JAVA: Reflection Class Information

## We will explore java reflection which allows an executing java program to examine and manipulate internal properties of the program.

* It’s possible for a java class to obtain the names of all its member and display them.
* Also useful when we don’t know the names of classes at compile time.

How to setup reflection class

**import** java.lang.reflect.\*;

EXAMPLE

##### public class DumpMethods {

##### public static void main(String args[])

##### {

try {

##### Class c = Class.forName(args[0]);

##### Method m[] = c.getDeclaredMethods();

##### for (int i = 0; i < m.length; i++)

##### System.out.println(m[i].toString());

##### }

##### catch (Throwable e) {

##### System.err.println(e);

##### }

##### }

##### }

Java reflection is useful because it supports dynamic retrieval of information about classes and data structures by name, and allows for their manipulation within an executing Java program. This feature is extremely powerful and has no equivalent in other conventional languages such as C, C++, Fortran, or Pascal.

Glen McCluskey has focused on programming languages since 1988. He consults in the areas of Java and C++ performance, testing, and technical documentation.