**Eclipse IDE (Integrated Development Environment)**

**Version 1.4 of Instructions**

Obtaining software (no license needed)

1. Go to: <http://www.eclipse.org>.
2. Select downloads menu (forth option on the left).
3. Select latest release (first option).
4. Select Windows 98/ME/2000/XP (first option).
5. Download should start automatically (Zip File).

For your convenience, your instructor has provided a direct link to the download at:  
  
http://download.eclipse.org/downloads/drops/S-M3-200211151436/eclipse-SDK-M3-win32.zip

Installing software (Win Zip needed)

1. First install the Java Development Kit (j2sdk1.4.0” from Sun. Please make sure it is installed in the root of C drive.
2. Second open the “eclipse-SDK-M3-win32.zip” file in Win Zip.
3. Extract it to the root of C (it will automatically create a directory called *eclipse* in the root of *C*).
4. To run: Start 🡪 Run 🡪 “c:\eclipse\eclipse” (no install needed)
5. If there are problems (such as it does not run), down load the file called “run\_elclipse.bat” to the root of C directory from your instructor’s software web site and run it instead. This will set the missing class paths for you. For example: Start 🡪 Run 🡪 “c:\run\_eclipse”

Running the First Time

1. After reading Welcome Window close it.
2. **Create a Java Project**.  
     
   File 🡪 New Project🡪 “Project Name”
   1. A new directory with this name will be created in the sub-directory called *eclipse\workspace*. It is recommended that no intervening spaces be used in project name.
   2. If you wish to write on a floppy disk as when working in the lab, it is best to leave this workspace where it is, and then export your Java project to a jar file on the floppy disk.. This can be easily imported to any other system.
   3. Eclipse will create in this project sub-directory a *dot project* file.  
      (Do not modify it. It can be viewed in notepad and it shows an XML file describing Java Environment used.
   4. It will also create in this sub-directory a *dot classpath* file.  
      (Do not modify it. It can also be viewed in notepad showing an XML file describing Java Class paths used.
3. **Create a Java Class**

File 🡪 New Class🡪 Opens a window called *Java Class*

* 1. *Java Class Name* should be inserted in text field labeled name.
  2. Package name is optional, if not used all classes go into default package.  
     It is recommended if you are in CIS 234 to omit the package.
  3. All browsers for directories should be left alone.
  4. If wish default methods to be generated check them. For example:  
     ***public******static******void*** *main(String[] args)*

1. **Complete the Java Class**

Fill in the class *data members* and *methods*; if the yellow light bulb comes on, fix the error before continuing.

* 1. Completion of Java statements will automatically become invoked after the dot.
  2. If you wish to force completion after partially spelled words, depress *control* and *space* bar *keys* together. This will bring up a list box of Java keywords and class names.
  3. Choosing a class name in another package will automatically generate import statements at th3e beginning of the class. For example *JOptionPane* will insert the import for the *javax.swing* package.
  4. If you right click in the editing window, a menu pops up with a formatter option.
  5. If you right click in the outline window a data member, a menu pops up with a generate getter and setter option.
  6. When done, save using: File 🡪 Save All This creates the Java class files.

1. **To run Java main method**
   1. Run 🡪 Run As 🡪 Java Application (to run the first time)
   2. Run 🡪 Run Last Launched (to run subsequent times)
2. **To run Java Applet (CIS 304 classes only)**
   1. Run 🡪 Run As 🡪 Java Applet (to run the first time)
   2. Run 🡪 Run Last Launched (to run subsequent times)
3. **To debug**
   1. First set break points using double click in left margin (or you may set cursor and use: Run 🡪 Add/Remove Breakpoint.
   2. Run 🡪 Debug As 🡪 Java Application or Applet (to debug the first time)
   3. Run 🡪 Debug Last Launched (to run subsequent times)
   4. Note that a *Variables Window* automatically displays.
   5. To continue: Run 🡪 Resume or Terminate or other option.
   6. If the Variables window does not show go to:   
      Window 🡪 Reset Perspective.
4. **To export a Java Project**
   1. File 🡪 Export 🡪 Select export destination   
      Choose Jar File not Zip File or File System.
   2. Choose package or Java files.
   3. Finish.

Setting Preferences and Odds and Ends

* + 1. It is possible to customize the settings. Since one of the options is format, you may, for example, insure the braces are lined up by going to: Window 🡪 Preferences 🡪 Java 🡪 Code Formatter, and apply the first three of the five check boxes. The default is only the last check box being on.
    2. It is also possible to get the line numbers to show by customization or preferences as well. Go to: Window 🡪 Preferences 🡪 Java 🡪 Editor, and apply check box “show line numbers.”
    3. Note that if you go to the menu Source 🡪 Format, the code will be formatted to perfect form for you.
    4. Note also that if you go to the menu Source 🡪 Generate Getter / Setter, the accessors and mutators will be generated for you.
    5. Also there is an import option. Be sure and create a project before importing to hold the classes. It will import almost any format. It is especially good at importing jar files. Note examples from instructor web sites are typically in a jar format.
    6. There is also an export option discussed above in part 8.

Eclipse instructions - Version 1.4 Winter 2003. If you find any errors please contact Professor Bell.