Development Tools

CSE 142, Summer 2002 Computer Programming 1

http://www.cs.washington.edu/education/courses/142/02su/

22-July-2002

Readings and References

- Reading
 - » Chapter 8, Compiling Java Classes, *Introduction* to Programming in Java, Dugan
- Other References

Our Environment



22-July-2002

Us



Development Environment



Source code (we write this)



22-July-2002

Java compiler and virtual machine



Compile it



Run it



22-July-2002

Tools - BlueJ

- BlueJ is a simple "Integrated Development Environment" or IDE
 - » BlueJ uses the regular Java compiler from Sun to convert our Dog.java source file into Dog.class class files
 - » Then we can create new objects (*instantiate* them) using the class definition and manipulate them
 - » BlueJ lets us do slowly and visibly what our code can do very quickly

Tools - jEdit editor

- You don't need to use BlueJ there are many other editors and integrated development environments available
- jEdit is an editor that is written in Java
 - » open-source
 - » free to download (from jEdit.org)
 - » available on the lab machines
- jEdit is a programming editor
 - » it understands Java syntax and can highlight code

Tools - javac

- javac.exe is the Java compiler from Sun
 - » javac converts your X.java file to X.class file
- located in the Java directory tree
 - » C:\apps\jdk140\bin\javac.exe
- Operating system must be able to find compiler
 - » set PATH variable so that it includes <java>\bin
 - » Settings -> Control Panel -> System -> Advanced
 - » then select Environment Variables
 - » add or edit PATH

Variable	Value		
path	c:\apps\jdk140\bin		
	C:\Documents and Settings\finson\Local C:\Documents and Settings\finson\Local		
	C. (Documents and Decongstrinson (cocal		
	<u>N</u> ew <u>E</u> dit <u>D</u> elete		
ystem variables —			
ystem variables — Variable	Value		
ystem variables — Variable NUMBER_OF_PR	Value		
ystem variables	Value I Uindows_NT		
ystem variables	Value . 1 Windows_NT C:\WINNT\system32\os2\dll;		
ystem variables	Value . 1 Windows_NT C:\WINNT\system32\os2\dll; C:\WINNT\system32;C:\WINNT;C:\WIN		
ystem variables Variable NUMBER_OF_PR OS Os2LibPath Path PATHEXT	Value Value Value Vindows_NT C:\WINNT\system32\os2\dll; C:\WINNT\system32;C:\WINNT;C:\WIN .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS; ▼		
ystem variables Variable NUMBER_OF_PR OS Os2LibPath Path PATHEXT	Value . 1 Windows_NT C:\WINNT\system32\os2\dll; C:\WINNT\system32;C:\WINNT;C:\WINCOM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS; ▼		

🚯 Opera - [SDK D	evelopment Tools]		- D ×		
🗅 Eile Edit Viev	v <u>N</u> avigation <u>B</u> ookmarks E- <u>m</u> ail New <u>s W</u> indow <u>H</u> elp		- 8 ×		
Back - Forw	vard 🛪 Reload Home Hotlist Print New				
♥ CSE142 ♥ CS	E490a \heartsuit History \heartsuit JavaAPI \heartsuit javadoc \heartsuit JavaSpec \heartsuit JavaSrc \heartsuit JavaTools	🖤 UWAnima 🖤 UWGrap 🛛	Google 🌀		
• Basic Tools (javac, java, javadoc, appletviewer, jar, jdb, javah, javap, extcheck)					
<u>Remote Method Invocation (RMI) Tools</u> (rmic, rmiregistry, rmid, serialver)					
Internationalization Tools (native2ascii)					
• Security Tools (keytool, jarsigner, policytool)					
• Java IDL and RMI-IIOP Tools (tnameserv, idlj, orbd, servertool)					
• Java Plug-in TM Tools					
Basic Tools					
These tools are the foundation of the Java 2 SDK. They are the tools you use to create and build applications.					
javac	The compiler for the Java programming language.	[Solaris and Linux] [Win32	2]		
java	The launcher for Java applications. In this release, a single launcher is used	[Solaris] [Linux] [Win32]			
	both for development and deployment.				
	The old deployment launcher, jre , is no longer provided.				
javadoc	API documentation generator. See <u>Javadoc 1.4 Home Page</u>	[Solaris and Linux] [Win32	2]		
appletviewer	Run and debug applets without a web browser.	[Solaris and Linux] [Win32	2]		
jar	Manage Java Archive (JAR) files.	[Solaris and Linux] [Win32	2]		
jdb	The Java Debugger.	[Solaris and Linux] [Win32	2]		
javah	C header and stub generator. Used to write native methods.	[Solaris and Linux] [Win32	2]		
javap	Class file disassembler	[Solaris and Linux] [Win32	2]		
extcheck	Utility to detect Jar conflicts.	[Solaris and Linux] [Win32	<u>_</u>]		
े ि 🕒 👌 file://localhost/C:/apps/jdk140/docs/tooldocs/tools.html					
SDK Development Tools					

Classpath

- Compiler must be able to find other classes that we use
 - » Standard java classes found by the default setup
 - » Special classes like uwcse Shape must be located
 - in BlueJ we set uwcse.jar as a user library
 - when running java ourselves we set the library as part of the class path
- Classpath tells the compiler where to look
 - » javac -classpath .;\cse\lib\uwcse.jar *.java

Tools - java

- java.exe runs the Java virtual machine from Sun
 - » the virtual machine reads your class files and actually runs the program you've written
- java.exe is located in the Java directory tree
 » C:\apps\jdk140\bin\java.exe
- Operating system must be able to find java.exe » Path should be set already for the compiler

Launching a java program

- The java tool launches a Java application. It does this by
 - » starting a Java runtime environment
 - » loading a specified class
 - » invoking that class' main method.
- The method declaration must look like the following:
 - » public static void main(String args[])

Classpath

- Just as with the compiler, java must be able to find other classes that we use in the program
 - » Standard java classes found by the default setup
 - » Special classes like uwcse Shape must be located
 - in BlueJ we set uwcse.jar as a user library
 - when running java ourselves we set the library as part of the class path
- Classpath tells java where to look
 - » java -classpath .;\cse\lib\uwcse.jar Director

Tools - File System

- It is important that you be able to find the source files that you write and understand what they are and how they relate
- Look at the directories
 - » find the .java files and the .class files
- Notice that the files that I supplied for homework 4 are in a subdirectory "skel"
 - » they are part of package "skel" as a result
 - » more about packages in later lectures