Java

Background of Java

* Sun Microsystems was analyzing why large software projects fail
* They wanted a language that would help address these problems
* Their key goal was Scalability
  + A lot of the aspects of Java programming language can be attributed to this goal
* Key Features Aspects
  + Java has a JVM
    - Java Virtual machine means that your application can execute the same on one computer vs another
    - Incredible portability and range of devices
  + OOP centric language (OOP is very modular and reusable)
  + Java is a compiled language
    - You write your human readable source code that gets turned into machine friendly bytecode for execution
  + Strongly and statically typed
    - Strongly typed means there is a lack of implicit type coercion
      * For example you cannot add 100 + “hello”
    - Statically typed languages force you to declare the type of a variable.
      * int x = 0 or String y = “”
  + Java does have primitives
    - Java is 99% OOP (has primitives and Lambda Java 8)
    - Primitives are more efficient than objects for operations
  + Java has automatic memory management. NO POINTERS!!!!
    - Pointers were one of the main causes or programmer error in applications
    - Java has its own garbage collection where objects that are no longer reachable in the program are marked for deletion
  + Java is 100% pass by value.
    - You pass in a copy of the value of the variable
  + Java is a multi-threaded
  + No multiple inheritance

Java Compiling

* You write human readable source code in a .java file
* Something.java compiles into a Something.class file
* The .class file is the machine readable Java Bytecode
* If you need to send a Java program to someone you only need to send them the bytecode.

Git

* Source code Version Control
  + - A ‘filing system’ that allows you to keep track of changes and edits to the code you are working on
      * Do not want to lose the code you have written. Laptop crashes or breaks.
      * You might want to go back to an earlier version of the project.
        + You have some new bug in your code that did not happen in a previous version
      * Allows you to share and merge your code with others. Vital in large applications where everyone is working on a single part of it.
* Keywords
  + - Repository – The bookshelf or location where you are storing all of the code. And notes of all the changes to that code.
      * Remote repository – exists on the web (github)
        + Every developer can access the remote repository
      * Local Repository - exists only on your machine
        + Only you can access your physical machine
    - Branch – Splinter off the main branch or source code