Java 21 Complete Study Guide

Basics

- Compiled language
  - Source code (.java) → compiler → bytecode (.class) → JVM → machine code
  - Platform independent: "write once, run anywhere"
- Hello World
  - `public class HelloWorld { public static void main(String[] args) { System.out.println("Hello"); } }`
  - `public static void main(String[] args)` — entry point
  - `System.out.println(...)` — prints to stdout
- Variables & types
  - Primitives: `int`, `long`, `double`, `boolean`, `char`, `byte`, `short`, `float`
  - Reference types: `String`, `Integer`, `Object`, `int[]`
  - Declaration: `int x = 5;`
  - Assignment: `x = 10;`
  - Constants: `final int MAX = 100;`
  - Type inference: `var x = 5;` (local only)
- Operators
  - Arithmetic: `+ - * / %`
  - Comparison: `== != < > <= >=`
  - Logical: `&& || !`
  - Bitwise: `& | ^ ~ << >> >>>`
  - Assignment: `= += -= *= /= %= &= |= ^= <<= >>= >>>=`
  - Ternary: `condition ? a : b`
  - Instanceof: `obj instanceof Type`
  - Precedence: `() > ++ -- > * / % > + - > << >> >>> > < <= > >= > == != > & > ^ > | > && > || > ?: > = += ...`
- Control flow
  - `if / else if / else`
  - `switch` (classic, fall-through)
  - `switch` expressions (arrow form, yield)
  - `for (int i = 0; i < n; i++)`
  - `for (Type t : collection)` — enhanced for
  - `while (condition)`
  - `do { ... } while (condition);`
  - `break` — exit loop
  - `continue` — skip to next iteration
  - `return` — exit method
  - `throw` — throw exception
  - Labeled `break` / `continue`
- Arrays
  - Declaration: `int[] arr = new int[5];`
  - Literal: `int[] arr = {1, 2, 3};`
  - Length: `arr.length`
  - Access: `arr[0]`
  - Multi-dimensional: `int[][] m = new int[3][4];`
  - Jagged: `int[][] j = {{1, 2}, {3, 4, 5}};`
  - Copy: `System.arrayCopy(...)`, `Arrays.copyOf(...)`
- Strings
  - Immutable
  - Literal: `"text"`
  - Concatenation: `+`
  - Escapes: `\n \t \r \" \\ \uXXXX`
  - Text block: `"""multi line"""`
  - Comparison: `.equals(...)` not `==`
  - Common methods: `length()`, `charAt(i)`, `substring(...)`, `indexOf(...)`, `contains(...)`, `startsWith(...)`, `endsWith(...)`, `trim()`, `toUpperCase()`, `toLowerCase()`, `replace(...)`, `split(...)`, `isBlank()`, `strip()`, `stripLeading()`, `stripTrailing()`, `repeat(n)`
  - Immutable: `toUpperCase()` returns new string
- Wrapper classes
  - `Integer`, `Long`, `Double`, `Boolean`, `Character`, `Byte`, `Short`, `Float`
  - Autoboxing: `int → Integer`
  - Unboxing: `Integer → int`
  - Caution: `==` compares identity, `.equals()` compares value
- Input / output
  - `System.out.print(...)`
  - `System.out.printf("%d %s %f", ...)`
  - `System.out.flush()`
  - `System.in`
  - `Scanner sc = new Scanner(System.in);`
  - `sc.nextLine()`, `sc.nextInt()`, `sc.nextDouble()`, `sc.hasNext()`, `sc.close()`
- Comments
  - `// line comment`
  - `/* block comment */`
  - `/** Javadoc */`
- Modules
  - Module system (JPMS)
  - `module-info.java`
  - `module name {}`
  - `requires other.module;`
  - `requires transitive other.module;`
  - `exports package.name;`
  - `exports package.name to target.module;`
  - `opens package.name;`
  - `opens package.name to target.module;`
  - `provides Service with Impl;`
  - `uses Service;`
  - Module path: `-p mods`
  - Compile: `javac -d out -p mods -m module.name`
  - Run: `java -p mods -m module.name/main.class`
  - Inspect: `jdeps -d -m module.name`
  - Create JAR: `jar --create --file app.jar --main-class main.class -C out .`
  - Run JAR: `java -jar app.jar`
  - Module JAR: `jar --create --file app.jar --module-version 1.0 -C out .`
  - Describe: `jar --describe-module --file app.jar`
  - Jlink: `jlink --module-path mods:$JAVA_HOME/jmods --add-modules module.name --launcher name=bin/run --output dist/runtime`
  - Jpackage: `jpackage --name App --module-path mods --module module.name/main.class --type app-image --dest dist`
  - Jpackage installer: `jpackage --name App --module-path mods --module module.name/main.class --type deb --dest dist`
  - Jpackage types: `app-image`, `deb`, `rpm`, `msi`, `exe`, `pkg`
  - Jpackage options: `--app-version 1.0`, `--vendor "Acme"`, `--description "..."`, `--icon icon.ico`, `--runtime-image dist/runtime`, `--jvm-options "-Xmx512m"`, `--arguments "..."`, `--win-console`, `--win-dir-chooser`, `--win-menu`, `--win-shortcut`, `--linux-shortcut`, `--linux-package-name name`, `--linux-app-category utility`, `--linux-deb-maintainer email`, `--linux-rpm-license-type GPL-3.0`, `--mac-package-identifier com.acme.app`, `--mac-sign`, `--mac-signing-key-user-name "..."`, `--mac-app-store`
  - Module path vs classpath
  - Automatic modules
  - Unnamed module
  - Reflection across modules
  - Service loading: `ServiceLoader.load(Service.class)`
  - Multi-release JAR: `META-INF/versions/9/...`
  - Debugging: `-Dlog.level=debug`, `-Dtrace.modules=true`, `-Dtrace.classloading=true`, `-Dtrace.resolution=true`, `-Dtrace.reflection=true`, `-Dtrace.jpms=true`, `-Dtrace.jar=true`, `-Dtrace.jlink=true`, `-Dtrace.jpackage=true`, `-Dtrace.all=true`
  - Trace output: `[trace:module] ...`, `[trace:classloading] ...`, `[trace:resolution] ...`, `[trace:reflection] ...`, `[trace:jar] ...`, `[trace:jlink] ...`, `[trace:jpackage] ...`
  - Module graph: `jdeps -d -m module.name`
  - Module dependencies: `jdeps -d -m module.name -v`
  - Module summary: `jdeps -d -m module.name -summary`
  - Module JAR: `jar --create --file app.jar --module-version 1.0 -C out .`
  - Describe module: `jar --describe-module --file app.jar`
  - Module path: `-p mods`
  - Module name: `-m module.name`
  - Module version: `--module-version 1.0