Creating and Modifying Text, part 2

Lecture 5

Taking Strings Apart with Indices

- Programming concept: access *mailboxes* by number (lists/arrays)
- Literal string: "Hello" …five *mailboxes* with five characters

Address	0 [0] [-5]	1 [1] [-4]	10 [2] [-3]	11 [3] [-2]	100 [4] [-1]
8-bit char	0100 1000	0110 0101	0110 1100	0110 1100	0110 1111
ASCII char	72	101	108	108	111
Latin-1	H	e	l	l	o

- "Length" is the number of *mailboxes*
- The address of the last mailbox is always (length – 1)

Tools to Use with Indices: length() and range()

- Indices: square brackets notation
  - `myString[0]`
  - `myString[index]`
- `len(myString)` …the number of items
- `len(myString - 1)` …the index of the last item
- `range()` …a useful counting tool: returns an array (a list) of numbers (integers)
  - `range(`y`)` : start at zero and count up to but not including the y
  - `range(`x`,`y`)` : start at x and count up to but not including the y
  - `range(`x`,`y`,`z`)` : start at x, count up to but not including the y, skip-counting by z

Using an Index in a FOR loop

```
#Program 14: Print the Parts of Any String
def parts(string):
  for letter in string:
    print letter

parts("Hello")
>>>
H
e
l
l
o
```

```
#Program 24: Print the Parts of Any String, with Indices
def parts2(string):
  for index in range(len(string)):
    print string[index]

parts2("Hello")
>>>
H
e
l
l
o
```

Counting tricks using len() and range()

- `for index in range(len(string)):`
  - Repeats the FOR loop once for each element in `string`
- `for index in range(len(string)/2):`
  - Repeats the FOR loop for only half the number of elements in `string`
- `for index in range(len(string)-1, -1, -1):`
  - Repeats the FOR loop once for each element in `string`, starting at the end and counting backwards

Useful methods to use on Strings

- Change the case of a String
  - `.upper()`
  - `.lower()`
  - `.title()`
  - `.swapcase()`
- Create a list of words: `.split()`
  - Returns a *list* of strings, delimited by the space character
- Search a string for the first occurrence of a char(s): `.find()`
  - Returns the index (a number) of the position of the char(s) if it exists (else -1)

More: https://thehelloworldprogram.com/python/python-string-methods/

What a Computer Can Do

There are only six things…

- Store data (e.g. variables)
- Store and follow instructions (e.g. functions or methods)
- Take data apart (e.g. myString[0])
- Transform data (e.g. chr(214) or ord("A")
- Follow a set of instructions repeatedly (e.g. FOR loops)
- Test data (true or not) (e.g. IF statements)