Contents

[**Lessons Learned** 1](#_Toc36393052)

[**UML Diagram** 1](#_Toc36393053)

[**User Guide** 1](#_Toc36393054)

[**Test Cases** 2](#_Toc36393055)

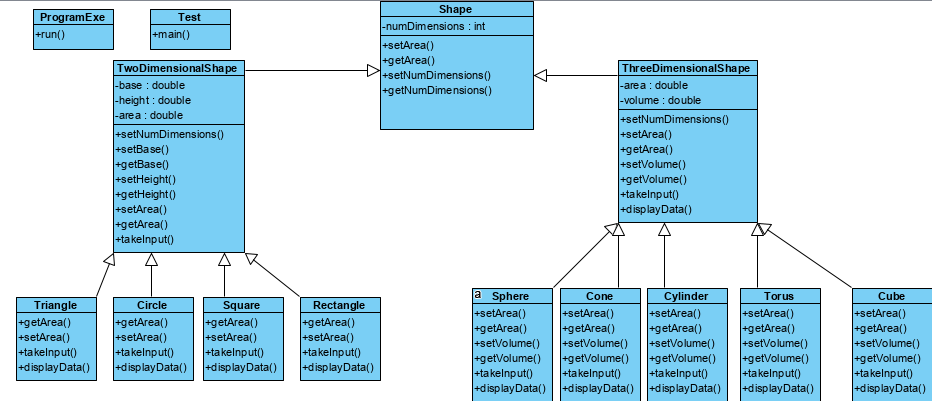
[**Additional Notes** 4](#_Toc36393056)

[Changelog 5](#_Toc36393057)

**Lessons Learned**

**Working on this program showed me the importance of using inheritance to share certain states and behaviors across Parent/Child classes. I’ve created some programs in past classes that could have been coded ‘cleaner’ by using some of the practices such as overriding methods, etc. I’m sure there are some redundancies and inefficiencies, some of which I discuss in the “additional notes” section, that can be improved on in my code and I will be amending the project for my own benefit moving forward.**

# **UML Diagram**



**User Guide**

1. **The program will run and prompt you to make a selection (figure 1).**

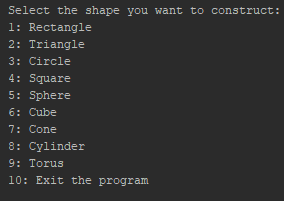


Figure 1

1. **If you choose a shape, you will be prompted to enter the parameters based upon its formula for calculating area and/or volume (figure 2).**
2. **On entering the values, the program will tell you the area and/or volume of your shape (figure 2).**

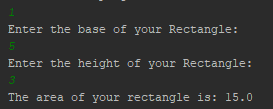


Figure 2

1. **You will be prompted if you would like to quit the program. ‘Y’ or ‘y’ will allow you to quit and prompt you on the current date and time (figure 3).**

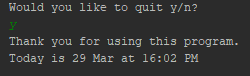
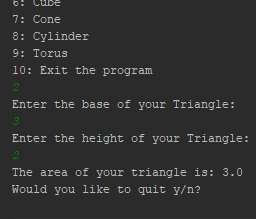


Figure 3

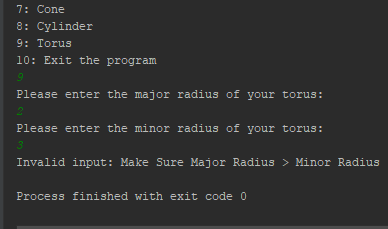
**Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case** | **Input** | **Expected Output** | **Actual Output** | **Success?** |
| **1** | **Option = 2, base =3, height = 2** | **The area of your triangle is: 3.0** | **The area of your triangle is: 3.0** | **Y** |
| **2** | **Option = 9, major radius = 2, minor radius = 3** | **Invalid input: Make sure Major Radius > Minor Radius** | **Invalid input: Make sure Major Radius > Minor Radius** | **Y** |
| **3** | **Option = ‘f’** | **Not a valid choice. Please choose a valid option.**  **Would you like to quit y/n?** | **Not a valid choice. Please choose a valid option. Would you like to quit y/n?** | **Y** |
| **4** | **Would you like to quit y/n? = ‘x’** | **Please select either y or n.**  **Would you like to quit y/n?** | **Program closes.** | **N** |

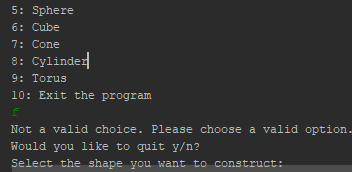
**Test Case 1:**



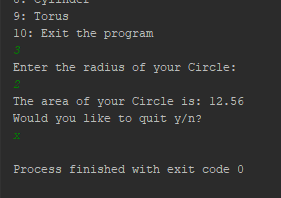
**Test Case 2:**



**Test Case 3:**



**Test Case 4:**



**Additional Notes**

**A note of some changes I will be making to improve the project moving forward.**

**Efficiency Changes:**

**Place try/catch blocks in takeInput method for values instead of switch statements.(??)**

**As of now, methods such as getBase/getHeight aren't used, either change or remove.**

**Change/Separate takeInput method, need a void method to calculate, then another method to specifically just set the return calculated amount.**

**Change takeInput methods in parent classes to not require parameters i.e.(double b, double h).**

**Incongruency in 'ThreeDimensionalShape' having displayData method but 'TwoDimensionalShape' does not.**

**Not working as intended:**

**Prompts the user to quit again even if they select option '10' because that logic is outside of the switch statements.**

**Only recognizes to quit the program if 'y' is entered, does not properly differentiate between 'n' and any other input for continuing.**

**If the user inputs the wrong type of data for a shape in its menu, then they will continue to get the catch block of "Not a valid Choice. Please choose a valid option" until they restart the program.**

**User should not have to exit the program if they enter incompatible values in the ‘Torus’ option. It should allow them to try again.**

**Not sure of:**

**Do the open scanners in my shape classes continue to drain resources or is closing the scanner in ProgramExe sufficient?**

**Not sure where ProgramExe class and Test class go on my UML chart.**

**Changelog**

**(3/29/2020)**

* **Added**

while ((quit != 'y') && (quit == 'n'))

**, which causes other input aside from ‘y’ and ‘n’ to exit the program. Need to change so that it prompts user to try a valid option.**

* **Added**
* /\* if(quit != 'y' && quit != 'n'){  
   System.out.println("Please select either y or n");  
  }\*/

**Prompts the user again but needs to loop back to initial question of “Would you like to quit y/n?” Commented out until further testing is done.**