

Go to the Oracle site:

www.oracle.com/technetwork/java/javase/downloads/

ORACLE

Sign In/Register Help Country ▾ Communities ▾ I am a... ▾ I want to... ▾ Search

Products Solutions Downloads Store Support Training Partners About OTN

Oracle Technology Network > Java > Java SE > Downloads

Overview Downloads Documentation Community Technologies Training

Java SE Downloads

 **DOWNLOAD ▾**
Java Platform (JDK) 8u25

 **DOWNLOAD ▾**
NetBeans with JDK 8

Java Platform, Standard Edition

Java SE 8u25
This release includes important security fixes. Oracle strongly recommends that all Java SE 8 users upgrade to this release.
[Learn more ▶](#)

- Installation Instructions
- Release Notes

JDK
DOWNLOAD ▾

Java SDKs and Tools

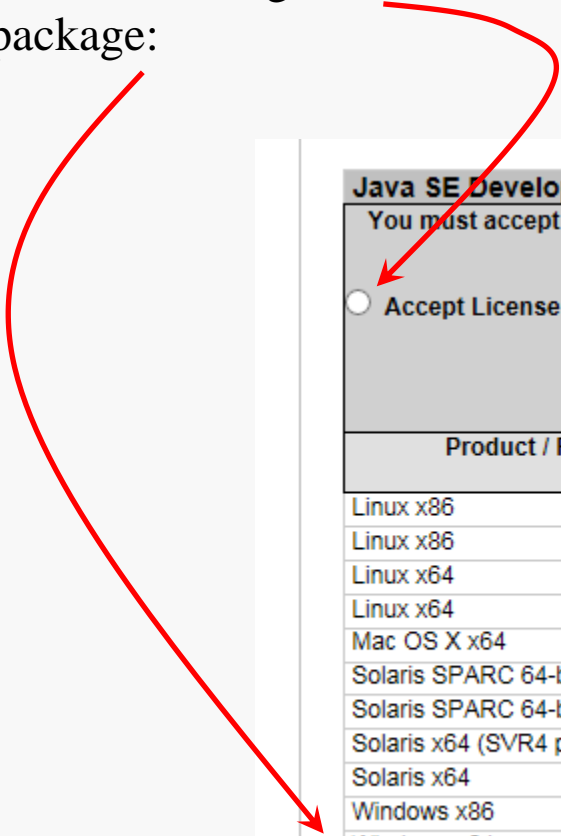
- Java SE
- Java EE and Glassfish
- Java ME
- Java Card
- NetBeans IDE
- Java Mission Control

Java Resources

- Java APIs
- Technical Articles
- Demos and Videos
- Forums
- Java Magazine
- Java.net
- Developer Training
- Tutorials

Select the download link

Scroll down, agree to the License Agreement, and select the appropriate installation package:



Java SE Development Kit 8u25
You must accept the [Oracle Binary Code License Agreement for Java SE](#) to download this software.

Accept License Agreement Decline License Agreement

Product / File Description	File Size	Download
Linux x86	135.24 MB	jdk-8u25-linux-i586.rpm
Linux x86	154.88 MB	jdk-8u25-linux-i586.tar.gz
Linux x64	135.6 MB	jdk-8u25-linux-x64.rpm
Linux x64	153.42 MB	jdk-8u25-linux-x64.tar.gz
Mac OS X x64	209.13 MB	jdk-8u25-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	137.01 MB	jdk-8u25-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	97.14 MB	jdk-8u25-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	137.11 MB	jdk-8u25-solaris-x64.tar.Z
Solaris x64	94.24 MB	jdk-8u25-solaris-x64.tar.gz
Windows x86	157.26 MB	jdk-8u25-windows-i586.exe
Windows x64	169.62 MB	jdk-8u25-windows-x64.exe

We recommend using the 64-bit version for your platform.

In my case, I downloaded the executable installation file

```
jdk-8u25-windows-x64.exe
```

The first few dialogs are straightforward. I recommend:

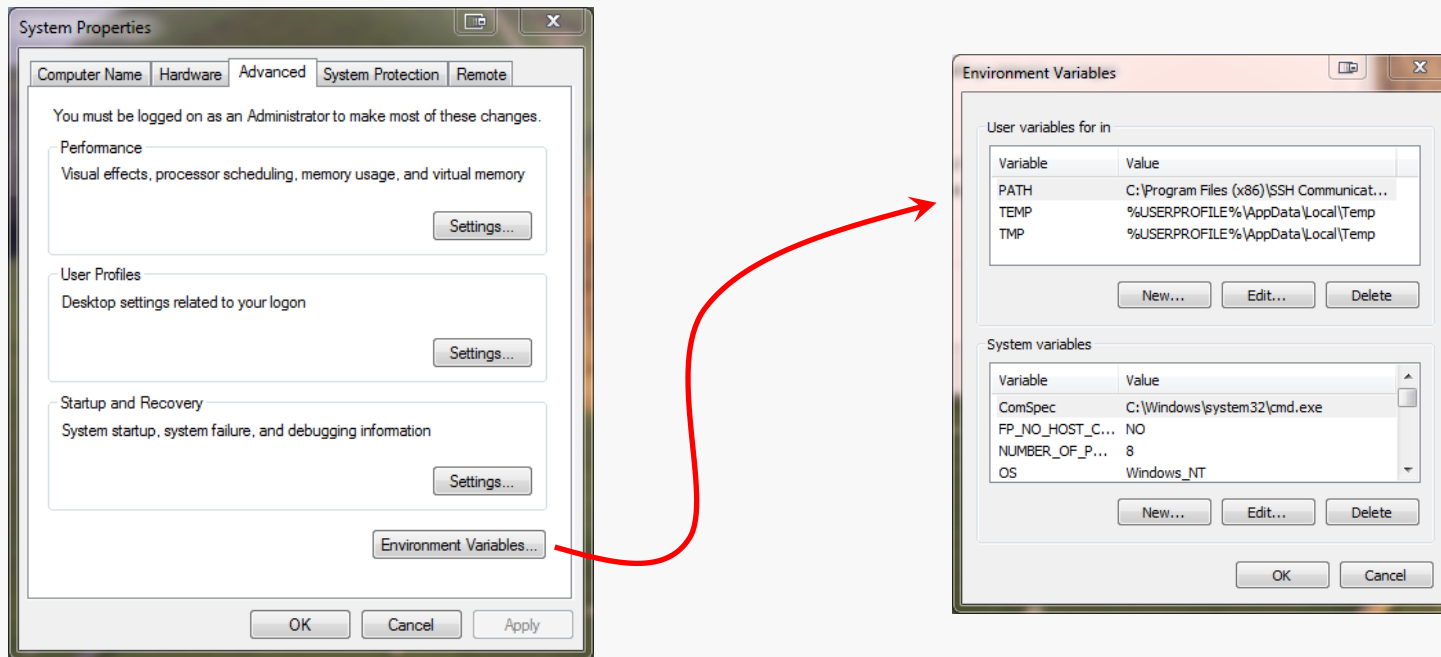
- Take the default option to install the Java source and the JRE
- Install below the root directory of a drive; in my case: C:\jdk1.8.0_25\

The installation may take awhile, especially on a slower machine.

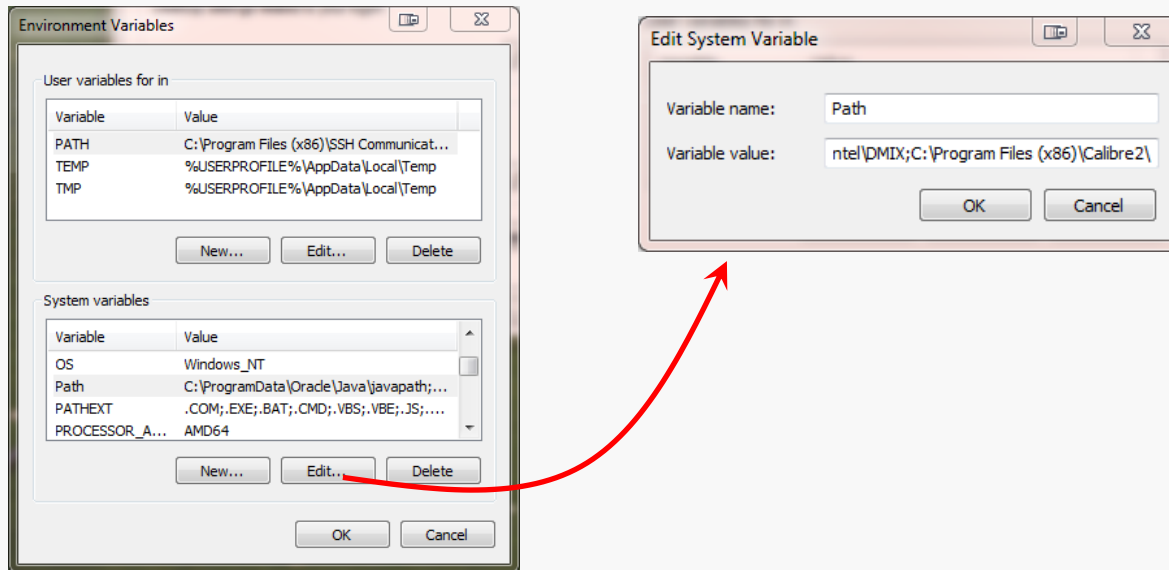
You'll need to add the bin directory for your Java install to your system path, or use the full path to the java compiler and/or interpreter when you want to use them.

In my case, the compiler (`javac.exe`) and interpreter (`java.exe`) are in the directory:
`C:\jdk1.8.0_25\bin\`

For Windows 7, go to Control Panel + System + Advanced System Settings



Scroll to Path under System variables and select Edit:



I recommend copying your current Path variable to a text editor (Ctrl-A, Ctrl-C to copy it, then Ctrl-V to paste it in).

Edit so your Java `bin` directory is in the Path, then copy and paste it back into the Edit System Variable dialog shown above.

Then "OK" out of the System applet from the Control Panel.

Open a Windows shell (command prompt) and see if it all worked:

```
C:\workspace> java -version
java version "1.8.0_25"
Java(TM) SE Runtime Environment (build 1.8.0_25-b18)
Java HotSpot(TM) 64-Bit Server VM (build 25.25-b02, mixed mode)
```

If you don't get confirmation this executes version 1.8, you missed something in the earlier slides.

Let's consider the simple case of a project that does not involve Java packages:

```
C:\Windows\system32\cmd.exe
X:\Spring2015\3114\MinorProjects\1.FileNavigation\code>dir
Volume in drive X has no label.
Volume Serial Number is CADC-5E87

Directory of X:\Spring2015\3114\MinorProjects\1.FileNavigation\code

01/19/2015  05:57 PM    <DIR>          .
01/19/2015  05:57 PM    <DIR>          ..
08/24/2010  09:10 PM             919 cmdParser.java
08/24/2010  08:39 PM             316 Command.java
01/18/2015  09:07 PM          1,022 CommandProcessor.java
08/24/2010  09:45 PM          1,224 dbParser.java
01/19/2015  05:55 PM          3,552 GISRecord.java
08/24/2010  08:41 PM          2,178 Latitude.java
08/24/2010  08:41 PM             43 LatitudeHemi.java
08/24/2010  08:41 PM          2,300 Longitude.java
08/24/2010  08:42 PM             42 LongitudeHemi.java
08/27/2010  09:10 PM          9,401 Project1Prof.java
                10 File(s)          20,997 bytes
                2 Dir(s)  94,553,534,464 bytes free
```

Assuming you've installed the JDK and set the Path variable correctly, you can compile the code by executing the Java compiler, `javac.exe`, and specifying what Java source files are to be included...

In this case, I want to include all of the `.java` files in the current directory, so I can just use a wild-card on the command line:

```
C:\Windows\system32\cmd.exe
X:\Spring2015\3114\MinorProjects\1.FileNavigation\code>javac *.java
Note: dbParser.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
```

The feedback from the compiler indicates there are some potentially serious issues with the code and suggests using `-Xlint` to get more information:

```
C:\Windows\system32\cmd.exe
X:\Spring2015\3114\MinorProjects\1.FileNavigation\code>javac -Xlint *.java
dbParser.java:17: warning: [unchecked] unchecked cast
        return (T) record;
               ^
   required: T
   found:    String
   where T is a type-variable:
     T extends Object declared in class dbParser
dbParser.java:32: warning: [unchecked] unchecked cast
        return (T) record;
               ^
   required: T
   found:    String
   where T is a type-variable:
     T extends Object declared in class dbParser
2 warnings
```

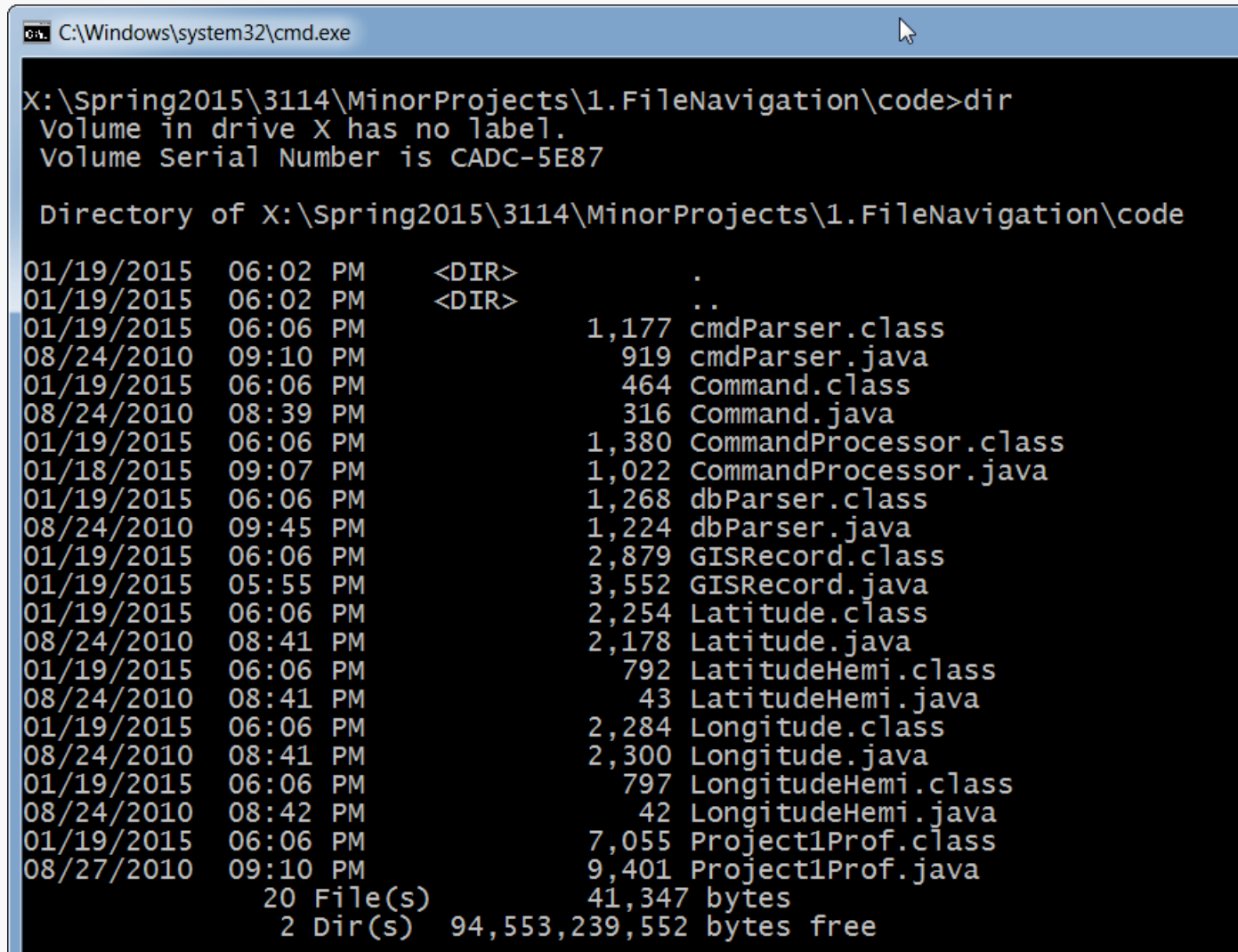

The warnings in this case are frequently found in code that uses Java generics.

The warnings about "unchecked casts" occur because of limitations in the Java language.

As long as the logic of the code is correct, the warnings can be safely ignored.

Compiling (w/o errors) yields Java bytecode files (.class):

The "main" file is the .class file produced from the .java file that contains the main() function...



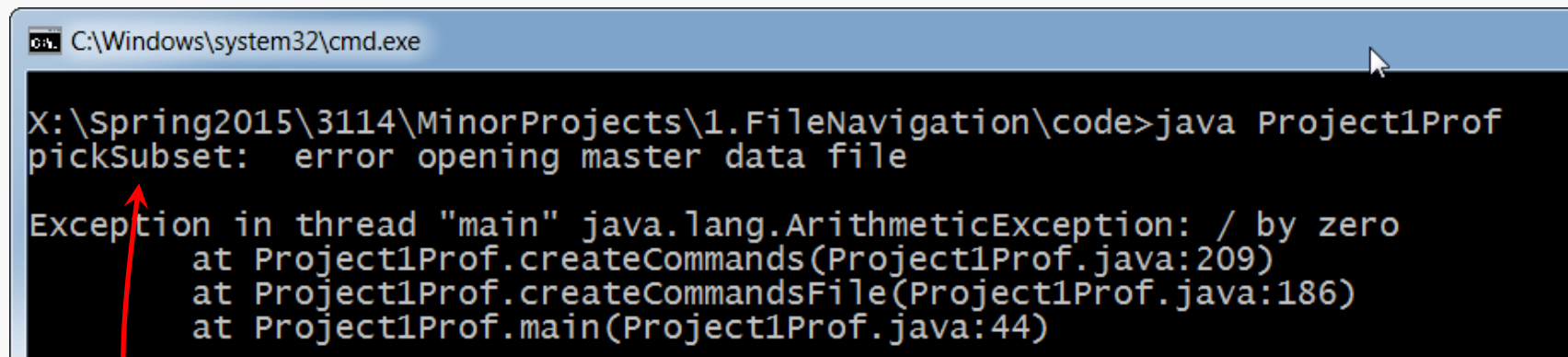
```
C:\Windows\system32\cmd.exe
X:\Spring2015\3114\MinorProjects\1.FileNavigation\code>dir
Volume in drive X has no label.
Volume Serial Number is CADC-5E87

Directory of X:\Spring2015\3114\MinorProjects\1.FileNavigation\code

01/19/2015  06:02 PM    <DIR>          .
01/19/2015  06:02 PM    <DIR>          ..
01/19/2015  06:06 PM             1,177 cmdParser.class
08/24/2010  09:10 PM             919 cmdParser.java
01/19/2015  06:06 PM             464 Command.class
08/24/2010  08:39 PM             316 Command.java
01/19/2015  06:06 PM             1,380 CommandProcessor.class
01/18/2015  09:07 PM             1,022 CommandProcessor.java
01/19/2015  06:06 PM             1,268 dbParser.class
08/24/2010  09:45 PM             1,224 dbParser.java
01/19/2015  06:06 PM             2,879 GISRecord.class
01/19/2015  05:55 PM             3,552 GISRecord.java
01/19/2015  06:06 PM             2,254 Latitude.class
08/24/2010  08:41 PM             2,178 Latitude.java
01/19/2015  06:06 PM             792 LatitudeHemi.class
08/24/2010  08:41 PM              43 LatitudeHemi.java
01/19/2015  06:06 PM             2,284 Longitude.class
08/24/2010  08:41 PM             2,300 Longitude.java
01/19/2015  06:06 PM             797 LongitudeHemi.class
08/24/2010  08:42 PM              42 LongitudeHemi.java
01/19/2015  06:06 PM             7,055 Project1Prof.class
08/27/2010  09:10 PM             9,401 Project1Prof.java
                20 File(s)              41,347 bytes
                2 Dir(s)  94,553,239,552 bytes free
```

The "main" Java `.class` file is NOT an executable file (in the usual sense).

It must be executed on a JVM (Java virtual machine) by running `java.exe`:



```
C:\Windows\system32\cmd.exe

X:\Spring2015\3114\MinorProjects\1.FileNavigation\code>java Project1Prof
pickSubset: error opening master data file

Exception in thread "main" java.lang.ArithmeticException: / by zero
    at Project1Prof.createCommands(Project1Prof.java:209)
    at Project1Prof.createCommandsFile(Project1Prof.java:186)
    at Project1Prof.main(Project1Prof.java:44)
```

Oops... the error message indicates we did not run the program correctly.

The runtime error indicates the programmer did not abort execution correctly after detecting the missing data file...

After copying the missing input file to the directory, executing the program yields some output files:

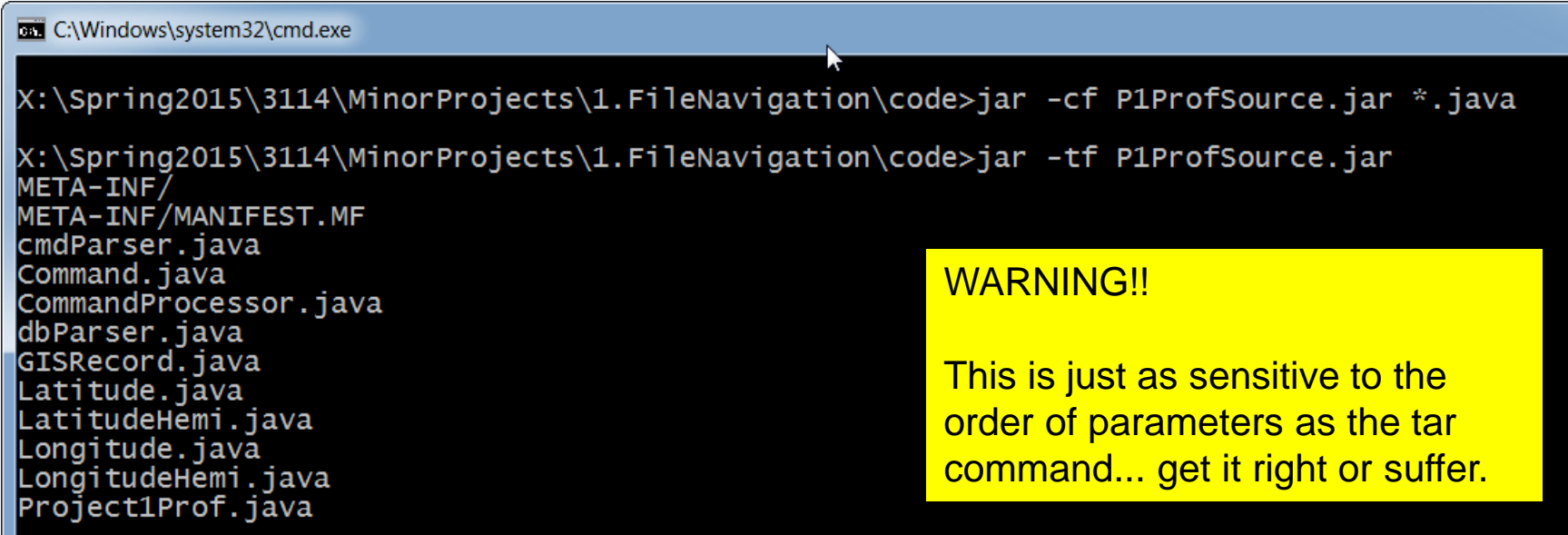
```
01/19/2015 06:19 PM <DIR> .
01/19/2015 06:19 PM <DIR> ..
01/19/2015 06:06 PM 1,177 cmdParser.class
08/24/2010 09:10 PM 919 cmdParser.java
01/19/2015 06:06 PM 464 Command.class
08/24/2010 08:39 PM 316 Command.java
01/19/2015 06:06 PM 1,380 CommandProcessor.class
01/18/2015 09:07 PM 1,022 CommandProcessor.java
01/19/2015 06:19 PM 341 Commands.txt
01/19/2015 06:06 PM 1,268 dbParser.class
08/24/2010 09:45 PM 1,224 dbParser.java
01/19/2015 06:19 PM 3,923 GISData.txt
01/19/2015 06:06 PM 2,879 GISRecord.class
01/19/2015 05:55 PM 3,552 GISRecord.java
01/19/2015 06:06 PM 2,254 Latitude.class
08/24/2010 08:41 PM 2,178 Latitude.java
01/19/2015 06:06 PM 792 LatitudeHemi.class
08/24/2010 08:41 PM 43 LatitudeHemi.java
01/19/2015 06:06 PM 2,284 Longitude.class
08/24/2010 08:41 PM 2,300 Longitude.java
01/19/2015 06:06 PM 797 LongitudeHemi.class
08/24/2010 08:42 PM 42 LongitudeHemi.java
11/13/2005 08:26 PM 33,698 MasterGISData.txt
01/19/2015 06:19 PM 4,026 profResults.txt
01/19/2015 06:06 PM 7,055 Project1Prof.class
08/27/2010 09:10 PM 9,401 Project1Prof.java
      24 File(s) 83,335 bytes
      2 Dir(s) 94,552,838,144 bytes free
```

When we test your programs, we will compile your code from the command-line, using the "real" JDK, not the compiler built into Eclipse.

Any "real" Java programmer knows now to use the tools.

A *jar file* is just a Java archive file, similar to a Linux tar file.

The JDK includes a utility, `jar.exe`, that can be used to create a jar file.



```
C:\Windows\system32\cmd.exe
X:\Spring2015\3114\MinorProjects\1.FileNavigation\code>jar -cf P1ProfSource.jar *.java
X:\Spring2015\3114\MinorProjects\1.FileNavigation\code>jar -tf P1ProfSource.jar
META-INF/
META-INF/MANIFEST.MF
cmdParser.java
Command.java
CommandProcessor.java
dbParser.java
GISRecord.java
Latitude.java
LatitudeHemi.java
Longitude.java
LongitudeHemi.java
Project1Prof.java
```

WARNING!!

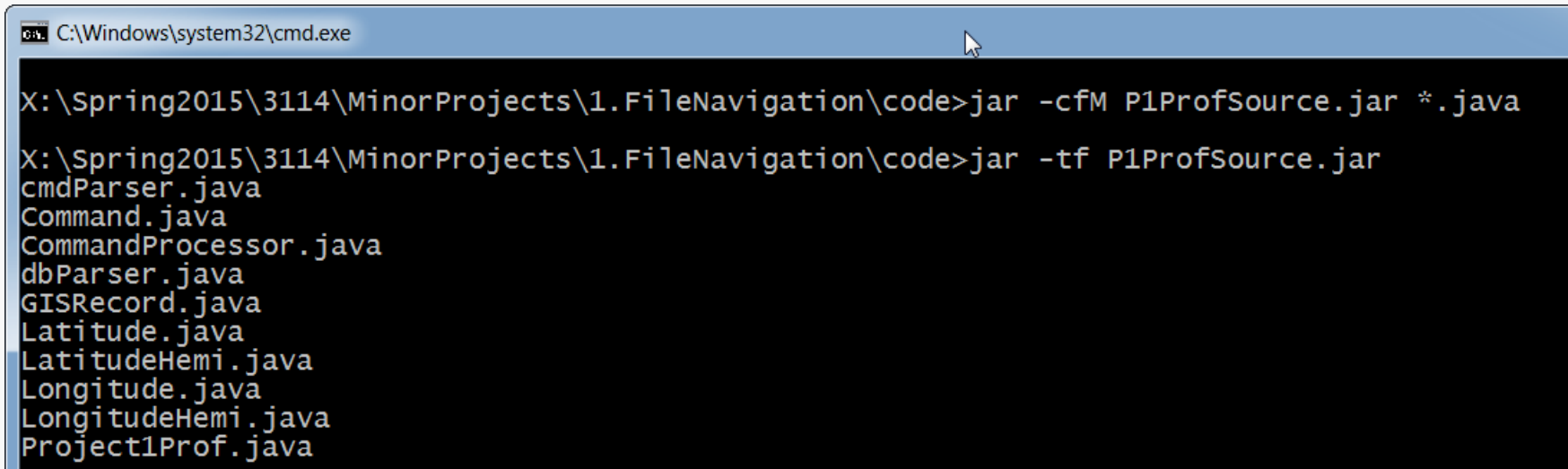
This is just as sensitive to the order of parameters as the tar command... get it right or suffer.

Execute `jar.exe` without any parameters to see a list of options.

The previous example created a jar file that packs up the Java source files in one convenient lump.

But, it contains some extra stuff you probably do not want, namely a manifest.

The `-M` option omits that content from the jar file:



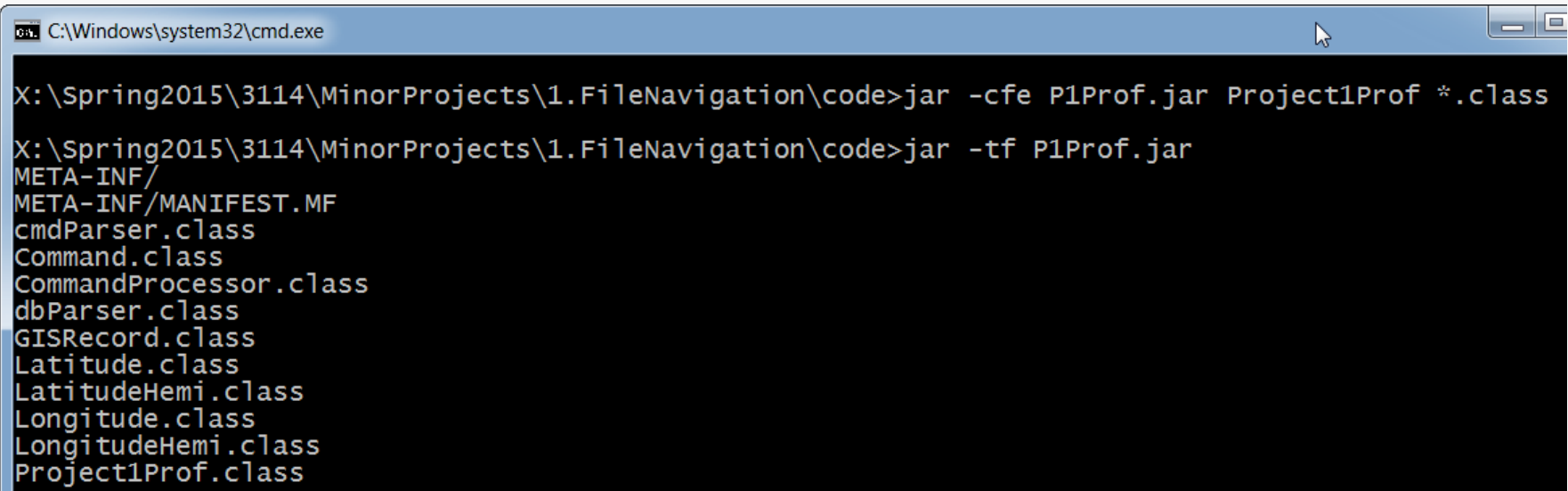
```
C:\Windows\system32\cmd.exe

X:\Spring2015\3114\MinorProjects\1.FileNavigation\code>jar -cfM P1ProfSource.jar *.java

X:\Spring2015\3114\MinorProjects\1.FileNavigation\code>jar -tf P1ProfSource.jar
cmdParser.java
Command.java
CommandProcessor.java
dbParser.java
GISRecord.java
Latitude.java
LatitudeHemi.java
Longitude.java
LongitudeHemi.java
Project1Prof.java
```

This example shows how you should prepare your source code for project submissions that do not use Java packages.

It's also possible to create a jar file that can be executed (via a JVM):



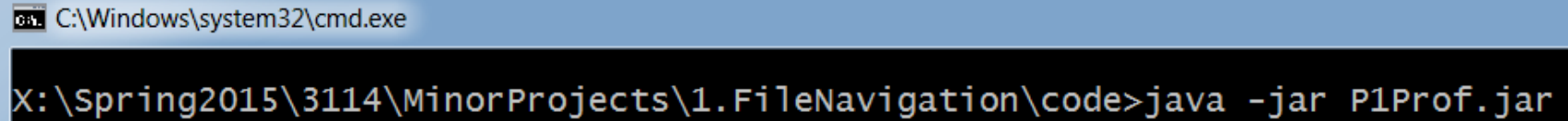
```
C:\Windows\system32\cmd.exe
X:\Spring2015\3114\MinorProjects\1.FileNavigation\code>jar -cfe P1Prof.jar Project1Prof *.class
X:\Spring2015\3114\MinorProjects\1.FileNavigation\code>jar -tf P1Prof.jar
META-INF/
META-INF/MANIFEST.MF
cmdParser.class
Command.class
CommandProcessor.class
dbParser.class
GISRecord.class
Latitude.class
LatitudeHemi.class
Longitude.class
LongitudeHemi.class
Project1Prof.class
```

The `-e` switch indicates that we are going to specify the "main" file for the program.

Note the syntax carefully.

You will probably not need to create any executable jar files in CS 3114, but you should know how anyway.

Execute the jar file by including the `-jar` switch:



```
C:\Windows\system32\cmd.exe
X:\Spring2015\3114\MinorProjects\1.FileNavigation\code>java -jar P1Prof.jar
```

If the `main()` function expects command-line parameters, list them after the name of the jar file.