CSC-123

Dictionaries / Maps Summary

By Blake Fitzpatrick

Maps are a form of data structure in java. They serve as an extremely organized and efficient way of storing and retrieving data from a single object. Maps can store just about every primitive data type, and do so by ordering elements in key-value pairs. However the keys must be unique and non-repeating. This organization allows for an easy search of a specific key in the Map.

Maps provide two interfaces and three Map classes that the classes can utilize the interfaces to perform their own special functions. The three Map classes are HashMap, LinkedHashMap, and TreeMap. The HashMap implements Map, but does not maintain the order. The LinkedHashMap also implements Map, but it inherits the HashMap class and retains its order of insertion. Finally TreeMap implements Map and SortedMap, which maintains ascending order.

As with many classes in java the Maps class is home to tons of methods. Each method provides additional functionality to how Map objects interact. A very useful method in the Maps class is the “containsKey(Object)” method. This method checks whether a key value of an element is being mapped into the Map, and returns true if found. Another method is the “get(Object)” method. This is used for retrieving the stored value in a key, and returns null if the Map doesn’t contain the key. Another method I found useful was the “put(Object, Object)” method. With this method you can input a new value with the specified key. Another method is the “remove(Object)” method, which is used for removing a value at a specified key. One last method that I use is the “size()” method. This method returns the number of key-value pairs in the entire Map.

Although both classes are useful in their own right, I would choose the HashMap if I didn’t need my Map in order or if there arent alot of key-value pairs. I would use TreeMap when I needed my elements in ascending order.