**Streams Labs**

1. Given a list of numbers (9, 10, 13, 76, 8, -9), stream the list, filter and count the numbers that are larger than 10.
2. Stream a list of names (“Andrew”, “Brian”, “Charlie”, “Aaron”, “Ed”), map to uppercase, filter and count the names that start with “A”.
3. Given a list of temperatures (98.4, 100.2, 87.9, 102.8), count how many are > 100.
4. Given a *Map* that maps dog names to ages, figure out how many dogs are older than 4.
   1. Note that a *Map* is not a *Collection* (i.e. *Map* is not in the *Collection* hierarchy); whereas a *Set* is a *Collection*. The *stream()* method is a *default* method in the *Collection* interface and thus all implementing classes inherit it. To convert the *Map* to a *Set* so we can stream it, we can use the *entrySet()* method i.e. *map.entrySet().stream()*
   2. “Spot” is 1; “Rex” is 4 and “Charlie” is 8.
5. Stream a list of names (“Andrew”, “Brian”, “Charlie”, “Aaron”, “Ed”, “Sean”), map them to uppercase and using *anyMatch*, see if any of the names begin with “S”. If so, count how many names begin with “S” (note: you will need to re-create the stream).
6. Create a class Purchase which has a *String* *item* and a *double* *price* instance variables. Note that *price* is a primitive and not a wrapper class. Code a constructor that enables you to initialise both instance variables based on the arguments passed in.  
   Stream a list of *Purchase*’s : a “Shirt” costing 100.1; “Shoes” costing 149.9 and a “Jumper” costing 90.3. Map to a stream of *double*’s (the *price*’s) and get the average of the price’s that are between 100 and 200.
   1. change the filter to between 10 and 20; does your code still work?