**Unit 2 Lesson 1.01**

Algorithmic: A set of instructions for completing a task.

Algorithmic thinking: Break a task into step by step instructions

Pseudo code: A technique for describing an algorithm.

* Compact: tells what to do but not overly detailed
* General: No specific set of numbers so any number can work
* Abstract: Any coding language can work with these instructions.

**Unit 2 Lesson 1.02**

Java API: Documentation for all of Java’s Classes

Packages: A group of related classes

[Package Frame](javascript:void(0);): The top left frame initially lists all the packages in the Java API. A package is a group of related classes.

[Classes Frame](javascript:void(0);): The bottom left frame initially lists all the classes in the Java API. Once a package is selected, only the classes in that package are shown.

[Documentation Frame](javascript:void(0);): The larger frame to the right list’s brief descriptions of each Java API package and gives detailed information about a package or class once it is selected in the other two frames.

Fields: Class Constants

Constructors: Create Objects that give you access to the methods of a class

Methods: contain pre-written, verified code for completing tasks. For example print() and println() are methods of print class.

String class: Represents all characters springs, anything such as abc are implemented as instances of this class.

**Unit 2 Lesson 1.03**

Pow(): Raises a value to a power, such as Math.pow(2,15).

Exercise 75

1. I think the black box is square rooting
2. Only accepts positive real numbers and unacceptable numbers cause Nan
3. If not number results in NaN

Math.sqrt(): Square roots input expects input to be double

.equal() for object

== primatives

Primitive data type: Boolean, int, char, double.

.compareTo(): compares each character in the string and return the difference