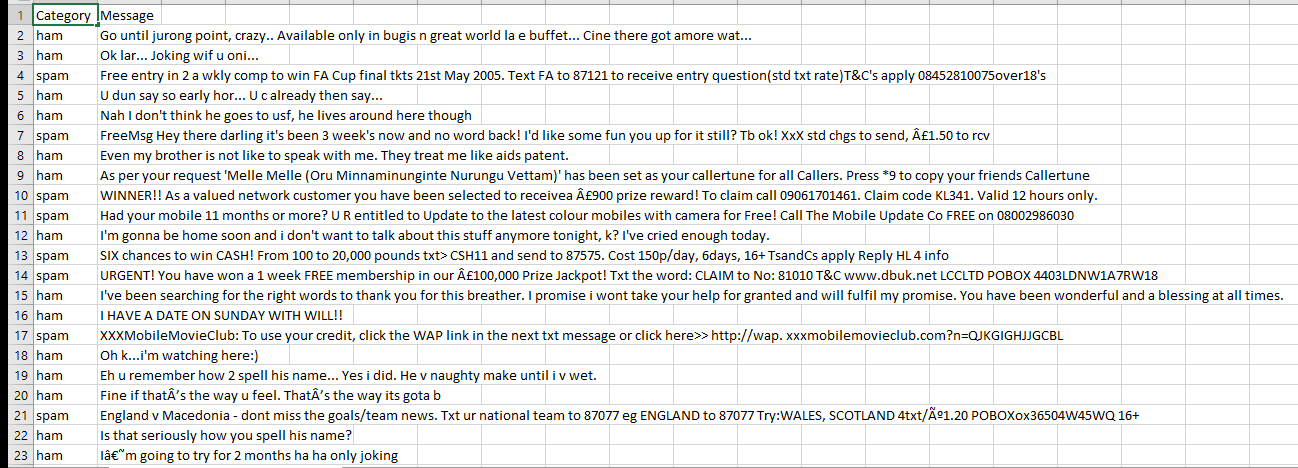
**Capstone Project Report**

1. **Problem Statement**

The aim of my code is to look at various emails/messages and decipher them to figure out if they are spam or ham (not spam). I am going to use the Decision Tree Algorithm.

1. **The Dataset**

The dataset that I am using looks like this:



As we can see, this dataset has two columns. The Category column and the Messages column.

Under the Category column, we can see the different emails have been classified as either spam or ham.

Under the Messages column, we can see exactly what the emails being classified are.

The dataset contains over 5000 emails and messages making sure that the predictions can be as accurate as it needs to be.

1. **Why do we need AI for this?**

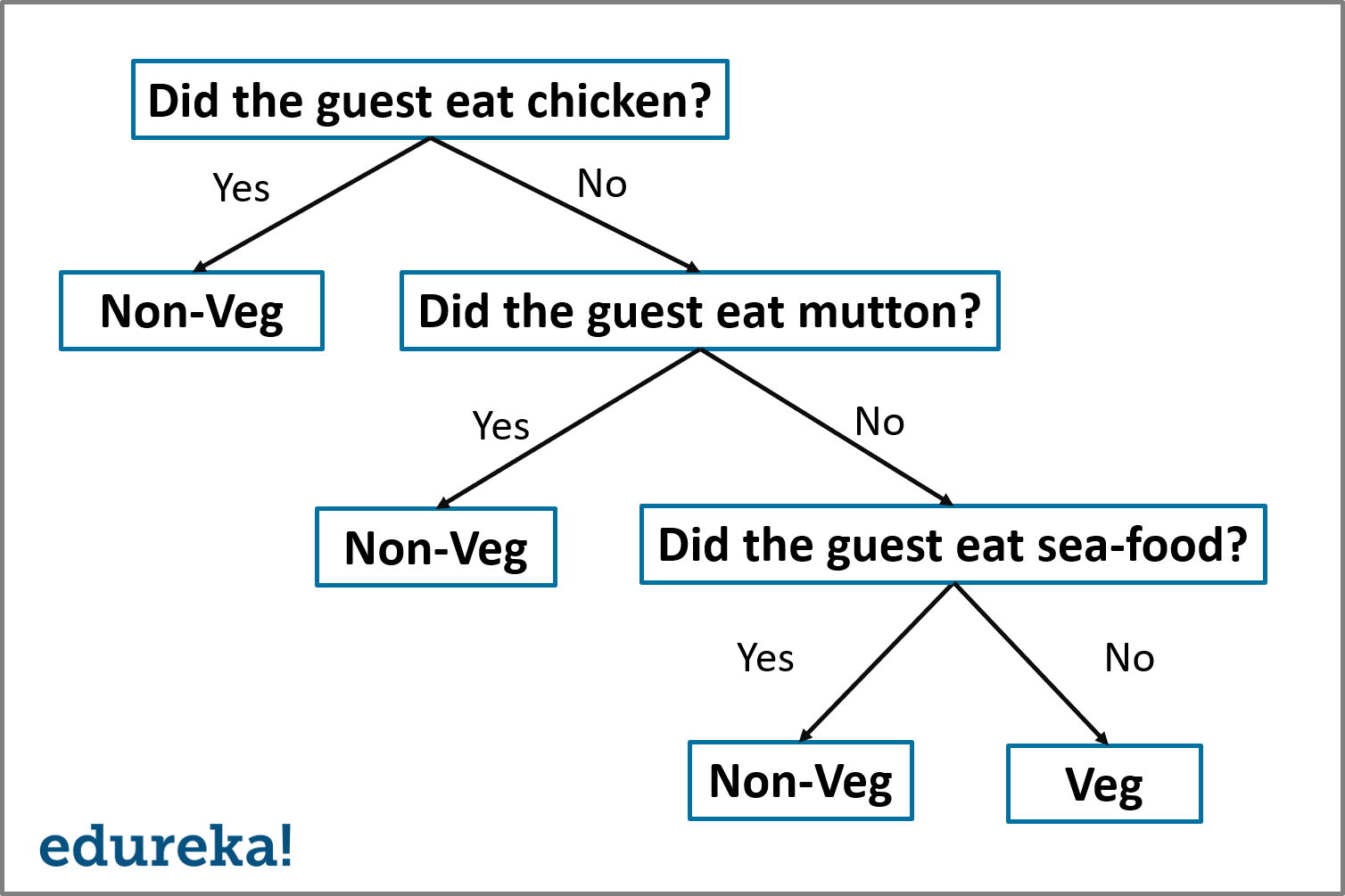
Instead of us having to go through our many messages and emails to delete all the spam, we can just use AI to do it instantly and only show us the important messages.

This saves time and effort that can be used for something more important.

1. **Decision Tree Algorithm**

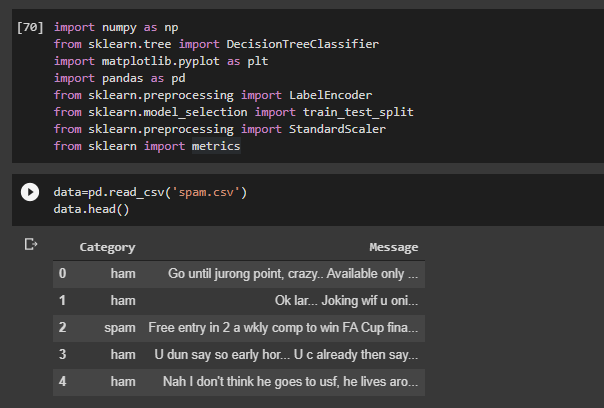
A decision tree is a flowchart-like structure in which each internal node represents a "test" on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label (decision taken after computing all attributes).

For example:



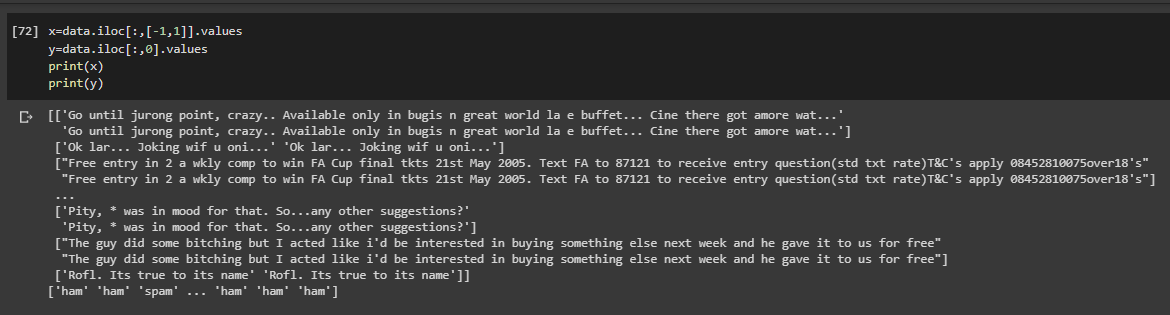
As we can see in this representation, the input goes through a series of tests that determine the attributes of the data. Let’s say that for the first question, the output is no this then leads into the next question “Did the guest eat mutton?” and if the answer comes out as yes, then it gives the final output, but if it is no, it moves on and the flowchart continues.

1. **The Code**

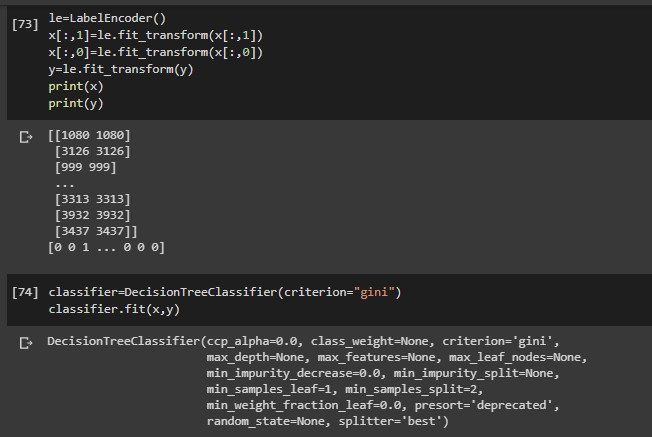


In this part of the code, we import all the libraries that we need to use for this code.

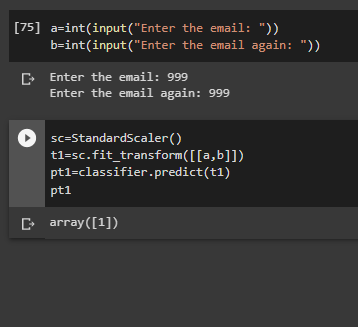
Then we use pandas to read the dataset and then we display the first five rows of it using data.head()



In this part of the code, we assign the values that we need to use from the dataset into the variables x and y, we use both -1 and 1 for x as we need a 2D array to train the model.



Over here we use the label encoder to encode the x and y variables as we cannot use the classifier for strings so we need to convert the values into numbers.



And finally, we accept the email in the form of numbers from the user and use it to predict whether it is spam or ham. If the output is 1 then the message/email is spam if it comes out as 0 then it is ham (not spam).