# Examining New Code

Things to Consider

* When you are reading someone else’s code and you find a keyword or package you are unfamiliar with implementing, what helps you best understand how to use it?
  + It helps to read the Java documentation on Oracle, GeeksforGeeks, or StackOverflow.
* What do you feel is most often missing from online tutorials or the documentation available through Oracle?
  + These resources are great if you want to see how to use a certain Java feature, but they don’t always teach you what’s happening “behind the scenes” in the code. So you can learn the “how”, but not the “why”.
* What is the point of understanding how introductory java utility packages are crafted before implementing those packages and their methods?
  + Like the previous bullet, this helps the user understand the “why” when they’re using Java utility packages. If you experience an issue or the package isn’t working the way you expected, it helps to know what Java is actually doing within that package so you can better troubleshoot the issue.

Synthesized List

1. Read the driver file to get a general understanding of the program.
2. Read the other files. Identify any packages, keywords, methods, or syntax you don’t understand.
3. Check online documentation to complete any gaps in knowledge.
4. Reread the code. Makes sense now?
5. If not, comment out portions of the driver file to see if it helps break down the code.
6. Run the code in full.

# Create your Plan

Proposed Hierarchy

1. Organization Interface
   1. Organization
      1. Employee Interface
         1. Generic Employee
            1. Supervisor

Non-Manager Employee

Internal Employee

Contract Employee

A diagram of a company

Description automatically generated