

# Troubleshooting problems

Sometimes the **javac** compiler or **java** interpreter will complain about errors so it's useful to understand their cause and how to quickly resolve the problem. In order to demonstrate some common error reports this code contains some deliberate errors:



Test.java

```
class test
{
    public static void main ( String[] args )
    {
        String text ;
        System.out.println( "Test " + text )
    }
}
```

A first attempt to compile **Test.java** throws up this error report:

```

C:\>javac Test.java
'javac' is not recognized as an internal or external
command, operable program or batch file.

C:\>_

```

- Cause – the **javac** compiler cannot be found
- Solution – edit the system **PATH** variable, as described on page 11, or use its full path address to invoke the compiler

## Hot tip



The path address must be enclosed within quotation marks if it contains any spaces, as indeed it does here.

```

C:\>"C:\Program Files\Java\jdk1.7.0\bin\javac" Test.java
javac: file not found: Test.java
Usage: javac <options> <source files>
use -help for a list of possible options

C:\>_

```

- Cause – the file **Test.java** cannot be found
- Solution – navigate to the directory where the file is located, or use the full path address to the file in the command

## ...cont'd

```
Command Prompt
C:\>javac C:\MyJava\Test.java
C:\MyJava\Test.java:6: ';' expected
    System.out.println( "Test " + text >
    ^
1 error
```

- Cause – the statement is not terminated correctly
- Solution – in the source code add a semi-colon at the end of the statement, then save the file to apply the change

```
Command Prompt
C:\MyJava>javac Test.java
Test.java:1: class test is public, should be declared in a
file named test.java
public class test
    ^
1 error
```

- Cause – the program name and class name do not match
- Solution – in the source code change the class name from **test** to **Test**, then save the file to apply the change

```
Command Prompt
C:\MyJava>javac Test.java
Test.java:6: variable text might not have been initialized
    System.out.println( "Test " + text > ;
    ^
1 error
```

- Cause – the variable **text** has no value
- Solution – in the variable declaration assign the variable a valid **String** value, for instance = "**success**", then save the file

```
Command Prompt
C:\MyJava>javac Test.java
C:\MyJava>java Test
Test success
C:\MyJava>_
```

Beware



You must run the program from within its directory – you cannot use a path address as the **java** launcher requires a program name, not a file name.