**JUSTIFICATION FOR CHOICE OF DATA STRUCTURE**

**Data structure used:**

There are numerous data structures such as stacks, queues, linked-list and trees and choosing the right data structure is a crucial task for a programmer. The way raw data is organised in memory must be structured in such a way that the data can easily be retrieved stored and updated.

Keeping in mind that the complexity of the program is proportional to the data structure that is used, the data structure adopted was the Python list. With other programming languages, this built-in data structure is normally referred to as an array. In Python, lists are more versatile because it can grow in size without the need to be explicitly declared its size it is also easily changeable. It is a similar concept to the array list because it is dynamic in size I.e. once the array has reached its limit, its size doubles. In addition to this, lists in python are not limited to storing one specific data type. Using three separate lists to store the data in each csv file for the pets, wild animal and treatment I decided to store every row of the csv files as objects in a list. I.e. every row of each CSV file was stored as an individual object in the template classes I created for each csv.

**Why I believe it is suitable:**

Lists enable me to keep data which is attributed to the same thing together, this allows me to condense my code, as well as perform the same methods on multiple values at once. Indexing my lists in code was easily achievable, practical and efficient because I could easily loop through the items in the documents by calling the attributes of the object I.e. if I needed to access the sanctuary ID of every object row, I search for the attribute ‘sanctuary\_identification’ of every object as opposed to indexing by number. I was also able to write to different parts of the csv such as the Vaccination by searching for an animal using its sanctuary id and writing to the vaccination attribute