

## The Java™ Tutorials

**Trail:** Deployment

**Lesson:** Packaging Programs in JAR Files

**Section:** Working with Manifest Files: The Basics

*The Java Tutorials have been written for JDK 8. Examples and practices described in this page don't take advantage of improvements introduced in later releases.*

### Modifying a Manifest File

You use the `m` command-line option to add custom information to the manifest during creation of a JAR file. This section describes the `m` option.

The Jar tool automatically puts a [default manifest](#) with the pathname `META-INF/MANIFEST.MF` into any JAR file you create. You can enable special JAR file functionality, such as [package sealing](#), by modifying the default manifest. Typically, modifying the default manifest involves adding special-purpose *headers* to the manifest that allow the JAR file to perform a particular desired function.

To modify the manifest, you must first prepare a text file containing the information you wish to add to the manifest. You then use the Jar tool's `m` option to add the information in your file to the manifest.

**Warning:** The text file from which you are creating the manifest must end with a new line or carriage return. The last line will not be parsed properly if it does not end with a new line or carriage return.

The basic command has this format:

```
jar cfm jar-file manifest-addition input-file(s)
```

Let's look at the options and arguments used in this command:

- The `c` option indicates that you want to *create* a JAR file.
- The `m` option indicates that you want to merge information from an existing file into the manifest file of the JAR file you're creating.
- The `f` option indicates that you want the output to go to a *file* (the JAR file you're creating) rather than to standard output.
- `manifest-addition` is the name (or path and name) of the existing text file whose contents you want to add to the contents of JAR file's manifest.
- `jar-file` is the name that you want the resulting JAR file to have.
- The `input-file(s)` argument is a space-separated list of one or more files that you want to be placed in your JAR file.

The `m` and `f` options must be in the same order as the corresponding arguments.

**Note:** The contents of the manifest must be encoded in UTF-8.

The remaining sections of this lesson demonstrate specific modifications you may want to make to the manifest file.