

Programmable Calculator Part III | Jaskirat, 170027 | User Stories and Test Cases

User Stories

1. As a user, I want the programmable calculator to support print statement so that I would be able to display my desired data on the console.
2. As a user, I want the programmable calculator to support input statement so that I would be able to make a generic code and enter my desired values during the runtime directly into the console.
3. As a user, I want the programmable calculator to support assignment '=' operator, so that I would be able to assign new values to variables during the runtime itself.
4. As a user, I want the programmable calculator to overwrite previous values assigned to variables through the direct use of assignment '=' operator, so that my time would be saved for doing the same thing manually.
5. As a user, I want to have user-friendly instructions and warnings in the debug console regarding the print and input statement, so that I am able to know what mistakes I have been doing on the basis of syntax.
6. As a developer, I want to follow the rules of OO programming and support the Model-View-Controller architecture as it will create a layer of independence that will enable me to transfer data more efficiently.

Test Cases

Test Case	Check Item	Description	Steps to Execute	Expected Result
TC01	Print Statement	Check whether the print statement is working correctly or not according to lexical analysis.	In the workspace, enter the code: <code>print "This is printing"</code>	Upon clicking on the run button, the console should have the text: "This is printing"
TC02	Input Statement	Check whether the input statement is working correctly or not according to lexical analysis.	In the workspace, enter the code: <code>input a</code> <code>print "The value of a is\n"</code> <code>print a</code>	Upon clicking on the run button, the console should ask the user for input. On entering any value (say 5), it should display like this on console: The value of a is 5

Programmable Calculator Part III | Jaskirat, 170027 | User Stories and Test Cases

TC03	Assignment Operator	Check whether the assignment operator is working correctly or not according to lexical analysis and use of parser-nodes.	In the workspace, enter the code: a=5 print a	This should appear on the console: 5
TC04	Overriding Variables	Check whether the variables are overridden directly through the execution of a program.	Load definitions from repository and then, enter the following code in the workspace: print "The value of a is" print a print "\nThe new value of a is" a=6 print a	Since the value stored in repository was a = 4, it should appear something like this in the console: The value of a is 4 The new value of a is 6
TC05	OO Programming and MVC Architecture	Check whether the MVC Architecture is implemented properly and object-orientation-programming standards are followed.		MVC Layers can be observed through the UML Class Diagram.