Ram Yadav

Professor- Johnnie Williams

Date: - May 17, 2016

First and Foremost, I wanted to make the project with the Login Page where, user must provide Password and Username to run the program. I wanted to create a Scientific Calculator with Login Page on it. However, I built the Login System, but I couldn’t put the Scientific Calculator inside it, so I decided to make two different projects.

So by saying that, I have created two different projects named Login System and Scientific Calculator.

**Login System:**

*Overview of the Program:*

In order to run this program, user must put the exact Username and Password that they have set as to Login page. If Username and Password doesn’t match then program will throw an error by displaying a dialogue box. Dialogue box will pop up and it would details “Username or Password is Invalid”. It has three different buttons namely Login, Reset, and Exit. After entered Username and Password, Once click on the Login button, if the Username and Password matched, it will go to the next page otherwise, show an error. Incase if one typed Username or Password wrong, they can delete it by clicking the reset button. And at the end, if one want exit, they can simply exit program by clicking Exit button.

*Overview of the Code:*

I have used JFrame to make this project. I have used textField and Password Fields Palette for the user name and Password respectively. For the Login, Reset, and Exit I have used button Palette. I have imported three different java libraries to run this program. I have used system.exit() to exit program. And I have used if/else statement to set the Username and Password as well as to run the program.

**Scientific Calculator:**

*Overview of the Program:*

I have used GUI(Graphics user Interface) to make this calculator. Being a scientific calculator, EasyCalc allows you to perform all the basic arithmetic operations, and several other complex mathematical operations. It also support a few basic statistical operations. Things that are missing from the calculator are quick access to common constants like pi, e, etc. However, it comes packed with operational features like arbitrary floating point arithmetic, multiple number systems (binary, octal, decimal and hexadecimal), logarithmic, trigonometric, and exponential functions, etc.

* Standard/Scientific modes

 EasyCalc starts in Standard mode which allows you to perform only the basic arithmetic operations (addition, subtraction, multiplication, division, modulus, reciprocal, and square roots), but one can switch to Scientific mode which allows more complex operations. In Scientific mode you can type expressions using functions like sin, cos, tan, log, ln (natural log), int (integral), factorial, cube root, exponentials, statistical operations (sum and average), etc. The calculator honors operator precedence in both Standard and Scientific modes.

* Operator precedence

EasyCalc takes the order of operation into account for all calculations. So expressions like 2 + 3 \* 4 will yield 14 rather than 20. According to the math that I know, exponents are at the top of the order of precedence (of course behind parenthesis) so an expression like 4 ^ 3 ^ 2 should be evaluated right-to-left as in 4 ^ (3 ^ 2) yielding 262,144. However most scientific calculators ignore this rule, evaluating left-to-right to yield an incorrect 4,096. EasyCalc obeys the rule here. The order of operator precedence implemented in EasyCalc is {roots, powers, reciprocal, factorial}, {functions (sin, cos, tan, log, nl, etc.)}, {explicit multiplicative operators}, {additive operators}, and {implicit multiplication}.

* Postfix input method

 The postfix input method allows you to type expressions just like you would write them on paper. For example, if you want to evaluate the expression sin 30, using Windows Calculator, it will require you to type the operand (30) first, and afterwards hit the sin function. In contrast, EasyCalc is declarative, allowing you to type expressions as is. So, for more complex expressions like 2 sin 30, you would type it as is without going through the imperative process of inputting it as sin(30) \* 2. Having said that, expressions like 2 sin 30 + 5! will give a tricky result of 241 instead of the expected 121. This is because implicit multiplications (in this case 2 \* sin(30)) are evaluated last in the order of operations.

* Multifarious features

EasyCalc supports calculations in four number systems: binary, octal, decimal and hexadecimal. You can perform any operation in any number system you choose, something that even many commercial calculators don't support. The application starts in decimal mode but you can switch to other number systems by clicking on the appropriate button. Other useful features built into EasyCalc are digit grouping (for all number systems), temporal data storage and retrieval, and copying unto the clipboard. These features are either accessible from the menu bar of the application or directly from the application workarea.

*Overview of the Code:*

I have used six different classes to make program easier to write, understand and fix it. I have used private and public inside methods and many different instant variables. I have used for loops, and many Booleans (True/False), and if/ else statement to run this program. Also I have imported many different java libraries as well. I have also used Switch and Case as well. I have used keyword Super inside a sub-class method to call a method defined in the super class. Private methods of the super-class cannot be called. Only public and protected methods can be called by the super keyword. It is also use dby class constructors to invoke constructors of its parent class.