Calculator application POC

======================================================

**Environment:**

Java 1.7

Junit 4

Windows 7

Maven 4

cobertura 2.5 : code coverage report generated at “calc-app\target\site\cobertura” path

Maven Build Goals: **clean install cobertura:Cobertura**

Calculator application that can add and subtract two numbers.

The inputs should initially be read from two separate files (numberA.txt, numberB.txt) which will both contain only one number.

NumberReaderUtil : This class read the input file and parse the string argument

NumberReaderUtil.getNumberForInputFile(CalculatorConstants.NUMBER\_A\_FILE\_PATH);

CalculatorConstants will hold input and output file paths

The result should be appended to another file (result.txt) where the latest result of each call is written as a new line at end of file.

The below method write the final results into result.txt file

ResultWriterUtil.*writeResult*(*buffer*.toString(),CalculatorConstants.***RESULT\_FILE\_PATH***);

Where buffer appended with equation like for each call and finally *buffer contenet written to the* result.txt file

2147 + 175 = 2322

2147 - 175 = 1972

2147 - 175 = 1972

Q) I am appending like *buffer*.append(numberA+" - "+numberB+" = "+result+"\n"); for each operation.

Is this the way are you looking for or de we need any changes?

         The application should be designed so it is easily extended.  For example, multiplication may be added later on but is not required as part of this exercise.

Defined **interface** CaluculationService with add(-,-), sub(-,-) and later multiplication be added

         The application should be thread safe

**Used synchronization : public** **synchronized** **int** add(**int** x, **int** y)., and StringBuffer

         Unit testing and code quality is essential.  Consider this a production quality application and ensure testing and code quality is sufficient.

Unit tested TestCaluculationService for add and sub methods, TestNumberReaderUtil and TestResultWriterUtil for file read and write test cases.

         All appropriate edge cases and exceptions should be considered.

Coded below runtime exceptions:

FileReadException for FileNotFoundException

InvalidNumberException for NumberFormatException

NegativeResultException for out of int range values / edge cases