The Calculator Application

Learning Objectives

The development process of the Calculator application will aid the students to:

- Create a simple Java console application
- Understand the object-oriented concepts of inheritance, polymorphism and data hiding
- Create application which request input from users, validate, process the input received and provide desired output.
- Use features of java like type conversion, interfaces, inheriting interfaces, looping and branching, packages and I/O classes.

Understanding the Calculator Application

The Calculator application performs both basic and scientific operations. The application provides user an option to choose between the basic mode and scientific mode. Based on the option selected by the user, the application calls the corresponding class and the user can perform various mathematical operations provided in the class. There is a base class in the application which contains all the Basic methods for calculation, there should be a Scientific Class for scientific methods. The application validates the user input also and provides appropriate messages when wrong input is given by the user.

Creating the Calculator Application

To create the Calculator application, 5 java files were created.
First, an interface iCalc, with the file name “iCalc.java” is created. Then, we create the base class Calculate, with the file name “Calculate.java” which contains all the methods for calculation.
After the base class, two classes, Calculator and ScientificCalculator, with the file names as “Calculator.java” and “ScientificCalculator.java” are created. These classes call the methods defined in the base class Calculate.
Class Calculator contains an instance of Class Calculate, whereas Class ScientificCalculator inherits Class Calculate and then uses its methods.
After creation of all the above classes, a main class UseCalculate is created, with the file name “UseCalculate.java” which provides creates instances of Class Calculator or Class ScientificCalculator, based on the option selected by user.

Creating the Java Files

The iCalc Interface (iCalc.java)

Interface iCalc provides the structure of methods which can be used in any calculator application. It contains the following two methods:

- doCalculation(): Declares a method for providing methods for calculation.
- getResult(): Declares a method for extracting the result of the calculation.

The Calculate Class (Calculate.java)

Class Calculate contains the business logic of the Calculator application. It contains the methods for calculation of various mathematical operations like addition, divide and tangent. Class Calculate uses interfaces by implementing Interface iCalc. The class contains following methods:

Method	Description
Calculate()	Default constructor for the class without any arguments.
Calculate(Double dblNum, char cOperator)	Constructor containing two arguments. This constructor is used for scientific calculations.
Calculate(int iFirstNum, char cOperator, int iSecondNum)	Constructor containing three arguments. This constructor is used for basic calculations.
doCalculation()	Calculates the result based on the numbers and operator inputted by the user. Overriding the doCalculation function of iCalc interface.
getResult()	Prints the result of calculation. Overriding the getResult function Of iCalc interface.
checkSecondNum()	In case of division of two numbers, it checks for value 0 in the second number entered.
checkInt()	Checks if basic calculation is performed
checkDouble()	Checks if scientific calculation is performed

The Calculator Class (Calculator.java)

Class Calculator calculates basic operations, namely, addition, subtraction, multiplication and division of two numbers. The class provides option to user to enter first number to be calculated, then the operation to be performed and then, the second number to be used for calculation. The input is received using java class BufferedReader. Class calculator creates an object of Class Calculate, by calling its constructor by passing three arguments, First Number, Operator and Second Number. After the creation of object of Class Calculate, doCalculation() method is called followed by getResult() method, which presents the result of calculation to the user.

Class Calculator also uses a do-while loop to provide an option to the user perform multiple calculations, till the user does not indicate the end of processing by typing ‘n’.

The ScientificCalculator Class (ScientificCalculator.java)

Class Calculator performs calculation of scientific operations, namely, sine, cosine, tangent and log of a number. The class provides option to user to enter the operation to be performed and the number to be calculated, the input is received using java class BufferedReader. Class ScientificCalculator inherits Class Calculate to use its methods. The class passes the user entered values to Class Calculate by calling its constructor having two arguments, Operator and the Number. The class calls doCalculation() method which is followed by getResult() method, which shows the result of calculation to the user.

Class ScientificCalculator also uses a do-while loop to provide an option to the user perform multiple calculations, till the user does not indicate the end of processing by typing ‘n’.

The UseCalculator Class (UseCalculator.java)

Class Calculator provides two options to the user, Basic or Scientific. Based on the option entered by the user, the class creates an instance of the Class Calculator for Basic operations or an instance of the Class ScientificCalculator for scientific operations.

Class UseCalculator also uses a do-while loop to provide an option to the user perform multiple calculations, till the user does not indicate the end of processing by typing ‘n’.