# **Foundations Week Day 2 - Exercises**

### **Non-Technical Questions**

* What would you say are your greatest strengths?
* What are your greatest weaknesses?
* What would your colleagues say about you?
* Tell me about a colleague you've worked with and where you gave them some constructive feedback? How did you go about doing this?

### **Theory Questions**

* What are exceptions in Java and what are they used for?
* In Java must you always have a catch block when you have a try? If not, what would be a use case for this?
* Must you always have a finally block if you have a catch? If not, what would be a use case for this?
* Can you have a finally block without a catch block? If not, what would be a use case for this?
* What is the difference between a checked and an unchecked exception?
* What is the difference between an error and an exception?
* What is the difference between the throw and throws keywords?

### **Practical Questions**

* **Most common element in list:** Write a function, using your character count function from yesterday that returns the most commonly occurring element
  + Extension: Do this exercise using Streams
* **Removing elements:** Using your function above write a function that removes every item to the right of each of the most common elements in the list, but do not remove the element if it is also the most commonly occurring value
  + Extension: Again do this exercise using Streams

### **Working with user requirements**

Here's a high-level overview of requirements for a banking system:

1. Create a banking system that allows users to perform transactions;
2. Users should be able to create accounts and deposit or withdraw money;
3. The account should have a front-end written in Angular and a back-end written in Java

Come up with a list of extra questions you would ask in order to flesh out the above requirements. Make sure your questions are clear and that it is clear *why* you are interested in this question where necessary

### **Working with longer user requirements**

Imagine you're working on a Java application for a company that sells various types of beverages and food. The company has a wide range of products, including cakes, breads, coffee, tea, soda, and energy drinks. They have recently decided to expand their product line to include a new type of fruit juice. Key information about the coffee is the beans that have been used, for cakes, the number of calories and sugar is important. This is also the case for soda and energy drinks. For fruit juice along with these, the type of fruit used and the percentage of real fruit is important. The marketing team has been working hard to come up with a catchy name and an attractive packaging design for the new product. They have also conducted extensive market research to determine the best price point and target audience for the new juice.

As a developer on the IT team, you have been asked to add support in the system for the fruit juice. The company's IT department has been tasked with updating the inventory management system to accommodate the new product. The system is responsible for tracking the stock levels of all products, as well as managing the ordering and delivery processes. The system is written in Java and uses an object-oriented design with classes representing different types of beverages and their properties.

The company is excited about the launch of the new fruit juice and is looking forward to seeing how it performs in the market. They believe that the combination of a great product and an effective marketing campaign will help them achieve success with this new addition to their product line.

* Taking the requirements above, extract the key requirements for *your* task and briefly describe any class(es) you will create based on these requirements, e.g. variables, methods
* Is there any missing information you need in order to complete your task? If so, what questions would you ask to get it?