

Modules

Introduction to Modules

Java Platform Module System (JPMS)

- in practice, you often need to use libraries (code) written by others
- one of the way to make these libraries part of your program is by using JAR files
 - you have to make sure you have compatible versions of all libraries at runtime
 - this creates a complex chain of dependencies and minimum versions
- another way is to use *modules*
 - they provide groups of related packages with some set of functionalities
 - developer chooses which packages are accessible outside the module

Benefits of Modules

- **better access control**
 - you can have packages which are only accessible to other packages
- **clearer dependency manager**
 - if dependency is missing, Java will complain when starting the program
- **custom Java builds**
 - you can choose which parts of the JDK you need
- **improved security**
 - you only have to worry about the modules you use
- **improved performance**
 - comes with smaller Java package

module-info.java

- a module is a group of one or more packages plus module-info.java file
- contents of this file is called *module declaration*
- module declaration defines *dependencies*
 - the way one module relies on code in another one
 - these are defined using module declaration keywords

module-info.java : keywords

Keyword	Description
<code>requires <module></code>	Module depends on the code in another module.
<code>requires transitive <module></code>	If module A "requires transitive" module B, and module C "requires" module A, then C will have access to B
<code>opens <package> to <module></code>	Makes named package accessible to the named module in the runtime
<code>exports <package></code>	Public members of the named package are accessible to other modules.
<code>uses <service></code>	The module uses a service (i.e. interface) defined elsewhere.
<code>provides <service> with <implementation></code>	The module provides a concrete implementation of a service (i.e. interface).

Service Provider Interface (SPI)

Compiling and Running Modules

Compiling with javac		
Directory for class files	-d <dir>	n/a
Module path	-p <path>	--module-path <path>
Running the program with java		
Module name	-m <name>	--module <name>
Module path	-p <path>	--module-path <path>