Debuggers

Mitsiu Alejandro Carreño
Sarabia

Agenda

Debuggers

There are two types of errors (logical and syntax errors). Logical errors do not prevent the application from compiling and running, but might produce unexpected results.

Debuggers allows us overlook an app execution to find and fix logical errors.

Jdb

The Java Debugger (JDB) is a simple command-line debugger for Java classes.

The jdb command and its options call the JDB. The jdb command demonstrates the Java Platform Debugger Architecture (JDBA) and provides inspection and debugging of a local or remote Java Virtual Machine (JVM).

Jdb

Start a jdb session

There are many ways to start a JDB session. The most frequently used way is to have JDB launch a new JVM with the main class of the application to be debugged. Do this by substituting the jdb command for the java command in the command line. For example, if your application's main class is MyClass, then use the following command to debug it under JDB:

```
jdb MyClass
```

When started this way, the jdb command calls a second JVM with the specified parameters, loads the specified class, and stops the JVM before executing that class's first instruction.

Jdb

Breakpoints

These are markers that can be set on any line of executable code. When the application reaches a breakpoint, execution stops, allowing us to examine the values of variables to help determine if there are any logical errors.

Breakpoints

Breakpoints can be set in JDB at line numbers or at the first instruction of a method, for example:

`stop at MyClass:22`	sets a breakpoint at the first instruction for line 22 of the source file containing MyClass.
`stop in MyClass.<method>`	sets a breakpoint at the first instruction in <method> inside MyClass.

jdb commands

Once the jdb session is running you can use any of the following commands:

- run: After you start JDB and set breakpoints, you can use the run command to execute the debugged application.
- step: Advances execution to the next line whether it is in the current stack frame or a called method.
- dump: For variables or fields of primitive types, the actual value is printed. For objects, the dump command prints the current value of each field defined in the object. Static and instance fields are included.

Jdb

jdb commands

- help or ?: Display the list of recognized commands with a brief description.
- next: Advances execution to the next line in the current stack frame.
- cont: Continues execution of the debugged application after a breakpoint, exception, or step.

References

- https://docs.oracle.com/javase/8/docs/technotes/tools/windows/jdb.html
- Cómo programar en Java. Deitel, 9th ed (2012)