**Presentation Notes**

1. What does the ASCII acronym stand for?

American Standard Code for Information Interchange

1. What is the ASCII code used for?

Computers use ASCII code to convert text into numbers in order for the computer to understand it. This is the standard for sending messages to other computers.

This represents and stores text within the computers (due to computers only understanding numbers (binary), so text must be encoded as numbers.

The computers used to send ASCII code must both use 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”

DEC: 65

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

DEC: 97

* 1. Why are they different?

They are different symbols, meaning the computer has to read them as 2 different numbers

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

DEC: 32

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

chr: =

* 1. What character corresponds to ASCII code 8 decimal

chr: BS/Backspace

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

The definition of “8” is not defined as chr = 8 in ASCII

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

Anything above 127 and anything below 1.

1. How would you code the string “Hello” in ASCII?  
   72 101 108 108 111  
   H e l l o
2. How would you code the string “127” in ASCII?  
   49 50 55

1 2 7

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

127 in ASCII is DEL, which deletes text, compared to the actual text of “127” which is 49 50 55 in ASCII

**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 can only read numbers, meaning that they don’t understand letters, and thus, they must convert the text into numbers.
2. How do computers communicate with people who speak different languages and use different alphabets? What is used instead of the ASCII code table?

An extension/remaster of ASCII used is called ISO 8859/Unicode/UTF developed by ISO

* Latin-1 (Western European languages)
* Latin-2 (Non-Cyrillic Central and Eastern European languages)
* Latin-3 (Southern European languages and Esperanto)
* Latin-5 (Turkish)
* Latin-6 (Northern European and Baltic languages)
* 8859-5 (Cyrillic)
* 8859-6 (Arabic)
* 8859-7 (Greek)
* 8859-8 (Hebrew)

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

a = chr(65)

b = chr(100)

print (a, b)

c = ord("a")

print (c)

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

a = chr(65)

b = chr(100)

print (a, b)

c = ord("a")

print (c)

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.

a = str(input("Please type a letter, symbol or number and press enter: "))

trueA = ord(a)

print ("Original number code:", trueA)

trueA = trueA + 3

print ("New number code:", trueA)

trueA = chr(trueA)

print ("Final text:", trueA)

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.

while (1 == 1):

a = input("Please type a letter and press enter: ")

print ("")

if (a.isalpha()) == True:

break

if (a.isalpha()) == False:

print ("Error: Text is not a letter.")

print ("")

break

trueA = ord(a)

print ("Original number code:", trueA)

print ("")

trueA = trueA + 3

print ("New number code:", trueA)

print ("")

trueA = chr(trueA)

if (a.isalpha()) == False:

print ("Final number:\*")

if (a.isalpha()) == True:

print ("Final text:", trueA)

**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