

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:

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



To edit the environment variables in Windows, you can open the **System Properties, Advanced** tab and click the **Environment Variables** button. Select **Path, Edit** and add **C:\Java\bin**, then click **OK** to apply the change.



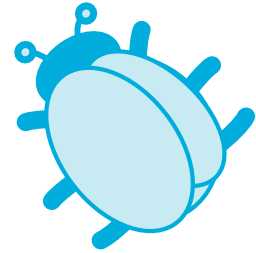
- Cause – the **javac** compiler cannot be found.
- Solution – edit your system environment with the command **setx path "%path%;C:\Java\bin"** or use the full path to invoke the compiler with the command **C:\Java\bin\javac Test.java**

```
Command Prompt
C:\MyJava> javac Test.java
error: file not found: Test.java
Usage: javac <options> <source files>
use --help for a list of possible options
C:\MyJava>
```

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



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

```
Command Prompt
C:\MyJava> javac Test.java
Test.java:1: error: 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: error: 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.

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.

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