Vocabulary Chapter 04

Scope: the part of a program in which a variable exists and can be accessed using its unqualified name

( The variable’s block )

Override: variable with the same name that takes precedence over another variable

**Shadowing:** the action that occurs when a local variable hides another variable with the same name that’s further away in scope

\*\*\*\*Variable’s within methods of a class override / hide the class’s field variables

Overloading: using one term to indicate diverse meanings, or writing multiple methods with the same name but with different arguments

Ambiguous: a situation in which the compiler cannot determine which method to use

Reference: an object’s memory address

**this** Reference: reference to an object that is passed to any objects non-static class method

~Class Methods: static methods that don’t have a ‘this’ reference bc there’s no objects associated

~Class Variables: static variables shared by every instantiation of a class

NaN: not a number

Package: library of classes : folder that provides convenient grouping of classes

~Fundamental Classes: basic classes contained in the java.lang package that are automatically imported

~Optional Classes: resides in a package that must be explicitly imported into the program

Enumeration: a data type that consist of a list of values

Composition: describes the relationship between classes when an object of one class is a data field within another class

Top-Level Class: the containing class in nested classes

~Static Member Class: type of nested class that has access to all static methods of its top-level class

~Non-Static Member Class (Inner Class): nested classes that require an instance

~Local Classes: type of nested class that are local to a block of code

~Anonymous Classes: nested, local classes that have no identifier