

Java Archive (JAR) Files

An Introduction

JAR Files

- Bundle together files, typically .class files
 - Uses zip file format
- Provides security and compression
- Provide support for versioning
- Portable

Manifest File

- JARs contain a *manifest* file
- The manifest file provides meta-information about the JAR file.
- You can modify this file to specify various characteristics of the JAR (e.g., dependencies on other JAR files, versioning, security).

name : value pairs

Creating JAR Files

- Java Archive Tool is part of the Java Development Kit (JDK)
- Create a JAR File:

```
jar cf jar-file input-file(s)
```

c is for “create”

f is for “file”

input-files is a space-separated list of files or use *

Example: `jar cf OutputJarName *.class *.jpg`

- There are other options (e.g., not compressing, v for verbose).
- IDEs also provide ability to create JARs
 - In eclipse, right click the class and select Export then follow the menu.

Creating Executable JAR Files

- Create a JAR File:

```
jar cfe jar-file DriverProgram input-file(s)
```

e is for the “entry point”- the name of the class you want to run-
the class with the main method

Extracting a JAR File

- View a JAR File:

```
jar tf jar-file
```

t is for “table of contents”

- Extract a JAR File:

```
jar xf jar-file
```

x is for “extract”

- Note: the JAR extraction will overwrite any existing files with the same name.

Running a JAR File

- Running a JAR File:

```
java -jar jar-file
```

Practice

- Create a JAR for text-based program and a GUI.
- Run the programs.
- Extract the contents.