**Sep 11th, 2022**

* Java is highly needed within industry.

* **Demand is high**  for developers within java.
* Java is a great **object oriented programming language.**

* Java was created to handle data through the use of object oriented structure. This approach keeps things clean, and fast development.
* Classes are blueprints for objects. Classes define characteristics of what each object should have and what it can do.

**Introduction into Class analogy:**

* Classes again are like blueprints.

* There can be classes created from one original class.
* This terminology is known as **SUPER CLASSES, AND THEIR SUBCLASSES**
* This base concept allows for the reusability of code.

September 12th – 2022 notes

* A compiler is a program that can take Java code and then convert into bytecode, which is something only the machine can understand.

* Each machine that you write code in whether it be a windows machine, mac machine, or Linux machine… the Java virtual Machine acts as a interpreter that will allow for code to be written from Java code to bytecode the OS can understand.

**Beginning to actually write in Java:**

* **Java language specification**: documentation for following specific rules for Java development. Things like: use an asterisk to multiply two numbers, put a parenthesis after this etc.

* **API specification:** Thousands, and thousands of specific words used within Java. Words like: listening to button clicks, query databases, etc.

**Understanding A simple Java program:**

* Everything in Java will have to be written as a class.
* No programs will be ran if they are not in a class form.
* Classes will have their attributes and their methods defined within them. (refer to code within netbeans)

**Looking into Object-Oriented-Programming!**

* **“What it means to be an account”** for the code example a pretend bank account class will be created.

* A class will contain it’s fields and methods.
* Data hiding revolves around the keywords: **default, public, protected, private.**
* For practise purposes variables within the code, access will be set to default. This is not actual practise, be sure to use encapsulation when necessary.
* Packages organize things within Java. **Java.base**  will be one of the main modules.
* Refer to code for first basic object oriented practise

SEPTEMBER 17th note review:

* Understanding as to why Java is highly needed.
* Concept of Object oriented programming provides faster code development, and reduces the use of repetitive coding.

* Basic class structure understood.
* How to create fields understood
* How to print objects created from classes understood
* How to use dot-notation to assign values to class fields understood.

Move forward

**September 17th, 2022**

* Within a methods brackets will be parameters.
* Parameters allow methods to become more versatile.

* The method with the parameter can be defined inside the main class of whatever that is being made. The method will be **public, have it’s data type, name starting with get, and then a data type for the parameter. There can be more than one parameter within a get method.**
* Data types for get methods must be passed as they are indeed returning a value.
* Organization with tax amount to be paid JAVA example created.
* Methods with parameters understood to this point. There can be multiple methods with different parameters. Its just a matter of more practise.

**October 2nd 2022 – Revisitation of concepts previously learned, too much time slipped, need to revisit OOP fundamentals.**

* The beginning sections of this chapter reviewed the concepts of using classes. Declaring fields, and then creating objects of the class within the class with the MAIN method.
* For practise code to refresh my memory I will be writing the example that wants to have an organization class written.

**Basic concepts of classes and objects revisted and thoroughly understood.**

**Data hiding** – (**Encapsulation.**)

* Encapsulation is a term where data is hidden from being accessed from outside classes.
* This is good practise as it allows for no errors to occur.
* **Within the example provided, the fields for the account class was set to private, and set methods passed one parameter were set the values of the fields.**
* **There were then get methods provided.**
* The example practised visits the organization class again but incorporates data hiding.