

These are a set of exercises to practice with looping and strings.

(1) Remember how I've told you that EVERY CHARACTER you type is represented in the computer is stored as a number?

Each and every letter on your keyboard (including space) is represented by an ascii code value. You can see these values at <http://www.asciitable.com/>.

Write your first name in comments in your code, and look up the decimal (Dec) ascii code by hand for each letter in your name. For example, "nick" = 110, 105, 99, 107. Add these ASCII values as COMMENTS in your code. Note that upper case letters and lower case letters have different numeric values.

(2) Remember that we can access any part of a string using it's INDEX VALUE. Index values go from 0, to (1 less than the length of the string). For example, my name as a string is "nick". The length of this string is 4 - it's 4 letters long. The index position of the last letter, the "k" is 3, because we start counting at 0.

Remember also that there is a function you can use that will tell you the length of ANY sequence. That's the len() function, where the thing you want to measure goes INSIDE the parentheses. For example:

```
>>>len("nick")
```

```
4
```

```
>>> len("the quick brown fox jumps over the lazy dog")
```

```
43
```

(a) Write code that asks the user to enter a word. Print the first letter of that word using it's index.

(b) Print the last letter of that word, making use of the len function to do so. Make sure your print statements are informative, e.g. "The first letter is...". This code should work with words of ANY length.

(3) You used this information in class to write a program that lets a user enter a word, and then prints each letter of this word, one at a time. For example:

```
Enter your word: nick
```

```
Letter is: n
```

```
Letter is: i
```

```
Letter is: c
```

Letter is: k

Now, write a program, so that it prints any word BACKWARDS - i.e. the last letter first.

(4) There is a function, `ord()`, which gives you the ascii value integer of a single character: e.g.

```
>>>ord("n")
```

```
110
```

Can you use this information to write a program to turn ANY string entered by the user into the ascii codes for each character in that string? Test it with your first name, and make sure you get the same answer as in (1). E.g.

Enter name: Nick

N is: 110

i is: 105

c is: 99

k is: 107

(5) The function `chr()` takes an integer, and gives you the resulting ASCII character. For example, `chr(110)` will give you 'n'. You should have noted that there are different numeric values for upper case and lower case letters.

Can you write a program that asks the user for a word in lower case, and translates it into UPPER case, one letter at a time. Do NOT use any special functions to do this.

HINT. Convert the letter to a number. Subtract 32. Turn the resulting number back into a letter.

Enter word: zebra

Z

E

B

R

A

(6) Write a program that counts the number of vowels in a word (a, e, i, o or u). The user should enter the word, and the program should print the total number of vowels in that word. Start with a count of zero, look at each letter of the word, and then add one to the count if...what?

(7) Write a program that asks the user for a letter, and then asks them for a word. Print FOUND AT INDEX if the letter is contained in the word, and include the index that it is found at. Print NOT FOUND if the letter isn't in the word. Make the code print ONLY ONCE, even if there are multiples of the letter. Example:

Enter word: nikki

Enter letter: k

Found at index 2

Another example:

Enter word: Zebra

Enter letter: z

Not found

(remember, 'z' and 'Z' are not the same)

HINT: How many reasons are there for this loop to keep going? Remember, we use ONE boolean expression for EACH reason.

HINT: This code might end up looking VERY similar to something we did a few classes ago, or that I asked you to do in a previous assignment.