**Presentation Notes**

1. What does the ASCII acronym stand for?

**American Standard Code for Information Interchange**

1. What is the ASCII code used for?

Representing and storing text in computers

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

Encoding text for electronic communication (e.g. web)

* Sending and receiving computers must both agree and understand the same encoding standard

1. Encoding characters (i.e. letters on the keyboard) into ASCII code numbers  
   1. What is the ASCII code for the letter “A”

65

* 1. What is the ASCII code for the letter “a”

97

* 1. Why are they different?

One is upper case and one is lower case, they’re different so the computer knows the difference.

* 1. What is the ASCII code for the space bar?

32

1. Decoding ASCII code numbers into characters and letters   
   1. What character corresponds to ASCII code 61 decimal

=

* 1. What character corresponds to ASCII code 8 decimal

Backspace

* 1. Why is the character 8 not the same as ASCII code 8

Character "8" is text symbol and code 8 is a number. Symbols and numbers are different things to a computer.

* 1. What is the range of non-printable characters in ASCII

Codes 0 to 31

1. How would you code the string “Hello” in ASCII?  
     
   72 101 108 108 111

1. How would you code the string “127” in ASCII?  
     
   49 50 55

1. What is the difference between 127 and “127”?

127 is an integer number and "127" is a string of text symbols.

**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 have to convert characters because they have built in binary systems and they can’t understand letters, only numbers.

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

The standard for international language codes is ISO 639.

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

# code point of integer

print(ord('5'))

# code point of alphabet

print(ord('A'))

# code point of character

print(ord('$'))

53

65

36

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

print(chr(97))

print(chr(65))

print(chr(1200))

a

A

Ұ

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.
   3. Add 3 to the code number.
   4. Convert the new code number back to a character (i.e. single letter or keyboard symbol)
   5. Print the new character to the console output.
2. yourCharacter= input('enter a character:')
3. theCode=(ord(yourCharacter))
4. print (theCode)
5. print (theCode + 3)
6. print('your new character is:')
7. print (chr(theCode+3))
8. 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.

yourCharacter= input('enter a character:')

theCode=(ord(yourCharacter))

if (127 > theCode or theCode < 65):

print ('warning: this character is not a letter')

print (theCode)

print (theCode + 3)

print('your new character is:')

print (chr(theCode+3))

newCharacter= (theCode+3)

if (127 > newCharacter) or (newCharacter<65):

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