

## Debugging Java applications

To debug Java programs, you can use the Java Debugger (JDB) application or other debuggers that communicate by using the Java Platform Debugger Architecture (JPDA) that is provided by the SDK for Windows.

More information about problem diagnosis using Java can be found in the [Diagnostics Guide](#).

### Java Debugger (JDB)

The Java Debugger (JDB) is included in the SDK for Windows. The debugger is invoked by the `jdb` command; it attaches to the JVM using JPDA.

To debug a Java application:

1. Start the JVM with the following options:

```
java -Xdebug -Xrunjdwp:transport=dt_shmem,server=y,address=<port>  
<class>
```

The JVM starts up, but suspends execution before it starts the Java application.

2. In a separate session, you can attach the debugger to the JVM:

```
jdb -attach <port>
```

The debugger will attach to the JVM, and you can now issue a range of commands to examine and control the Java application; for example, type `run` to allow the Java application to start.

For more information about JDB options, type:

```
jdb -help
```

For more information about JDB commands:

1. Type `jdb`
2. At the `jdb` prompt, type `help`

You can also use JDB to debug Java applications running on remote machines. JPDA uses a TCP/IP socket to connect to the remote JVM.

1. Start the JVM with the following options:

```
java -Xdebug -Xrunjdwp:transport=dt_shmem,server=y,address=<port>  
<class>
```

The JVM starts up, but suspends execution before it starts the Java application.

2. Attach the debugger to the remote JVM:

```
jdb -connect com.sun.jdi.SocketAttach:hostname=<host>,port=<port>
```

The Java Virtual Machine Debugging Interface (JVMDI) is not supported in this release. It has been replaced by the Java Virtual Machine Tool Interface (JVMTI).