**Presentation Notes**

1. What does the ASCII acronym stand for?
2. What is the ASCII code used for?
3. Encoding characters (i.e. letters on the keyboard) into ASCII code numbers  
   1. What is the ASCII code for the letter “A”
   2. What is the ASCII code for the letter “a”
   3. Why are they different?
   4. What is the ASCII code for the space bar?
4. Decoding ASCII code numbers into characters and letters   
   1. What character corresponds to ASCII code 61 decimal
   2. What character corresponds to ASCII code 8 decimal
   3. Why is the character 8 not the same as ASCII code 8
   4. What is the range of non-printable characters in ASCII

1. How would you code the string “Hello” in ASCII?
2. How would you code the string “127” in ASCII?
3. What is the difference between 127 and “127”?

**Student Questions**

1. Why do computers have to convert characters (i.e. letters on the keyboard) into numbers? Why can’t computers just use the letters directly?

Computers convert letters to numbers because the computer does not understand letters. Rather it understands numbers. The computer uses the binary system because there are fewer computational errors.

1. How do computers communicate with people who speak different languages and use different alphabets? What is used instead of the ASCII code table?

They have different tables that add characters to the ASCII table that the language might need. Other tables that may be used are 7-bit codes, 8-bit codes, and Unicode.

1. Research online-documentation for the Python **ord()** function. Provide some sample code that demonstrates the use of the **ord()** function.

The ord() function is used to classify what letter represents the integer. An example is if ord(J) is typed, I will be displayed with an answer of 74 in decimal

1. Research online-documentation for the Python **chr()** function. Provide some sample code that demonstrates the use of the **chr()** function.

The chr() function does the opposite of ord(). If you enter this code: print((chr(99)), it will display with the answer of c. Another example is print((chr(74)), it will display 72 in decimal.

1. Write a Python program that uses the ord() and chr() functions to do the following:
   1. Read a single character (i.e. single letter or keyboard symbol) from the console input.

print(chr(99))

Output:

c

Example 2

print(chr(98))

Output:

b

* 1. Convert the character to an ASCII code number

Example 1

print(ord(('a'))

Output:

97

* 1. Add 3 to the code number.

Another Example::

number1 = ord(input("Enter a character: "))

number2 = number1 + 3

* 1. Convert the new code number back to a character (i.e. single letter or keyboard symbol)

number3 = chr(number2)

* 1. Print the new character to the console output.

 print(number3)

-----------------------

num1 = ord(input("Type a character: "))

num2 = num1 + 3

num3 = chr(num2)

print(num3)

--------------------

1. Enhance your program to add the following features:
   1. After reading the single character from console input, check to make sure that the character is a letter (i.e. a to z or A to Z). Print a warning message if the character is not a letter.
   2. After converting the code number back to a character, print a “\*” if the character is not a letter.

**Extension (Optional)**

1. Extend your program to operate on a string read in from the console input.
   1. Use a loop to process the string as a sequence of single characters
   2. Use your original code process the characters
   3. Append the characters to make a new output string
   4. Print the new string to console output