Scenario 1

Considering I have a CSV file, represented as above:

Retailer	Currency	Price
ABC	USD	4.00
ZXW	EUR	3.77
LMN	GPB	3.50

First I would convert all prices to a single currency, Euro or US Dollar, and store that value in a new column. To do that, I would like to know how many different currencies there are. If not too much, maybe less than 10 currencies, I would manually find the exchange rate for the all currencies, save them to a file, and create a script to add that new column based on the retailer's price, its currency, and the correspondent exchange rate. The final CSV file would be like this:

Retailer	Currency	Price	Price EUR
ABC	USD	4.00	3.80
ZXW	EUR	3.77	3.77
LMN	GPB	3.50	3.65

I would use Java to write the script because it is the language I am most used to, but considering I would be doing a lot of these data manipulation tasks, I would definitely start to do it in Python, which has a better support and libs for this type of tasks.

In the second part, I would try to find valuable information on the dataset, starting with some basic aggregate functions like average, minimum, maximum, standard deviation, and median values.

It also would be interesting to produce some Top Lists, like Top 5 Max Price or Top 5 Min Price. Considering there are some retailers with more than one price listed, it would also be interesting to create per Retailer reports, counting how many different prices each retailer has, and to generate all the aggregate functions information.

Considering it's a huge list, I would consider using something like Apache Spark or Kinesis from AWS. Although, taking into account it's a list for only one product, I guess it might not be a huge list, so I could again use Java or Python to generate all that information.

Scenario 2

To identify differences between the two descriptions I would have to research the topic due my lack of experience in this type of problem. From previous experience, I know there are some libs to help with similarities like Apache Commons Text, and I would love to find one that fits the task, either in Java, or Python, or C/C++, or a CLI application.

But I understand I should come up with my own implementation. In this case, as a simple comparison, I would iterate both character sequences and when found a different character, I would advance the pointer only in one sequence until the same character is found, and keep a counter of how many different characters there are. At the end, I can calculate a similarity rate by dividing the different characters counter by the length of the first sequence.

A better approach to not using an already existing library, would be to implement a well known algorithm for that problem. In a fast Google search, I was able to list at least four of them.