Module 133.1.1.2 :: OOP Concepts :: Modularity

Modularity

Outline

- What is Modularity
- Classes vs Modules
- Java Packages
  - Overview
  - Files and packages
- Coupling and Cohesion

Reading Material

1. For a broad design focused view of this topic, take a look at 1.7 in **Object-Oriented Design & Patterns (https://www.horstmann.com/design_and_patterns.html)**
2. For a detailed look at creating packages, take a look at the **Oracle tutorial on packages (https://docs.oracle.com/javase/tutorial/java/package/index.html)**

Practice Exercises

There are no practice exercises for this Module. We will return to much of the detail of this chapter in later modules.

Video Lecture

In this video I briefly go over the general concepts of modularity, coupling, and cohesion. I also talk about Java packages. We will be getting cozier with packages this semester than you may have in the past.

Alternate Video Lecture

There are two short video lectures below on different aspects of this topic.

On Using and Creating Packages

Rex Jones II (https://www.youtube.com/channel/UCBU_TzaPsDXXeQ3ItUYKaJg) gives a no-nonsense introduction to package names and how to create you own packages in Eclipse. The process is similar in IntelliJ. What I want you to keep in mind is that there is nothing special about using an IDE in creating packages. They simply create the associated directly structure for you. You can easily create you own directly structure and then use the package name convention to access the various subdirectories, hence, sub-packages.

Modularity and *Modules* in JDK9+

Modularity in this section should not be confused with the new concept of ***modules* available as part of the language in JDK9+.** **(https://blogs.oracle.com/java/modular-development)** Modules in JDK9+ provide a new mechanism that allows the developer to more precisely control access to packages within the *module*. If you're still trying to get packages down, then feel free to ignore all of this, but, if you're interested in a little more depth on this new feature, then check this video from Alex Buckley, a maintainer of the Java specification.