Lists and Strings

IT Adventures: Smart IT

Reading Assignment

- How to Think Like a Computer Scientist: Interactive Edition
  - https://runestone.academy/runestone/books/published/thinkcspy/index.html
- Required : 9.1-9.13
- Recommended: Whole Chapter
- Required : 10.1-10.14

Lists

- A list is a variable that can store multiple values that can be of different types.
- They are indexed.
  - They can have the same value stored more than once.
- They are malleable.
  - You can change the contents of each index whenever you want.
- They are ordered.
  - When you add a value to the list it will be placed at the end of the list.
    - There are methods of adding values to other places in the list as well.
- They are created by assigning multiple values, separated by commas to a variable while surrounded by square brackets.
  - X = [“apple”, 5, ‘t’, True]

Basic Tools for Lists

- Change individual elements by treating each index like its own variable.
  - list[2] = 4 will change the value of list[2] to 4.
- Add items to list at a certain index by using .insert(x,value)
  - list.insert(1, 43) will insert the value 43 into the list at index 1 and moving all other indexes up 1.
- Add elements to the end of the array with .append(value).
  - list.append(“apple”) places the value of apple at the end of the array.
- Combine 2 lists with .extend(list2).
  - list.extend(list2) appends list2 to the end of list .
- Remove a unique value from the list with .remove(value).
- Remove a value from the list based on index with .pop(i).
  - list.pop(1) removes the second value from the list and decreases the index of all other elements by 1.
  - list.pop() removes the last element from the list.
- .clear() will clear the list.
  - Printing the list will print []
- Many more tools available and documented online.

What is a String? Refresher

- A string is a “string”, or list, of characters
  - More specifically it is a list of individual characters
- They are a mix between a normal variable and an object, which we’ll talk about later
  - They are declared just like a normal variable, but you surround them with quotation marks
    - foo = “bar”
- However, unlike normal variables, there are many properties that we can easily manipulate to do some interesting things.
- Moreover there are a lot of built in tools that we can use with strings as well

Basic Tools for Strings

- len(string) will return the length of the string. This is useful for loops and many other things.
  - X = “foo”
  - len(X) will return 3.
- String[i] will return the character at the ith place in the string.
  - Remember that strings are arrays and arrays start at 0.
  - X[0] will return f.
- in can be used to test for substrings.
  - “yummy yummy” in “fruit salad yummy yummy” will return true.
    - This is a boolean expression and other boolean expressions can be used as well.
      - “hill” not in “mountain” will return true.
- string.upper() returns the string with all uppercase letters.
  - string.lower() returns all lowercase.
- Negative indexing, is the idea of going to the back of the string and returning each character until, but not including the second argument.
  - X[-3:-1] will return “fo”.
  - [::-1] will return the string but backwards.

Technical Challenge

- Create a list of words of varying lengths and then write a program that will accept a number from the user and then print manipulations of the string.
- It will print:
  - The string
  - The string in all lowercase
  - The string in all uppercase
  - The first and last characters
  - The string backwards