Vocabulary - Chapter 11

Concrete Classes: non-abstract classes from which objects can be instantiated

Abstract Classes: one from which you cannot create any concrete objects, but from which you can inherit

Virtual Classes: the name given to abstract classes in other programming languages, such as C++

Abstract Method: is declared with the keyword ‘abstract’ & has no body; **a subclass must override a base class abstract method**

Non-Abstract Method: a method that is inherited

Dynamic Method Binding: the ability of an application to select the correct method during program execution AKA - Late Method Binding

Static / Fixed Method Binding: the opposite of dynamic method binding; it occurs when a method is selected when the program complies rather than when it is running

**Ad-hoc Polymorphism:** occurs when a single method name cane be used with a variety of data types because various implementations exist 🡪 **another name for Method Overloading**

Pure / Inclusion Polymorphism: occurs when a single method implementation can be used with a variety of related objects, because they are objects of subclasses of the parameter type

Object Class: defined on the java.lang package that is imported automatically into every program you write; EVERY JAVA CLASS DESCENDS FROM THE OBJECT CLASS

Hash code: a calculated number used to identify an object

Multiple inheritance: the capability to inherit from more than one class

Interface: looks much like a class, except that all of its methods must be abstract and all of its data (if any) must be static & final; it declares method headers, but not the instructions / implementations within those methods

Java Archive (JAR) file: compresses the stored data

Default packages: the unnamed one in which a class is placed if you do not specify a package for the class

Collision: a term that describes a class naming conflict