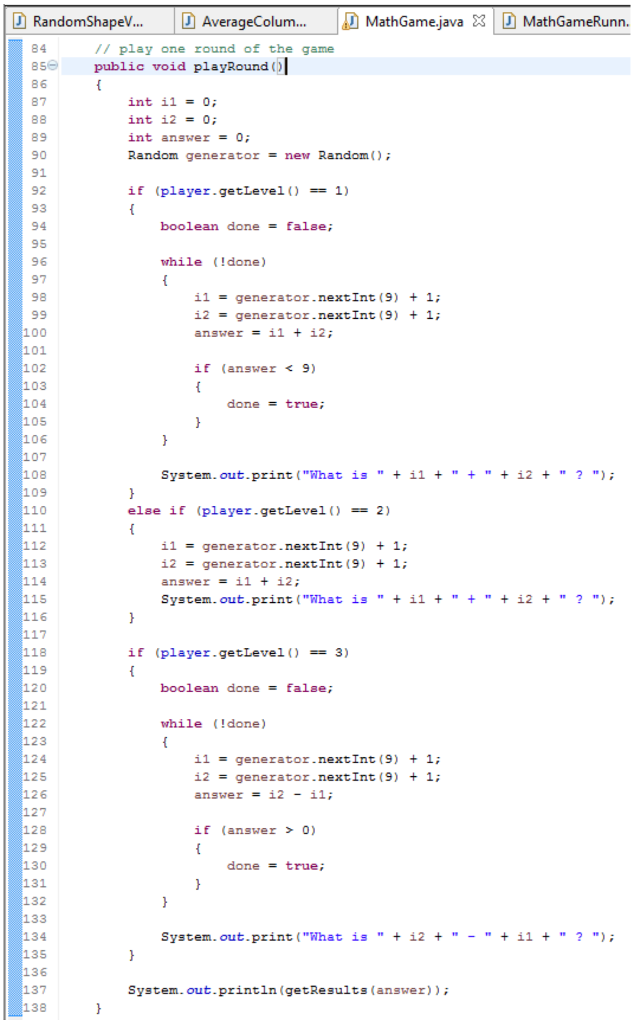
```java
/**
 * Devante Wilson
 * November 25, 2015
 *
 * Question 5
 *
 * Program teaches arithmetic to a young child.
 * Testing only on addition and subtraction.
 * Generating random problems including 3 levels.
 *
 * Class creates the game.
 */

// import classes
import java.util.Scanner;
import java.util.InputMismatchException;
import java.util.Random;

public class MathGame
{
    // instance variables
    private Player player;
    private Scanner in;

    // constructor
    public MathGame()
    {
        player = new Player();
        in = new Scanner(System.in);
    }

    // play the game
    public void play()
    {
        readPlayerInformation();
        String response = "";
        boolean done = false;

        while (!done)
        {
            playRound();
            System.out.print("Do you want to play again? (Y/N)  ");
            response = in.next();

            if (!response.equalsIgnoreCase("y"))
            {
                done = true;
            }
        }
    }
```

Devante Wilson
Group 7

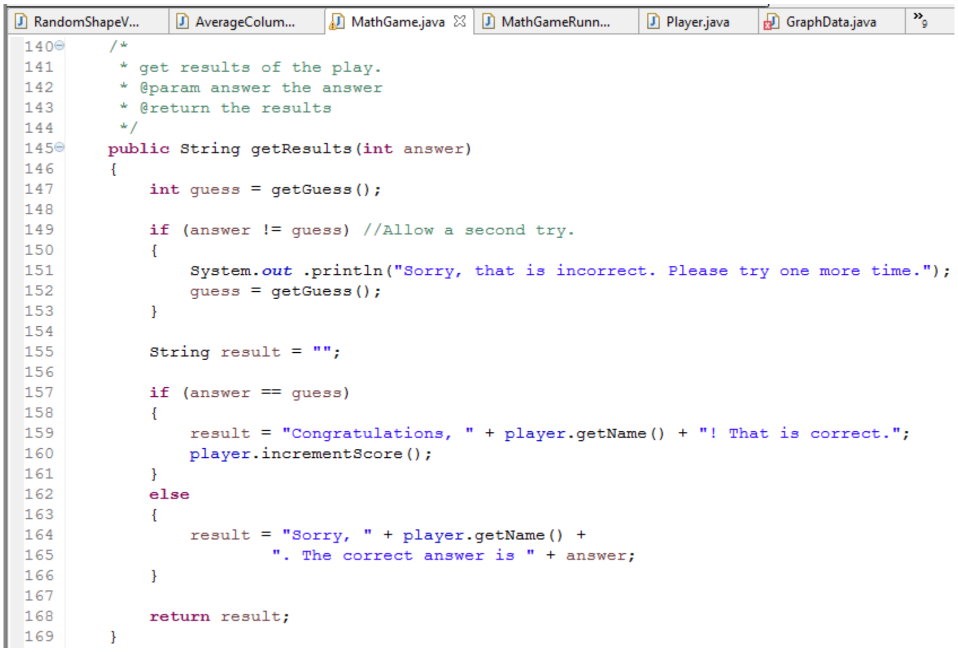
Assignment 4: Object Oriented Design



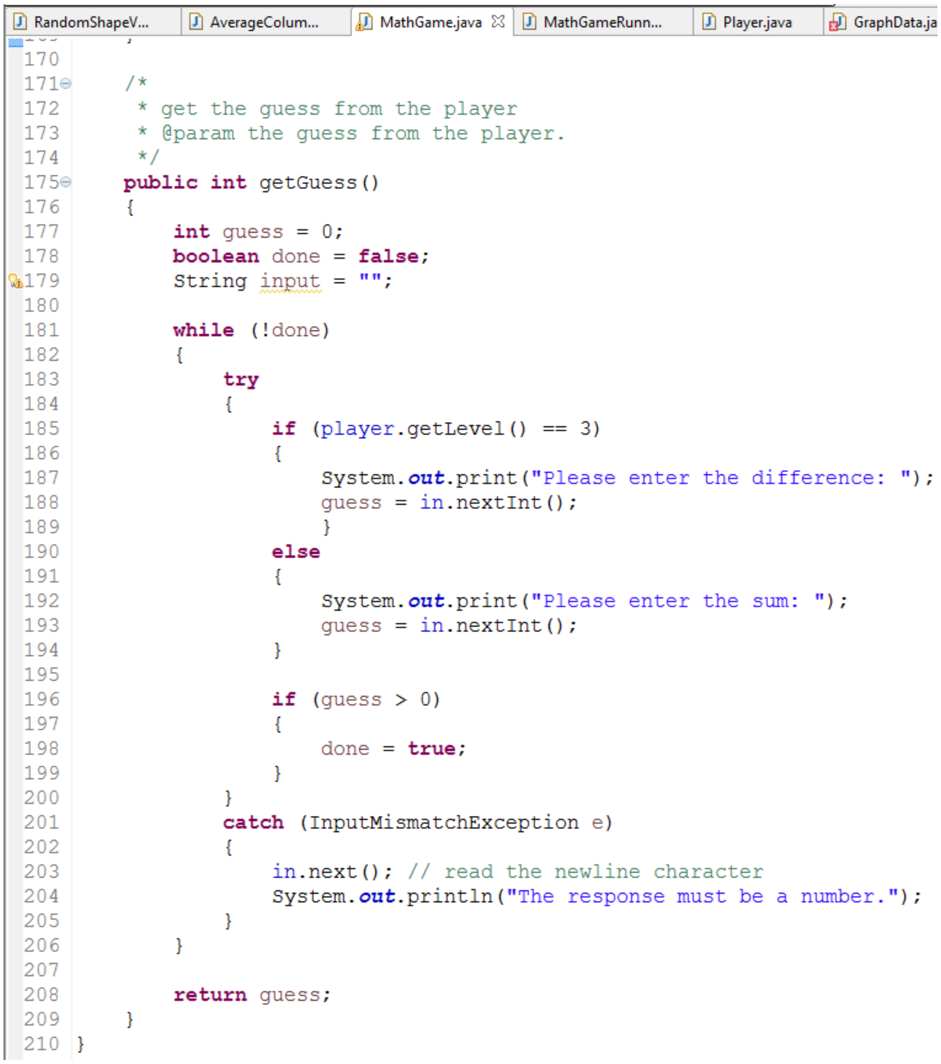
```
    // read user information
    public void readPlayerInformation()
    {
        String name = "";
        System.out.print("What is your name? ");
        name = in.next();
        int initialLevel = 0;
        boolean done = false;

        while (!done)
        {
            System.out.print("At what level do you want to start? (1-3) ");

            try
            {
                initialLevel = in.nextInt();

                if (initialLevel >= 1 && initialLevel <= 3)
                {
                    done = true;
                }
            }
            catch (InputMismatchException e)
            {
                in.next();  // read the newline character
                System.out.println("Plese, enter a number between a 1 and 3.");
            }
        }

        player = new Player(name, initialLevel);
    }

```



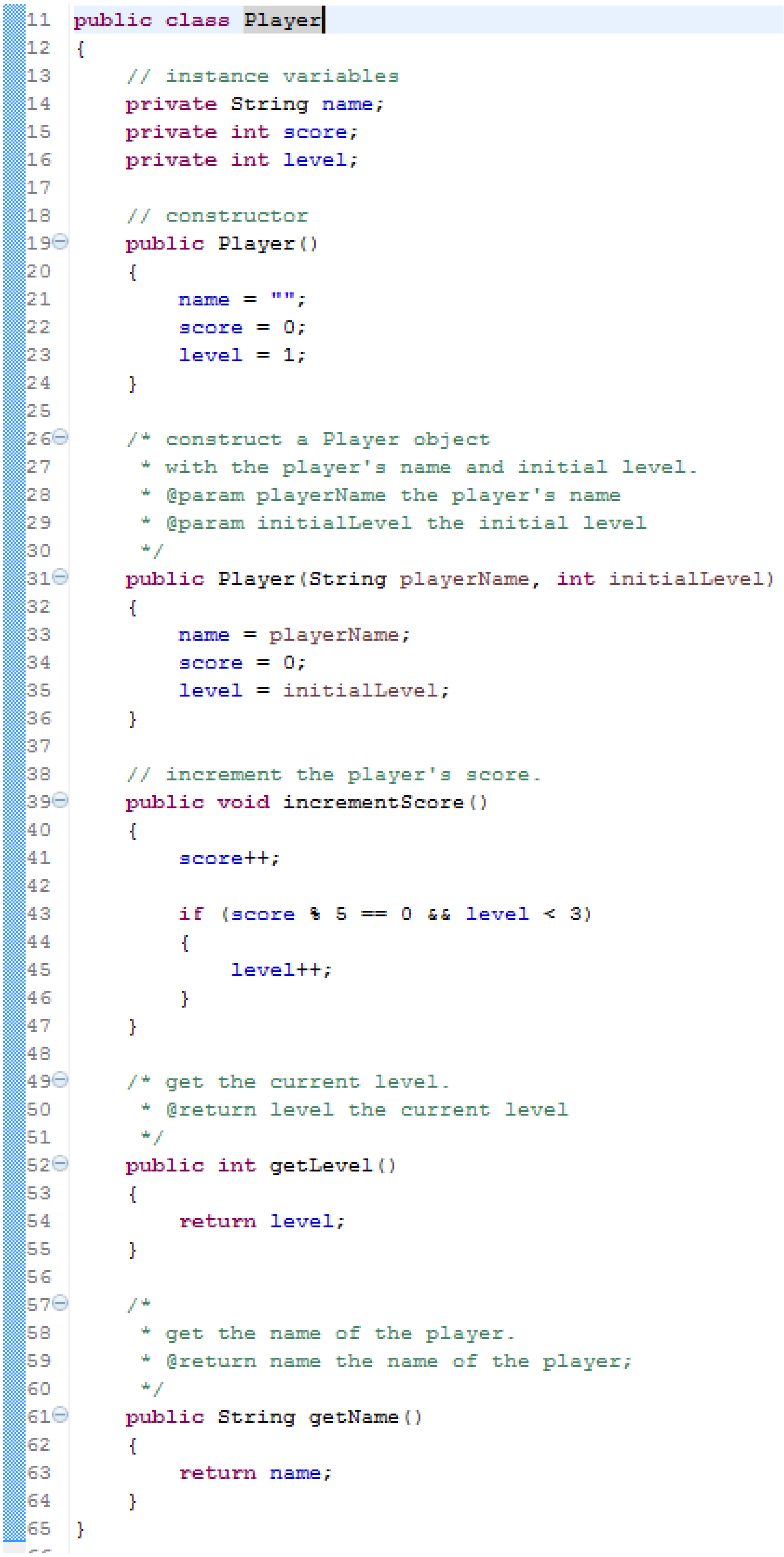
```
    // play one round of the game
    public void playRound()
    {
        int i1 = 0;
        int i2 = 0;
        int answer = 0;
        Random generator = new Random();

        if (player.getLevel() == 1)
        {
            boolean done = false;

            while (!done)
            {
                i1 = generator.nextInt(9) + 1;
                i2 = generator.nextInt(9) + 1;
                answer = i1 + i2;

                if (answer < 9)
                {
                    done = true;
                }
            }

            System.out.print("What is " + i1 + " + " + i2 + " ? ");
        }
        else if (player.getLevel() == 2)
        {
            i1 = generator.nextInt(9) + 1;
            i2 = generator.nextInt(9) + 1;
            answer = i1 + i2;
            System.out.print("What is " + i1 + " + " + i2 + " ? ");
        }

        if (player.getLevel() == 3)
        {
            boolean done = false;

            while (!done)
            {
                i1 = generator.nextInt(9) + 1;
                i2 = generator.nextInt(9) + 1;
                answer = i2 - i1;

                if (answer > 0)
                {
                    done = true;
                }
            }

            System.out.print("What is " + i2 + " - " + i1 + " ? ");
        }

        System.out.println(getResults(answer));
    }
```



```
    /*
     * get results of the play.
     * @param answer the answer
     * @return the results
     */
    public String getResults(int answer)
    {
        int guess = getGuess();

        if (answer != guess) //Allow a second try.
        {
            System.out .println("Sorry, that is incorrect. Please try one more time.");
            guess = getGuess();
        }

        String result = "";

        if (answer == guess)
        {
            result = "Congratulations, " + player.getName() + "! That is correct.";
            player.incrementScore();
        }
        else
        {
            result = "Sorry, " + player.getName() +
                    ". The correct answer is " + answer;
        }

        return result;
    }
```



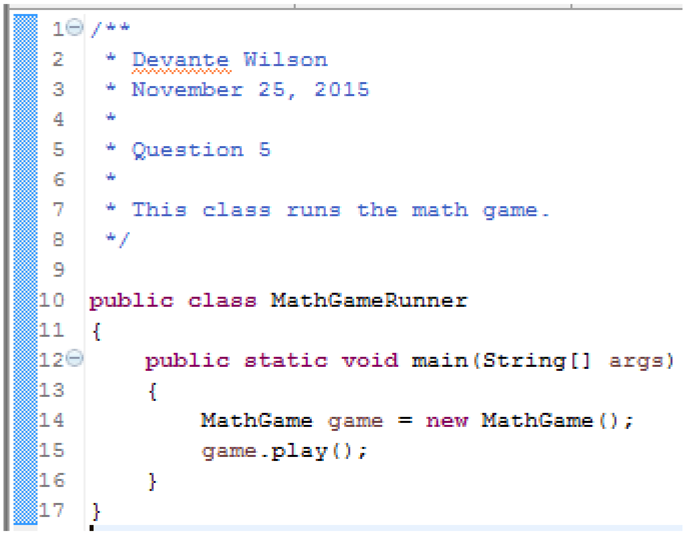
```
    }

    /*
     * get the guess from the player
     * @param the guess from the player.
     */
    public int getGuess()
    {
        int guess = 0;
        boolean done = false;
        String input = "";

        while (!done)
        {
            try
            {
                if (player.getLevel() == 3)
                {
                    System.out.print("Please enter the difference: ");
                    guess = in.nextInt();
                    }
                else
                {
                    System.out.print("Please enter the sum: ");
                    guess = in.nextInt();
                }

                if (guess > 0)
                {
                    done = true;
                }
            }
            catch (InputMismatchException e)
            {
                in.next(); // read the newline character
                System.out.println("The response must be a number.");
            }
        }

        return guess;
    }
}
```



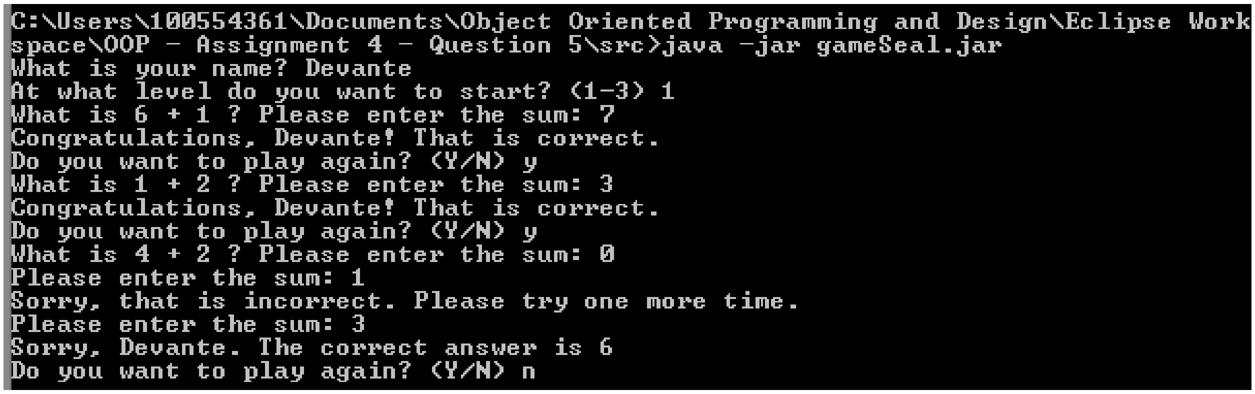
```java
public class Player
{
    // instance variables
    private String name;
    private int score;
    private int level;

    // constructor
    public Player()
    {
        name = "";
        score = 0;
        level = 1;
    }

    /* construct a Player object
     * with the player's name and initial level.
     * @param playerName the player's name
     * @param initialLevel the initial level
     */
    public Player(String playerName, int initialLevel)
    {
        name = playerName;
        score = 0;
        level = initialLevel;
    }

    // increment the player's score.
    public void incrementScore()
    {
        score++;

        if (score % 5 == 0 && level < 3)
        {
            level++;
        }
    }

    /* get the current level.
     * @return level the current level
     */
    public int getLevel()
    {
        return level;
    }

    /*
     * get the name of the player.
     * @return name the name of the player;
     */
    public String getName()
    {
        return name;
    }
}
```

Devante Wilson
Group 7

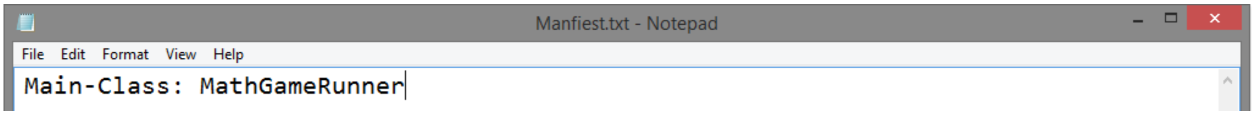
Assignment 4: Object Oriented Design



```java
/**
 * Devante Wilson
 * November 25, 2015
 *
 * Question 5
 *
 * This class runs the math game.
 */

public class MathGameRunner
{
    public static void main(String[] args)
    {
        MathGame game = new MathGame();
        game.play();
    }
}
```



```
C:\Users\100554361\Documents\Object Oriented Programming and Design\Eclipse Work
space\OOP - Assignment 4 - Question 5\src>java -jar gameSeal.jar
What is your name? Devante
At what level do you want to start? (1-3) 1
What is 6 + 1 ? Please enter the sum: 7
Congratulations, Devante! That is correct.
Do you want to play again? (Y/N) y
What is 1 + 2 ? Please enter the sum: 3
Congratulations, Devante! That is correct.
Do you want to play again? (Y/N) y
What is 4 + 2 ? Please enter the sum: 0
Please enter the sum: 1
Sorry, that is incorrect. Please try one more time.
Please enter the sum: 3
Sorry, Devante. The correct answer is 6
Do you want to play again? (Y/N) n
```



```
Main-Class: MathGameRunner
```