**Lab Exercise 3**

1. Write a program showing that a method with its own try block does not have to catch every possible error generated within the try. Some exceptions can slip through to, and be handled in, other scopes.
2. Write a subclass called **ExceptionSubClass** that is derived from **ExceptionClass** and that adds an integer data field called data2 and a public method called CheckException() that will check if data1 is equal to 10 and data2 is equal to 15, the CheckException () method should return “Method Executed!” otherwise this method should throw an exception manually and which should be handled in other scope.

Also, create methods called SetData2() and GetData2() for setting and retrieving the value of data1 and data2, as well as a constructor that accepts arguments for the starting values of data1 and data2. data1 is data member of **ExceptionClass**.

1. Write a program that illustrates rethrowing an exception. Define methods **Method1** and **Method2**. **Method2** should initially throw an exception. **Method1** should call **Method2**, catch the exception and rethrow it. Call **Method1** from method **main**, and catch the rethrown exception.