Exercise koda bir daha bakılabilir

Iterator: How traversal of data structures happens and who makes it happen.

1. Iteration (traversal) is a core functionality of various data structures
2. An iterator is a class that facilitates the traversal
   1. Keeps a reference to the current element
   2. Knows how to move to a different element
3. Java has Iterator<t> and Iterable<t>
   1. Iterator<t> specifies the iterator API
   2. A class needs to be Iterable in order to support for(Foo foo :bar) loops
4. Iterator: An object that facilitates the traversal of a data structure.

Tree traversal

1. In example code we are traversing in order.
2. For it to be able to be used in a for loop it has to implement an iterable. (example of binary tree)

Array Backed Properties

1. Property = field + getter + setter
2. You have a creature that implements Iterable
3. In this creature you keep the stats of the creature in an array and when you want to do some operations like sum or average you can simply use this array.
4. When you add a new field to the creature you don't have to modify your sum or average functions.

Summary

1. An iterator species how you can traverse an object
2. Iterator cannot be recursive (no coroutines)
3. Iterator implements Iterator<t> , iterable object implements Iterable<T>