**Presentation Notes**

1. What does the ASCII acronym stand for?

This stands for American Standard Code for information interchange

1. What is the ASCII code used for?

It is used for representing and storing text in computers

* Computers can only understand numbers(binary)
* Text symbols must be encoded as numbers

1. Encoding characters (i.e. letters on the keyboard) into ASCII code numbers  
   1. What is the ASCII code for the letter “A”  
      65
   2. What is the ASCII code for the letter “a”  
      97
   3. Why are they different?  
      Upper case and lower case are different symbols. The computer doesn't really know what the alphabet is or how to read and write.
   4. What is the ASCII code for the space bar?

32. The ASCII code also includes some "un-printable" characters.

1. Decoding ASCII code numbers into characters and letters   
   1. What character corresponds to ASCII code 61 decimal  
      The = symbol
   2. What character corresponds to ASCII code 8 decimal  
      backspace
   3. Why is the character 8 not the same as ASCII code 8  
      it is because it is a number and not a symbol
   4. What is the range of non-printable characters in ASCII

The range is from 0-32

1. How would you code the string “Hello” in ASCII?  
     
   You would type 72,101,108,108,111
2. How would you code the string “127” in ASCII?  
     
   You would type 49, 50, 55
3. What is the difference between 127 and “127”?

The difference is that 127 is the integer number and “127” is a text symbol

**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?

It is because computers only use numbers (binary)

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

It is because ASCII is only for the English and for other languages it is the extended ASCII using ALT. Unicode is also something that is used for different languages. It is a 16-bit character set where all the characters occupy the same space. In Unicode each meaning is given its own meaning. There is also the EBCDIC code

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

Given a string of length one, return an integer representing the Unicode code point of the character when the argument is a unicode object, or the value of the byte when the argument is an 8-bit string. For example, **ord**('a') returns the integer 97, **ord**('€') (Euro sign) returns 8364.

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

The **chr()** method returns a string representing a character whose Unicode code point **is** an integer. ... The **chr()** method returns a character whose unicode point **is** num, an integer. If an integer **is** passed that **is** outside the range then the method returns a ValueError.

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.
   2. Convert the character to an ASCII code number.

num = ord(input(""))

print(num)

* 1. Add 3 to the code number.

num = ord(input(""))

num = num + 3

print(num)

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

num = ord(input(""))

num = num + 3

print(num)

print(chr(num))

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

num = ord(input(""))

num = num + 3

print(num)

print(chr(num))

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.

num = ord(input(""))

print(num)

print(chr(num))

if (num < 57):

if (num > 47):

print ("Warning")

* 1. After converting the code number back to a character, print a “\*” if the character is not a letter.

num = ord(input(""))

print(num)

if (num < 57):

if (num > 47):

print ("\*")

**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