# Week 05: Coursework

## Java Advanced

1. Fork and clone the following Github repository.

<https://github.com/busyQA-java-developer/busyqa-05-java-3-coursework-initial.git>

1. Import the Gradle project into Eclipse.
2. There are several exercises in the repository. Please read the following instructions to complete the Java programs:
3. *Exercise CollectionClass:* This program reads the file named *colors.txt*, which resides in this project's *src/main/resources* folder. That file contains colour names that are populated into a string ArrayList. (Already coded in the Util class). Write code to remove duplicate elements in the ArrayList (hint: use a HashSet). Then sort the ArrayList with a Low time and Space Complexity algorithm. (hint: use the sort() method from the built-in Collections class). Finally, search for the string “Blue” and print out its position in the ArrayList. (Hint: use the binarySearch() method from the Collections class).
4. *Exercise GenericClass:* Write a generic class called AnimalHouse with a generic property of type E called animal. The class must have getter() and setter() methods for the property, and a method called *printAnimal*() that prints out the property value on the console using the *System.out.println()* method. Uncomment the block of code indicated in the main() method and run the program.
5. *Exercise GenericMethod*: Write a static generic method that prints out all the elements of a Collection on the console. Then use the method coded to print out the elements of a HashSet and an ArrayList.
6. *Exercise Maps:* This program reads the file named *pairs.txt*, which resides in this project's *src/main/resources* folder. That file contains pairs of values id(Integer) and name(String) names populated into a HashMap. (Already coded in the Util class). Write code to print out all the Hashmap’s entries sorted by Key on the console. Hint: Copy the HashMap entries into an ArrayList. Then sort the ArrayList by the entry’s key using a Comparator class. A more efficient way is to use a TreeMap (A Map sorted by key). However, we didn’t cover TreeMaps in the course, so that this solution may need additional research.
7. *Exercise MapsWithLambda*: Modify your previous solution of the Maps exercise to print the map entries with a Lambda expression. (Hint: Use the forEach() method of the Map Interface)
8. After you finish the five exercises, please commit and push your changes to your GitHub repository.
9. You can find the solutions for the exercises above in the Github repo:

<https://github.com/busyQA-java-developer/busyqa-05-java-4-coursework-complete.git>

// End