The purpose of this set of exercises it recap some of the concepts seen so far during the course.

Exercise 1 – Generics (Optional)

Open the es1 folder in your IDE. It contains a class hierarchy of beers and a list containing different type of beers. The goal of this exercise is to implement the following methods:

a) An *add* method that accept a list of beers, a Class parameter *type* and a Collection of Beer *target.* The target collection will be populated with all beers that match the given parameter type. Use **bounded wildcards** for the collections and type.
b) a *filter* method that accepts a list of beers and a Class parameter *type* and returns a new list of beers containing only beers of the specified class type.
c) a **generic** method *filterGeneric* that accepts a list of beers and a Class parameter *type* and returns a new list of beers containing only beers of the specified type. For this method the returned list must be a specific list of the same type of Beer that has been passed as parameter.

```
final List<Beer> beers …;
add(beers, Weisse.class, addedWeisse);
final List<Beer> trappists   = filter(beers, Trappist.class);
final List<PaleAle> paleAles = filterGeneric(beers, PaleAle.class);
```

Hint: use java's reflection to verify if an element of the list is the same type as the method's parameter.

Exercise 2 – ChangeLog annotation (Optional)

Implement the *ChangeLog* annotation to track changes on classes, interfaces, fields, methods and constructors. It should be possible to specify the type of change (ADDED, FIXED, CHANGED, REMOVED), a brief description text and the version in which the change has been performed. While developing, take into account that multiple annotations can be added e.g. to a method to track all major changes.
Finally, write the logic to generate a report, where all changes are grouped by the version and type:

```
# Changelog for assignment08.assignments.es2.Coordinate

## Version 1.0.0
### Added
 - first implementation [assignment08.assignments.es2.Coordinate]

## Version 1.0.1
### Fixed
 - Distance computation [assignment08.assignments.es2.Coordinate.distance]

## Version 1.0.2
### Removed
 - compareTo implementation [assignment08.assignments.es2.Coordinate]

## Version 1.0.3
### Changed
 - added final [assignment08.assignments.es2.Coordinate.lat]
 - added final [assignment08.assignments.es2.Coordinate.lon]
```

Exercise 3 – Streams (Optional)

Open the source code es3/Es3.java in your IDE. It contains an example on how to obtain a Stream of Characters from a String. The goal of this exercise is to use Java's Streams API to:

- Find the number of unique characters
- Find the number of letters and non-letters
- Check if there are any numbers (digits)
- Number of occurrences of each Character, sorted by the character
- 10 most common letters
- Find the most common Vocal