**Dependency Management**

1. **Dependency management** 
   1. Modem software almost never built from scratch
   2. Built on top of other packages written by other people
      1. These packages are often reliant on other packages
         1. This complexity is why we need dependency management, to manage all the mess
   3. “Import requests”
   4. **Terminology** 
      1. **Package or library** 
         1. Self-contained unit of code usually designed to be reusable and accomplish a well-defined range of tasks
      2. **Dependency** 
         1. A relationship between two packages in which one package requires is dependent on) another in order to do what it does
      3. **Package manager** 
         1. A type of utility software that keeps track of
            1. Which packages have been installed
            2. What their dependencies are
            3. Whether of not dependencies have been installed
            4. The versions of each package that are installed
   5. **Example packages** 
      1. Making http requests
   6. **Package managers** 
      1. A picture containing text, scoreboard, plaque, screenshot

         Description automatically generated
   7. **Choosing a package** 
      1. Web search, stack overflow,
      2. When was it written, how often is it updated, how many downloads,
   8. **Where are packages stored**
      1. **Globally** 
         1. Specified location on he system where all programs written in a given language have access to them
      2. **Locally** 
         1. Downloaded packages are stored in a subdirectory of the current software project and are only accessible by that one project