**Java Reflection API**

Java Reflection is a *process of examining or modifying the run time behavior of a class at run time*.

The java.lang.Class class provides many methods that can be used to get metadata, examine and change the run time behavior of a class.

The java.lang and java.lang.reflect packages provide classes for java reflection.

### Where it is used

The Reflection API is mainly used in:

* IDE (Integrated Development Environment) e.g. Eclipse, MyEclipse, NetBeans etc.
* Debugger
* Test Tools etc.

## A Simple Example of Java Reflection

Here is an easy to understand example of some code that searches up a class hierarchy to see if a particular method is defined somewhere in the inheritance tree. In other words, we are just looking for a method in the parent classes of a given class.

*import java.lang.reflect.Method;*

*import java.lang.Class;*

*Method findInheritedMethod(Class classType,*

*String theMethodName, Class [] methodParamTypes)*

*{*

*Method inheritedMethod = null;*

*while(classType != null) {*

*try {*

*inheritedMethod =*

*classType.getDeclaredMethod(theMethodName, methodParamTypes);*

*break;*

*}*

*catch (NoSuchMethodException ex) {*

*classType = classType.getSuperclass( );*

*}*

*return inheritedMethod ;*

*}*