Hokodo - Data Science Test

For the following exercises, the readability of the code will be taken into account.

Exercise 1 - Python

Please solve this in Python 3. Feel free to use any (standard) library that you find helpful.

Write a function that computes the frequency of the words from the input string. The output should be sorted alphanumerically by word.

Please also write tests for this function, and remember that docstrings are an important part of code readability.

Suppose the following input is supplied to the program:

```
“How much wood would a woodchuck chuck if a woodchuck could chuck wood? A
woodchuck would chuck as much wood as a woodchuck could chuck if a
woodchuck could chuck wood.”
```

Then, the output should be:

```
[('A', 1),
 ('How', 1),
 ('a', 4),
 ('as', 2),
 ('chuck', 5),
 ('could', 3),
 ('if', 2),
 ('much', 2),
 ('wood', 2),
 ('wood.', 1),
 ('wood?', 1),
 ('woodchuck', 5),
 ('would', 2)]
```

Exercise 2 - Pandas

You have been provided with data in two CSV files: `pokemons.csv` and `battles.csv`.
The `pokemons.csv` file contains information about Pokemons. It is indexed by the Pokemon number, which is unique to each Pokemon. The index is on the first column of the csv.
The `battles.csv` file contains a list of battles between two Pokemons, registering the winner.

We want to build a predictor of which Pokemon would win a given match given its statistics and the statistics of its opponent. You don't have to write any tests for the code you write for this exercise.

1. Build a DataFrame containing all necessary information to train this predictor. This DataFrame should contain at least the following statistics for both Pokemons and types of both Pokemons: HP, speed, attack, defense, Sp. Atk, Sp. Def. Feel free to include any other information you think is useful.
2. Train a model to predict which Pokemon will win given two Pokemon and their statistics. Please provide details on your thought process. A good way to present the result of your work in parallel with comments is to use a Jupyter Notebook, but you can use another tool if you prefer. In particular, we are interested in how you would evaluate the quality of the model(s).
3. [Bonus round] If you were to deploy this model, what additional steps would you take?

Exercise 3 - Modelling

This exercise does not require any code to be written. We will discuss these questions during the technical interview, and you don't have to send us back any answer beforehand.

We can get public data for all companies in a country at a given date, for a cost. This includes information about the characteristics of a company, such as the sector it is operating in, its filed financials information and the number of employees, as well as whether the company filed for insolvency. Typically, around 1% of companies become insolvent every year.
We want to build a model predicting the probability for a given company to become insolvent.

1. Which metrics would you use to evaluate the model? Which would you avoid? Why and why not?
2. We want to reduce the cost of acquiring the dataset by selecting a sample of the companies, instead of taking all of them. How would you select the companies? What would you consider?
3. How would you build a dataset for evaluating the model? When would you do it?
4. Any other comments / ideas you would like to explore