Table of Contents

[Introduction 1](#_Toc72771033)

[Software Engineering 1](#_Toc72771034)

[Version control 1](#_Toc72771035)

[GitHub 2](#_Toc72771036)

[My Pages and Shapes in Single Line Program 2](#_Toc72771037)

[Main Page 3](#_Toc72771038)

[MOVETO Command 4](#_Toc72771039)

[Draw Circle 4](#_Toc72771040)

[Draw Rectangle 5](#_Toc72771041)

[Draw Triangle 5](#_Toc72771042)

[Testing 6](#_Toc72771043)

[1. White box testing: 6](#_Toc72771044)

[2. Black Box Testing: 6](#_Toc72771045)

[Testing Levels: 6](#_Toc72771046)

[Integration Testing: 6](#_Toc72771047)

[System Testing: 6](#_Toc72771048)

[Bibliography 15](#_Toc72771049)

# Documentation

## Introduction

It is difficult to draw the shapes using programming language but if we use our own code or language, it is so much easier to do this.

Here I’ve design and developed the software application which can draw Shapes using user command. This is documentation, you can see my software application which helps to design different types of shapes.

## Software Engineering

Software engineering write the concept of program’s components/objects. In software engineering, team assembles to complete the program. And programming is in the large scale. Software engineering is built by teams where programming is in the small scale and it is done by individual or small group. Software engineering last many year. IEEE in its standard 610.12-1990, said that “engineering as the application of a systematic, disciplined, which is a computable approach for the development, operation, and maintenance of software” (IEEE, 2020). There are many roles of software engineering which are as following:

* Constructing large software application.
* Defining problem clear and completely
* Tools and techniques to support process
* Team oriented Experience

## Version control

In my definition, version control is the system which control the version of application of source code. Its help to record our source code by changes. Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later. It is the practice of tracking and managing changes to software code. There are many version control systems and some are as following:

* Git
* Helix VCS
* Microsoft Team Foundation Server
* Subversion
* Bitbucket
* GitHub

Among this popular version control system, I am going to used Bitbucket version control system for my project. And also going to briefly explain about it.

### GitHub

I am using GitHub as my version control system. It is easier to use when I need to commit and it is also popular than any other version control system. It is like social media of programmer when needed.

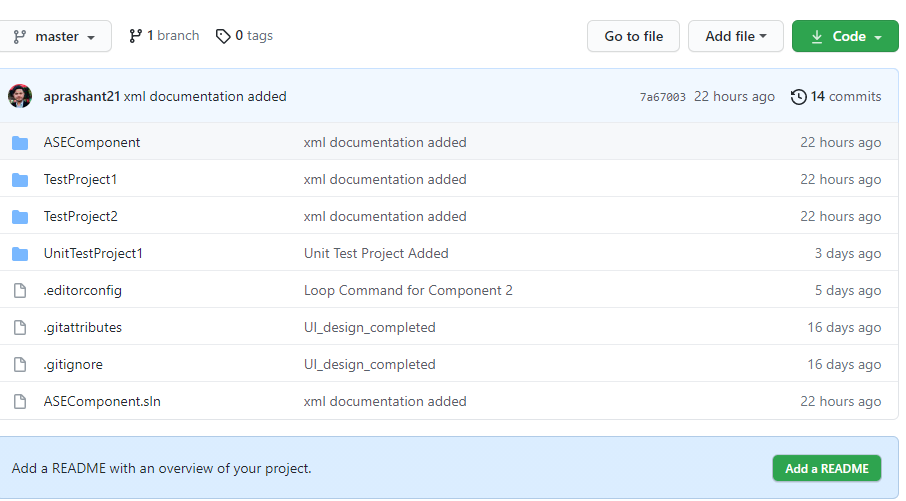
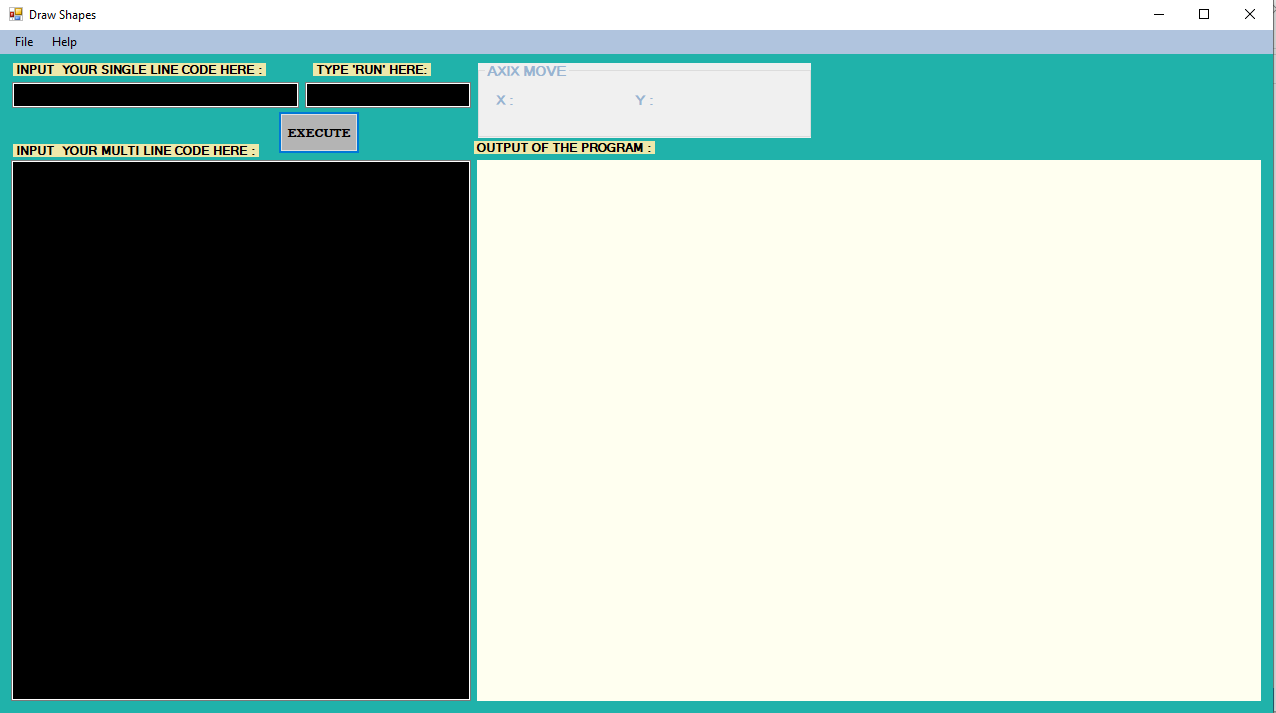


Fig 1: My Version control system and commit

## My Pages and Shapes in Single Line Program

Here are some of my design and commands which helps to design shape in single line.

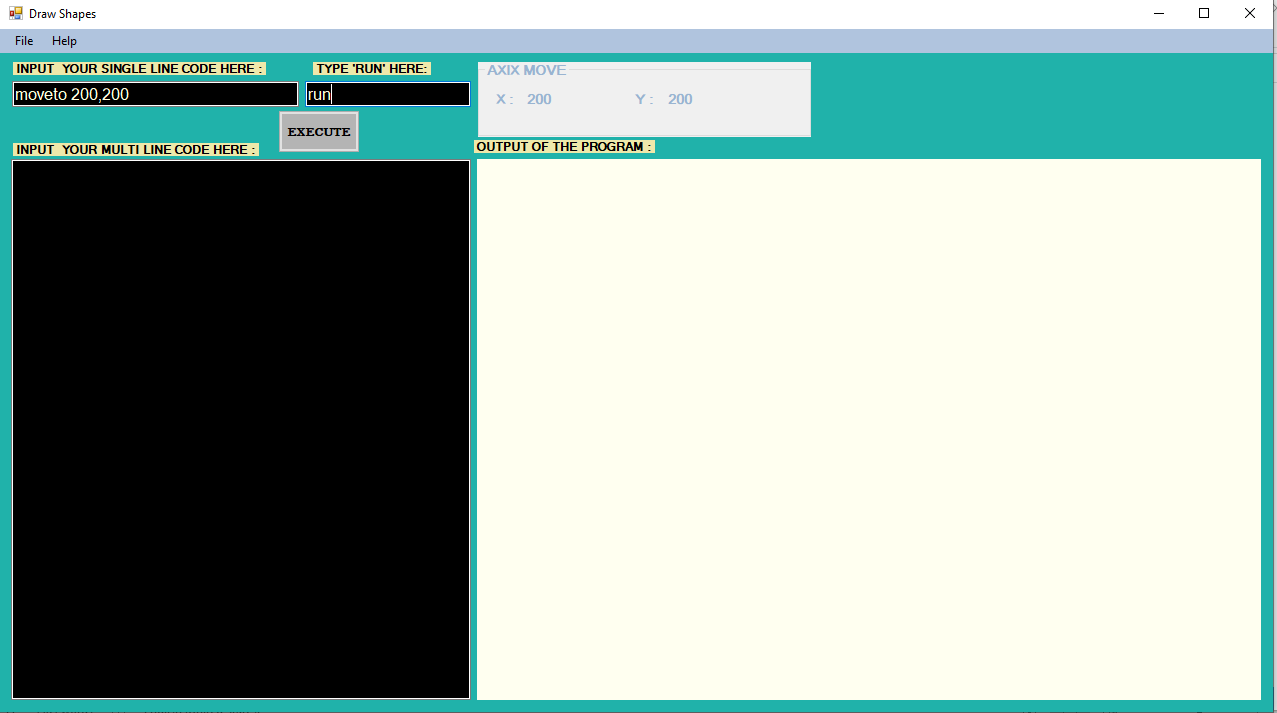
### Main Page



*Figure 1: Main page*

The above figure is the figure of my main page of “Draw Shapes” application, where shape will be created at the right side of white box, and there is single line command line where user can enter single line command and type “run” and the text box to execute the command and draw shapes and there is also multiline command line where user can write a program and click on the “EXECUTE” button to run the program. There is group box named “AXIX MOVE” which contain x and y axis.

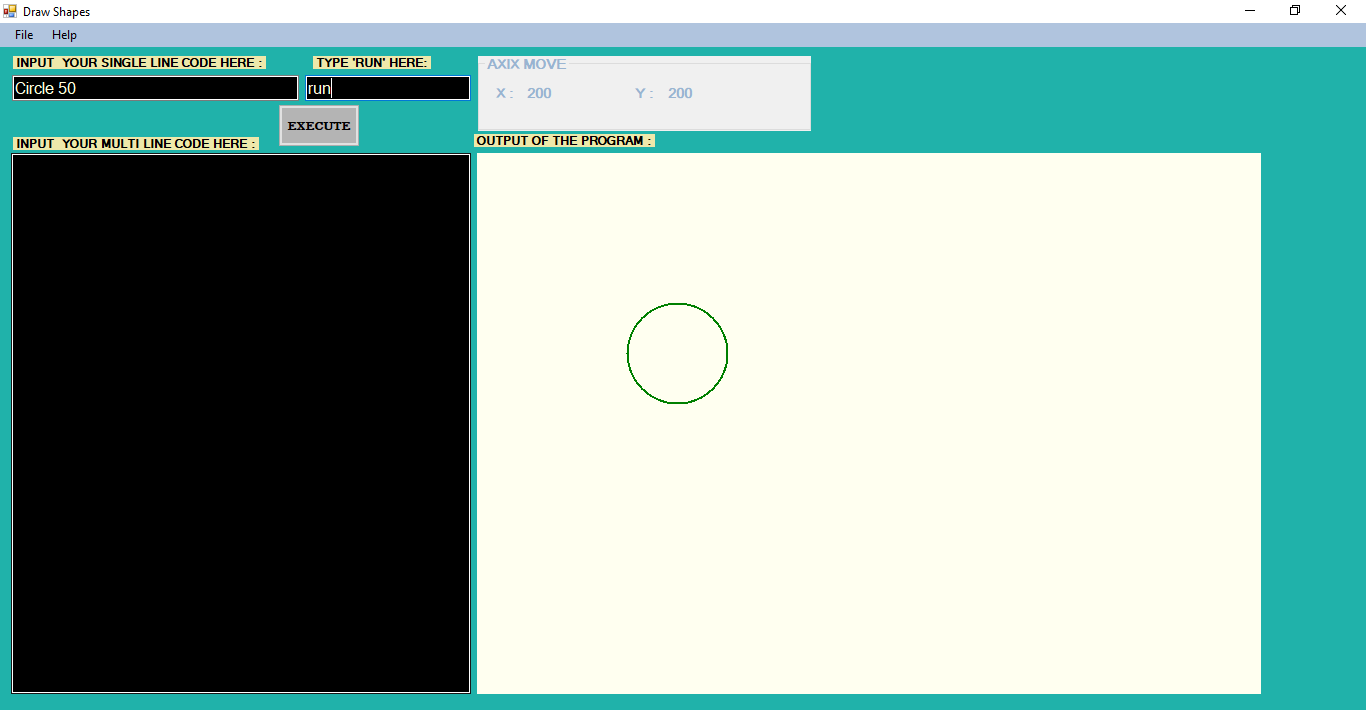
### MOVETO Command



*Figure 3: using moveto command*

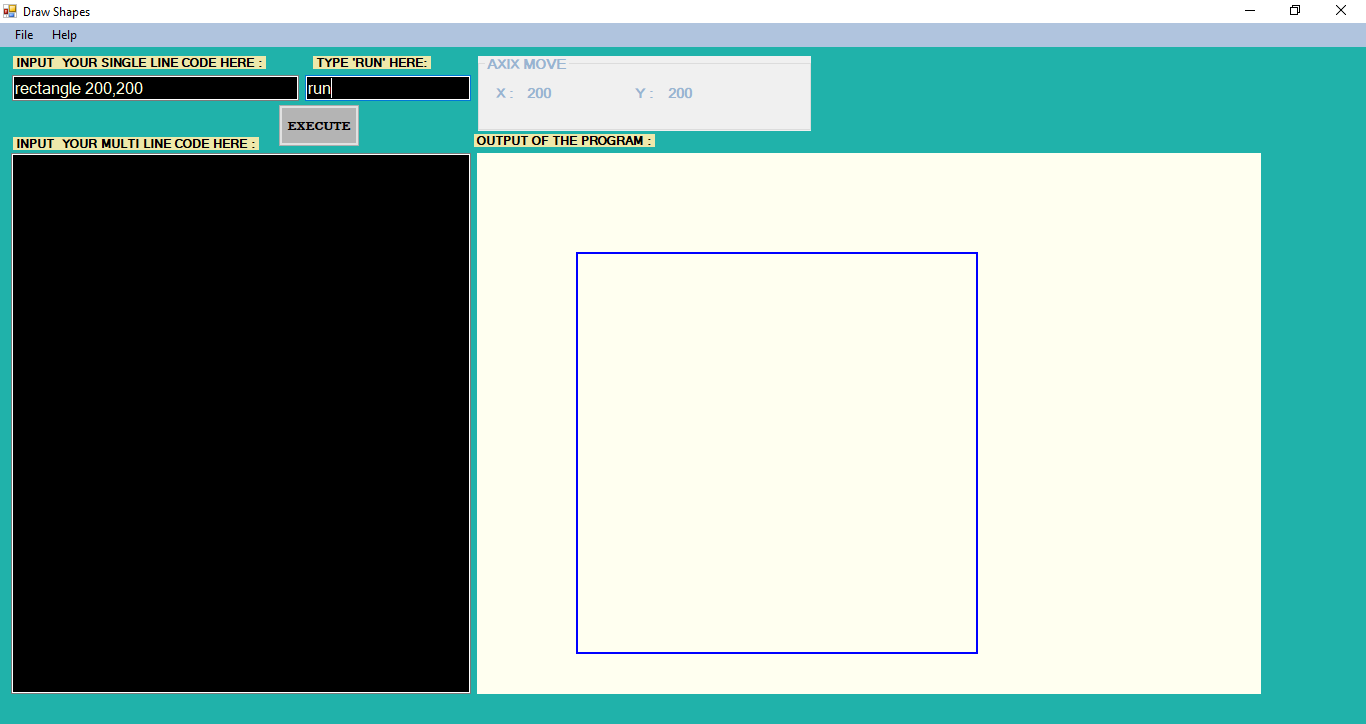
With the help of Moveto command, we can move the co-ordinate of x and y axis in the output panel.

### Draw Circle

**

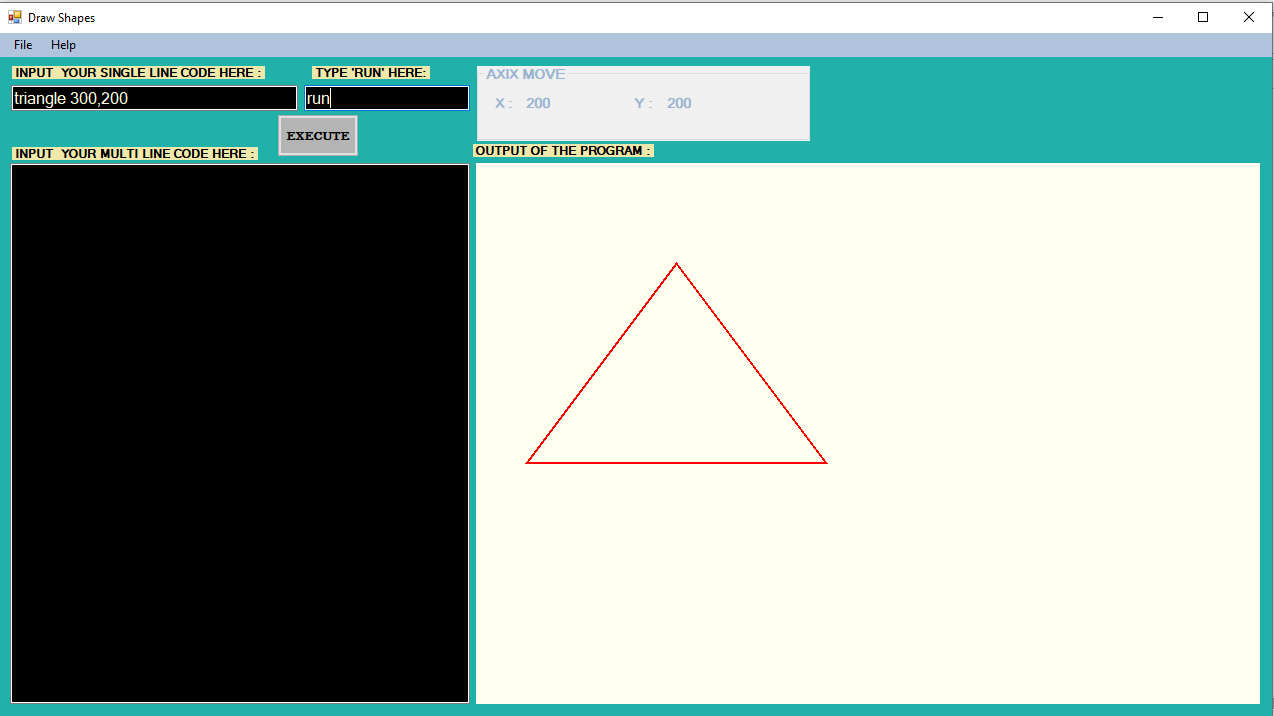
*Figure 4: using drawcircle command*

### Draw Rectangle



*Figure 5: using draw rectangle command*

### Draw Triangle



*Figure 6: using draw tringle command*

## Testing

Testing is the process to test the application. Unit testing is the type of software testing where individual unit of the software are tested (Anon., 2021). There are different type of testing which are:

1. White box testing:

It is conducted to test program and its implementation, in order to improve code efficiency or structure. It is also known as ‘Structural’ testing. In this testing method, the design and structure of the code are known to the tester. Programmers of the code conduct this test on the code. This testing is usually done at the unit level.

1. Black Box Testing:

It is carried out to test functionality of the program. It is also called ‘Behavioral’ testing. The tester in this case, has a set of input values and respective desired results. On providing input, if the output matches with the desired results, the program is tested ‘ok’, and problematic otherwise. In this testing method, the design and structure of the code are not known to the tester, and testing engineers and end users conduct this test on the software.

## Testing Levels:

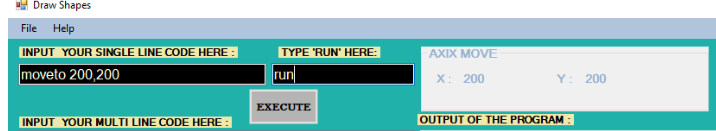
### Integration Testing:

Integration Testing is the process of testing the connectivity or data transfer between a couple of units tested modules. It is AKA I&T Testing or String Testing. It is subdivided into Top-Down Approach, Bottom-Up Approach and Sandwich Approach (Combination of Top Down and Bottom Up).

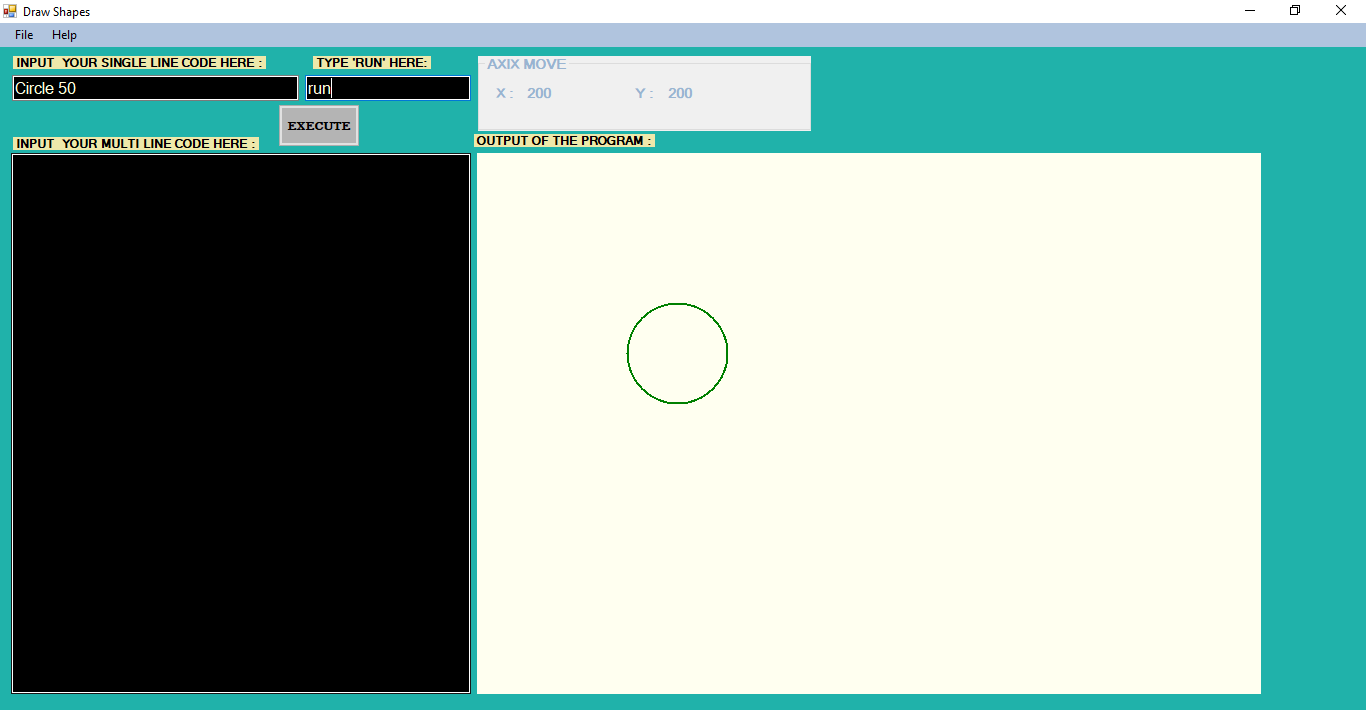
### System Testing:

It’s a black box testing. Testing the fully integrated application this is also called as end to end scenario testing. To ensure that the software works in all intended target systems. Verify thorough testing of every input in the application to check for desired outputs. Testing of the user’s experiences with the application.

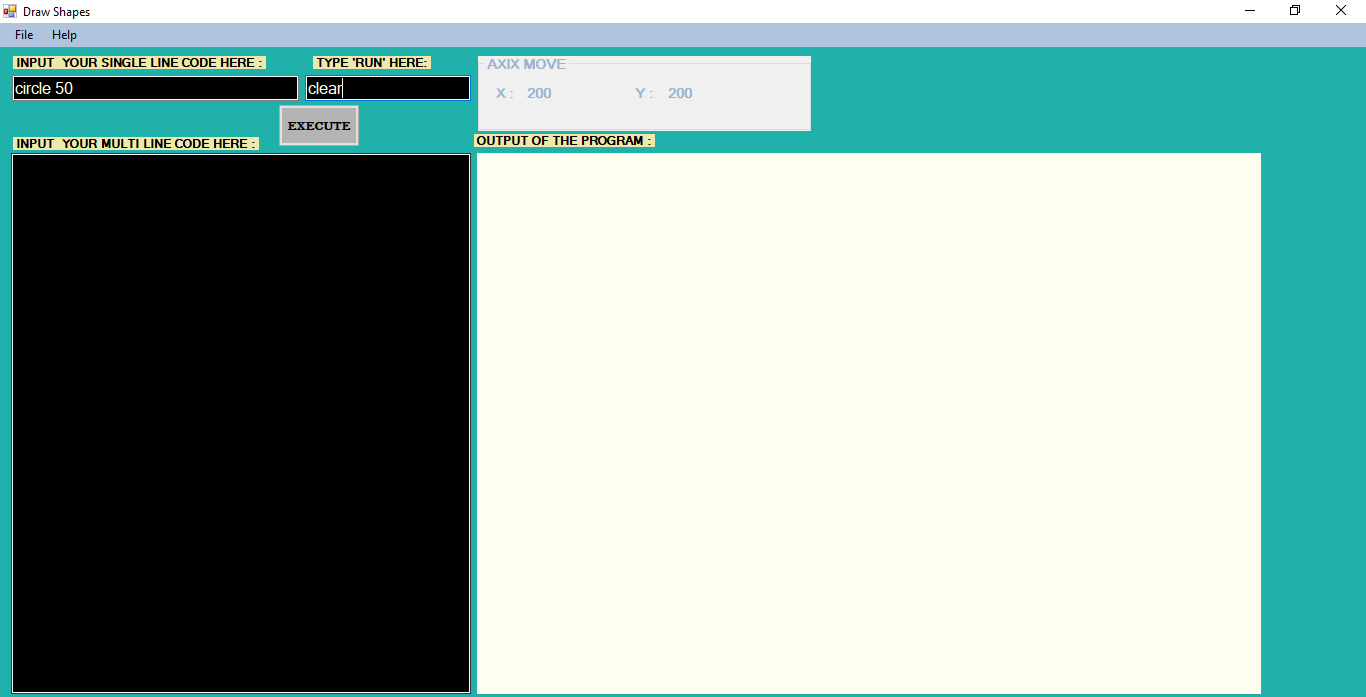
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | | | | 01 | **Test Case Description** | | | Test the Functionality in | | | | | | |
| **Created By** | | | | Prashant  Adhikari | **Reviewed By** | | | The British College | | **Version** | | | 1.0.0 | |
|  | |  | |  |  | |  |  |  |  |  | |  |  |
| **QA Tester’s Log** | | | | Review comments from British college | | | | |  |  |  | |  |  |
|  | |  | |  |  | |  |  |  |  |  | |  |  |
| **Tester’s Name** | | | | Prashant  Adhikari | **Date Tested** | | | 10-may-2021 | | **Test Case (Pass/Fail/NOT)** | | | Pass | |
|  | |  | |  |  | |  |  |  |  |  | |  |  |
| **S#** | | **Prerequisites** | | | | |  | **S#** | | **Test Data** | | | | |
|  | |  | | | | |  | 1. | | moveto 200.200 | | | | |
|  | |  | | | | |  |  | |  | | | | |
|  | |  | | | | |  |  | |  | | | | |
| **Test Scenario** | | Verifying on entering valid command, the user can change drawing position. | | | | | | | |  |  | |  |  |
|  | |  | |  |  | |  |  |  |  |  | |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | | **Actual Results** | | | | | | **Pass/Fail/Not executed/suspended** | | |
| **1** | Open Shape Drawing application | | Application should open | | | As Expected, | | | | | | Pass | | |
| **2** | Enter “moveto” and two variables separated using comma | | Text can be entered | | | As Expected, | | | | | | Pass | | |
| **3** | Type “run” command in the text box | | AXIX MOVE group has changed its value | | | As expected, | | | | | | Pass | | |



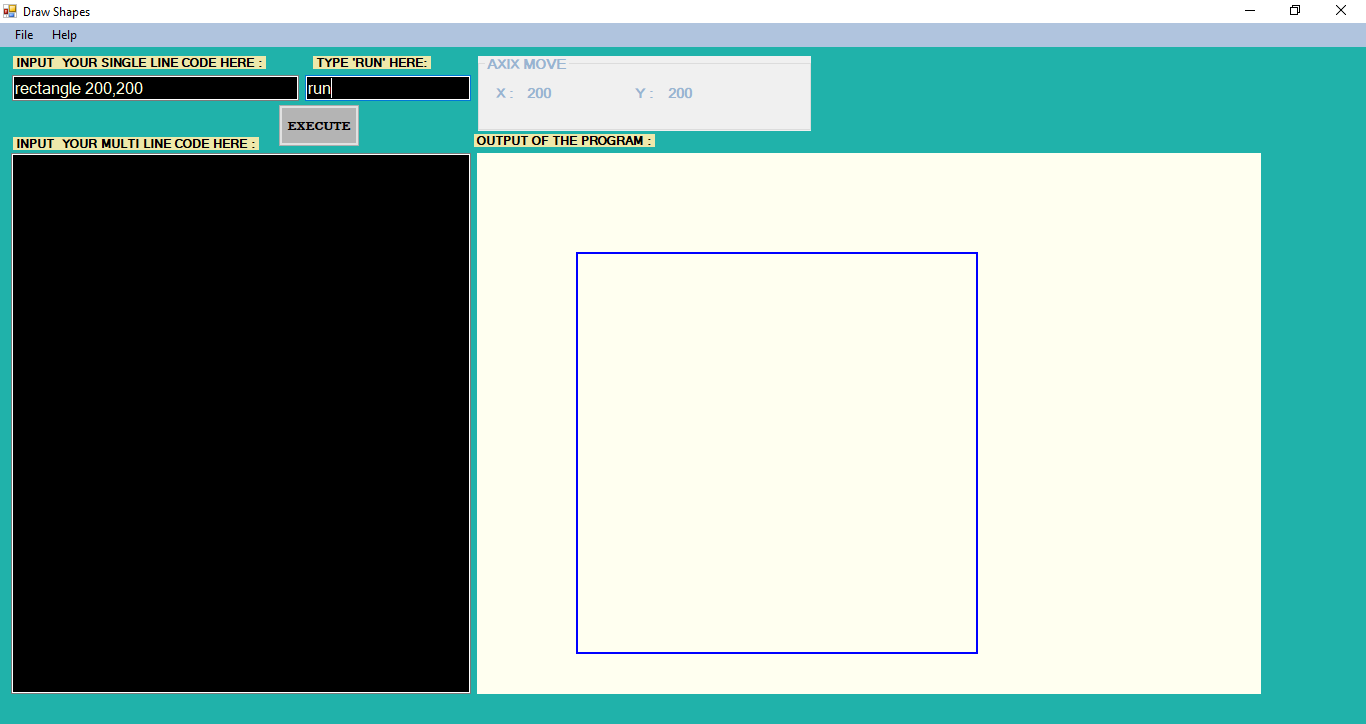
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | | | 02 | | **Test Case Description** | | | Test the Functionality in Shape Drawing application | | | | | | |
| **Created By** | | | Prashant  Adhikari | | **Reviewed By** | | | British College | | **Version** | | | 1.0.0 | |
|  | |  |  | |  | |  |  |  |  |  | |  |  |
| **QA Tester’s Log** | | | Review comments from British college | | | | | |  |  |  | |  |  |
|  | |  |  | |  | |  |  |  |  |  | |  |  |
| **Tester’s Name** | | | Prashant  Adhikari | | **Date Tested** | | | 12-May-2021 | | **Test Case (Pass/Fail/NOT)** | | | Pass | |
|  | |  |  | |  | |  |  |  |  |  | |  |  |
| **S#** | | **Prerequisites** | | | | |  | **S#** | | **Test Data** | | | | |
|  | |  | | | | |  | 1. | | Circle 150 | | | | |
|  | |  | | | | |  |  | |  | | | | |
|  | |  | | | | |  |  | |  | | | | |
| **Test Scenario** | | Verifying the Command and draw circle | | | | | | | |  |  | |  |  |
|  | |  |  | |  | |  |  |  |  |  | |  |  |
| **Step #** | **Step Details** | | | **Expected Results** | | **Actual Results** | | | | | | **Pass/Fail/Not executed/suspended** | | |
| **1** | Open Shape Drawing application | | | Application should open | | As Expected, | | | | | | Pass | | |
| **2** | Enter circle and radius’s value | | | Text can be entered | | As expected, | | | | | | Pass | | |
| **3** | Type “run” command in the textbox | | | Circle will be drawn in panel | | As expected, | | | | | | Pass | | |



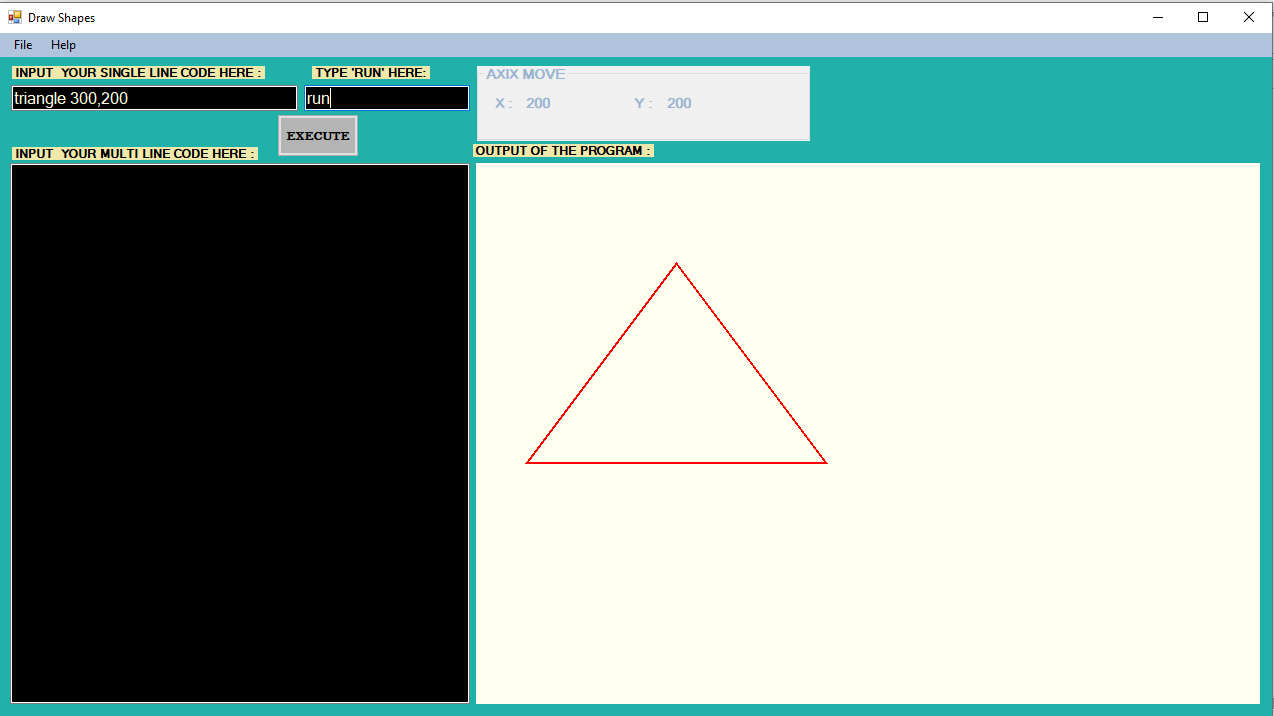
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | | | | 03 | | **Test Case Description** | | Test the Functionality in Shape Drawing application | | | | | | |
| **Created By** | | | | Prashant | | **Reviewed By** | | British College | | **Version** | | | 1.0.0 | |
|  | |  | |  | |  |  |  |  |  |  | |  |  |
| **QA Tester’s Log** | | | | Review comments from British college | | | | |  |  |  | |  |  |
|  | |  | |  | |  |  |  |  |  |  | |  |  |
| **Tester’s Name** | | | | Prashant | | **Date Tested** | | 13-may-2021 | | **Test Case (Pass/Fail/NOT)** | | | Pass | |
|  | |  | |  | |  |  |  |  |  |  | |  |  |
| **S#** | | **Prerequisites** | | | | |  | **S#** | | **Test Data** | | | | |
|  | |  | | | | |  | 1. | | Clear | | | | |
|  | |  | | | | |  |  | |  | | | | |
|  | |  | | | | |  |  | |  | | | | |
| **Test Scenario** | | Verifying valid “clear” command | | | | | | | |  |  | |  |  |
|  | |  | |  | |  |  |  |  |  |  | |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | | | | | | **Pass/Fail/Not executed/suspends** | | |
| **1** | Open  Drawing Shapes application | | Application should open | | As Expected, | | | | | | | Pass | | |
| **2** | Enter circle and radius value | | Text can be entered | | As Expected, | | | | | | | Pass | | |
| **3** | Type run in execute textbox | | Drawing position has changed | | As expected, | | | | | | | Pass | | |
| 4 | Write clear in the text box | | Panel has been cleared | | As expected, | | | | | | | Pass | | |



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | | | 04 | | **Test Case Description** | | | Test the Functionality in Shape Drawing application | | | | | | |
| **Created By** | | | Prashant | | **Reviewed By** | | | British College | | **Version** | | | 1.0.0 | |
|  | |  |  | |  | |  |  |  |  |  | |  |  |
| **QA Tester’s Log** | | | Review comments from British college | | | | | |  |  |  | |  |  |
|  | |  |  | |  | |  |  |  |  |  | |  |  |
| **Tester’s Name** | | | Prashant | | **Date Tested** | | | 15-may-2021 | | **Test Case (Pass/Fail/NOT)** | | | Pass | |
|  | |  |  | |  | |  |  |  |  |  | |  |  |
| **S#** | | **Prerequisites** | | | | |  | **S#** | | **Test Data** | | | | |
|  | |  | | | | |  | 1. | | Rectangle 50,50 | | | | |
|  | |  | | | | |  |  | |  | | | | |
|  | |  | | | | |  |  | |  | | | | |
| **Test Scenario** | | Verifying on user can draw rectangle or not | | | | | | | |  |  | |  |  |
|  | |  |  | |  | |  |  |  |  |  | |  |  |
| **Step #** | **Step Details** | | | **Expected Results** | | **Actual Results** | | | | | | **Pass/Fail/Not executed/suspended** | | |
| **1** | Open Shape Drawing application | | | Application should open | | As Expected, | | | | | | Pass | | |
| **2** | Enter rectangle and two variables separated using comma | | | Text can be entered | | As Expected, | | | | | | Pass | | |
| **3** | Type “run” command in the textbox | | | User can draw rectangle | | As expected, | | | | | | Pass | | |



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | | | | 05 | **Test Case Description** | | | Test the Functionality in Shape Drawing application | | | | | | |
| **Created By** | | | | Prashant | **Reviewed By** | | | British College | | **Version** | | | 1.0.0 | |
|  | |  | |  |  | |  |  |  |  |  | |  |  |
| **QA Tester’s Log** | | | | Review comments from British college | | | | |  |  |  | |  |  |
|  | |  | |  |  | |  |  |  |  |  | |  |  |
| **Tester’s Name** | | | | Prashant | **Date Tested** | | | 15-may-2021 | | **Test Case (Pass/Fail/NOT)** | | | Pass | |
|  | |  | |  |  | |  |  |  |  |  | |  |  |
| **S#** | | **Prerequisites** | | | | |  | **S#** | | **Test Data** | | | | |
|  | |  | | | | |  | 1. | | Triangle 120,200 | | | | |
|  | |  | | | | |  |  | |  | | | | |
|  | |  | | | | |  |  | |  | | | | |
| **Test Scenario** | | Verifying can draw triangle or not | | | | | | | |  |  | |  |  |
|  | |  | |  |  | |  |  |  |  |  | |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | | **Actual Results** | | | | | | **Pass/Fail/Not executed/suspended** | | |
| **1** | Open Shape Drawing application | | Application should open | | | As Expected, | | | | | | Pass | | |
| **2** | Enter triangle and two variables separated by comma | | Text can be entered | | | As Expected, | | | | | | Pass | | |
| **3** | Type “run” command in the textbox | | User can draw triangle | | | As expected, | | | | | | Pass | | |



**Unit test1**

|  |
| --- |
| **Correct result** |
| Here, I tested different shapes like circle, triangle and rectangle in the unit test. |
|  |

## Conclusion

So, above are the design of my developed shape drawing application, in which user can draw many shapes like circle, triangle and rectangle using our own program.

# Bibliography

Anon., 2021. *unit-testing-guide.html.* [Online]   
Available at: https://www.guru99.com/unit-testing-guide.html

IEEE, 2020. *what-is-software-engineering.html.* [Online]   
Available at: https://www.guru99.com/what-is-software-engineering.html  
[Accessed 2020].

Rebus, 2020. *Version control.* [Online]   
Available at: https://press.rebus.community/programmingfundamentals/chapter/version-control/  
[Accessed 1 Dec 2020].