I want to write an inventory program. This will read in lines from the user that are formatted as

<object> <count>

For example,

hammer 23

This indicates that we have found 23 hammers. If we see another line

hammer 5

then we need to update our count of hammers to 28.
What we are doing here is associating Strings with ints. We could do this by keeping a list of strings and a corresponding list of ints but that would be a pain to maintain. In Python you would use a Dictionary to hold this data. In Java such an associative structure is called a map.
The Collections Framework has two map structures -- HashMaps and TreeMaps. We will implement both this semester, so you'll be able to see the differences. For now we'll just do this with a HashMap.
You can think of HashMaps in the way you think of dictionaries in Python -- they associate values to keys. The *key* field is like an index; in our inventory program it will be the name of the item, like "hammer". The *value* field is the value being associated with the corresponding key; in our program this is the count of how many such items we have.
HashMaps take two class parameters -- one class for the key and one for the values, as in
   HashMap<Key, Value>
For our inventory program the structure is
   HashMap<String, Integer>

We construct our inventory structure with

   HashMap<String, Integer>inventory = new HashMap<String, Integer>();
There is a \textit{put} method for inserting into the HashMap:

\begin{verbatim}
  inventory.put("hammer", 5)
\end{verbatim}

associates the number 5 to the string "hammer".
Similarly, there is a *get* method for finding the value associated with a particular key:

```python
inventory.get("hammer")
```

tells you the number associated with string "hammer".
Just as with Python, there is a runtime error if you try to get the value associated with a string that is not one of the current keys, so it is necessary to check that something is a key before using it as an argument to get. After getting name and count values from the input, here is the code for updating the HashMap:

```java
if (inventory.containsKey(name)) {
    int current = inventory.get(name);
    inventory.put(name, current+count);
} else
    inventory.put(name, count);
```
In Python there is a method for obtaining a list of the current keys in a HashMap; in Java the keySet() method gives a set of the current keys. You can't iterate through sets with a for-loop the way you can with lists, but you can obtain an iterator structure that will do this. You will implement iterators in Lab 4. Here is the code for using an iterator to print the data in our HashMap:
Set keys = inventory.keySet();
Iterator<String> iter = keys.iterator();
while (iter.hasNext()) {
    String name = iter.next();
    Integer count = inventory.get(name);
    System.out.printf( "%-10s %3d \n", name, count);
}