

Appendix
B

WORKSHOP

*Get on the
Fast Track!*



TM

**SYS-ED/
COMPUTER
EDUCATION
TECHNIQUES, INC.**

1 Object Oriented Programming

Objectives:

- C Defining class and its members.
- C Understanding scope and lifetime.
- C Managing Inheritance in Java.

Exercises:

- C Create a class to maintain information about the agency computer hardware inventory.
- C Have the class contain methods to add, delete and search for computer hardware.
- C A get/set will be written for each attribute. Read only attribute will only have a get method.
- C A special hardware class will be built for tracking Computer Servers.
This class will be a child (inherited) from the regular computer hardware inventory class.
The classes will be stored internally within the source and externally in a class library.

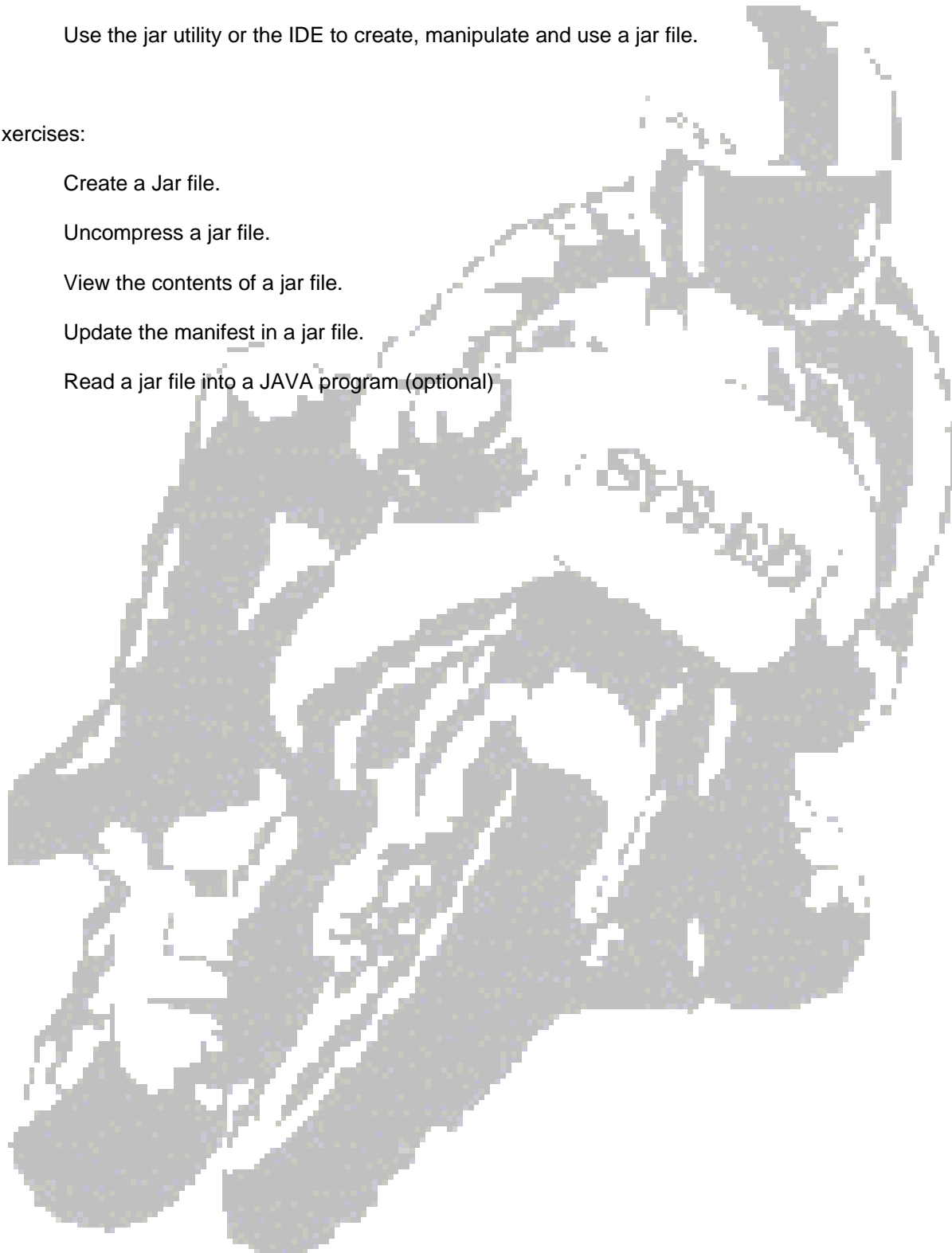
2 Java Archive File Format

Objectives:

- C Use the jar utility or the IDE to create, manipulate and use a jar file.

Exercises:

- C Create a Jar file.
- C Uncompress a jar file.
- C View the contents of a jar file.
- C Update the manifest in a jar file.
- C Read a jar file into a JAVA program (optional)



3 Collections

Objectives:

- C To work with various types of collections.
- C Code the interfaces, group operations, and methods

Exercises:

- C Create a vector, enqueue and dequeue elements, clear all elements and read an element with removing it.
- C Create an array and sort the contents.
- C Search a sorted array based on a key.
- C Create a property list, retrieve a single element and enumerate all elements.

4 UI with JFC

Objectives:

- C Create a GUI window with various components.
- C Perform code when an event occurs.
- C Handling error situations.

Exercises:

- C Layout a window with various layout components.
- C Create a container with a text box, radio button, list box and push buttons.
- C Associate code with each of the push buttons to retrieve the values from the other components.
- C Create a simple menu with associated events.
- C Create and display a dialog box.

5 UI with JFC - Advanced Topics

Objectives:

- C To create a GUI window with additional components.
- C Display standard type dialog boxes.
- C Create more sophisticated windows and menus.

Exercises:

- C Create windows with combo boxes, JOptionPane, panels and Scroll Panes.
- C Create Listeners for text boxes and other components (List Boxes).
- C Code Tool tips for the controls.
- C Display icons and pictures.
- C Enable keyboard operations.
- C Create and load a table.

6 Input/Output with Streams

Objectives:

- C Code programs with the OutputStream, InputStream and PrintStream.
- C Code programs for performing sequential and random I/O.

Exercises:

- C Write a program that reads and write a sequential file.
- C Write a program that reads and writes a random file.
- C Write a program that will read and write to a memory buffer (optional).
- C Write a program that will read and write to a pipe (optional).

7 JDBC**Objectives:**

- C Understand a Connection String and how to create it.
- C Access a database and handle exception and error conditions.
- C Update a database and understand locking considerations.

Exercises:

- C Access a database and retrieve a row.
- C Access database and retrieve multiple rows.
- C Access a database and update a single row.
- C Retrieve all column names and data type for a single table.

8 Debugging

Objectives:

- C List the debugging tools available in the JAVA SDK.
- C Understand the concepts of debugging a JAVA program.
- C Use the supplied debugger or the IDE debugger

Exercises:

- C The instructor supplies two programs with bugs.
- C A comprehensive variety of tools will be used for locating all errors.

9 System Resources

Objectives:

- C Retrieving and setting system properties.
- C Forcing the system to do finalization.

Exercises:

- C Code a program to enumerate all system properties.
- C Add a property and retrieve the value.



10 Threads and Sockets

Objectives:

- C Code a multithread program.
- C Starting and synchronizing threads.

Exercises:

- C Write a program that performs a chat with another program. The programs could be local or remote.
- C Allow the user to have multiple chats.

