When it comes to Sets and Maps in Java, they have some similarities and differences. They both are interfaces in the collection framework, and they store collections of objects as a wider overview. Maps are stored in key value pairs and this is different from a Set as it is unordered and contains different elements. Duplicate items will be ignored when it comes to Sets. There are HashSet, LinkedHashSets and TreeSets. All unique to each other. The feature of providing only unique values is an advantage at times. The biggest to a HashSet is that it doesn’t maintain its insertion order. Whereas a LinkedHashSet does. Knowing when to use each is very beneficial.

The main way we create a Set Object is similar to our ArrayList. It can look like:

Set<String> set = new HashSet<String>();

There are HashMaps, LinkedHashMaps and TreeMaps. A Map is a way of storing objects a key value pair. Each key is linked to a certain value and once these are established, we can access our values by using only our values if we choose. When we create a Map Object it is important to note that we must declare both the key and value types. It looks similar to our Set with a subtle but important difference. A major advantage to a HashMap is you have fast access to elements due to hashing and that it allows for key value pair insertion.

Map<Integer, String> map = new HashMap<Integer, String>();

Here you can see that we have declared an Integer Type for our Key and String Type for our value.