README\_file Chen Chen CS304 Technical Writing

**Project Summary**

The program was designed to execute with its assigned argument from the designer, without the interaction with the users. Also, the program was a simple practice for Java program beginners to learn the idea of creating different classes and later access through these classes by creating multiple object with different values or arguments values. However, the program was assigned to complete two different objects from each shape classes and display their areas separately at the output window. It was an individual assignment that could complete rarely easy to the Java beginners.

**Project Tool(s)**

1. The project requires NetBeans 8.2 compilers since it can create an excellent NetBeans Project folder to the file location, and it can easily build and compile and execute the designer’s code. Furthermore, NetBeans has its own application that could correcting the syntax errors or notify the designer to implement valid method to adjust the program codes.

**Classes of the Program:**

1. Circle.java

2. Rectangle.java

3. Triangle.java

4. Lab5A.java

**Class Construction**

**Class: Circle**

The class was designated as components holders specifically for the Shape of the circle, which it was involving the main component radius to determine the area of the circle.

**Java imports: N/A**

**Attribute**

|  |  |
| --- | --- |
| Private double radius | Radius represents as double value |

**Constructor**

|  |  |
| --- | --- |
| Public Circle() | Assign the radius initially to 1, 0 is not valid value to determine the area of the circle |
| Public Circle(double newRadius) | Set the radius to the constant of 1 at the beginning, and it’s designed to accept the new value of the radius each time the programmer assigned |

**Methods**

|  |  |
| --- | --- |
| **Methods** | **Description or returns** |
| setRadius(double radius) | Set the valid radius to the valid value of 1 and return whatever is the value radius to the parameter |
| getRadius() | Return the radius |
| Area() | Return the standard formula to calculate the correct area based on the value of the radius |
| toString() | Return the final result of the circle formula based on the method above |

**Class: Rectangle**

The class was creating to accept two parameters, which were length and height, to calculate or execute the final area at the end of the output.

**Java Imports: N/A**

**Attribute**

|  |  |
| --- | --- |
| Private double length | Length represents as double |
| private double height | Height represents as double |

**Constructor**

|  |  |
| --- | --- |
| Rectangle() | Assign the length value to 1, the height value to 1 since 0 will invalidate the output of the area |
| Rectangle(double newLength, double newHeight) | Set the length and height both to the initial value of 1, and it also set to accept the new value of length and height |

**Methods**

|  |  |
| --- | --- |
| **Methods** | **Description or returns** |
| getLength() | Return length |
| getHeight() | Return height |
| setLength(double length) | Set the length to greater or equal to 1 as a valid length and assign the new length to the area |
| setHeight(double height) | Set the height greater or equal to the valid height and assign the valid height to the area formula |
| Area() | Return the standard formula for the area of the rectangle |
| toString() | Return the final area of the rectangle and display the output to the output window |

**Class: Triangle**

The class was created specifically to calculate the area of the triangle, which it accepts two components as parameters.

**Java Imports: N/A**

**Attribute**

|  |  |
| --- | --- |
| Private double base | base represents as double |
| Private double height | Height of the triangle represents as double |

**Constructors**

|  |  |
| --- | --- |
| Triangle() | Assign the corresponding value of the base and height to valid value of 1 |
| Triangle(double newBase, double newHeight) | Set the valid value of base and height, and accept the new value of base and height |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description or returns** |
| getBase() | return the base of the triangle |
| getHeight() | return the height of the triangle |
| setBase(double base) | set the valid value of base and assign it to the attribute |
| SetHeight(double height) | Set the valid value of height and assign it to the attribute |
| Area() | Return the standard formula of the triangle and executes it |
| toString() | Display the result of the triangle to the output window |

**Class: Lab5A**

The class was the main class to the entire project, and it could create multiple objects through the shape classes that we implemented to the project. It would execute the area of each created Shape Object and display the total area by the given parameters.

**Java Imports: N/A**

**Attribute**

|  |  |
| --- | --- |
| Circle c# = new Circle() | It can create brand new objects to the corresponding Circle class (# was symbolizing the multiple object) |
| Rectangle r# = new Rectangle() | It can create brand new objects to the corresponding Rectangle Class (# was symbolizing the multiple object) |
| Triangle t# = new Triangle() | It can create brand new objects to the corresponding  Triangle class (# was symbolizing the multiple object) |
| double total | Total (area) represents as double |

**Methods**

|  |  |
| --- | --- |
| Total += C#.area() | It calculates the total area between multiple Circle objects |
| Total += R#.area() | It calculates the total area between multiple rectangle objects |
| Total += T#.area() | It calculates the total area between multiple triangle objects |
| System.out.println() | It displays the final result when the formula of totals was completed |