**Presentation Notes**

1. What does the ASCII acronym stand for?

**American Standard Code for Information Interchange**

1. What is the ASCII code used for?

**An 8-bit system used for identifying characters and numbers on a keyboard**

1. Encoding characters (i.e. letters on the keyboard) into ASCII code numbers  
   1. What is the ASCII code for the letter “A” **065**
   2. What is the ASCII code for the letter “a” **097**
   3. Why are they different? **Because they are different cases**
   4. What is the ASCII code for the space bar? **32**
2. 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: **Backspace**
   3. Why is the character 8 not the same as ASCII code 8: **Because decimal 8 stands for Backspace**
   4. What is the range of non-printable characters in ASCII: **009 to 037**

1. How would you code the string “Hello” in ASCII?  
     
   **072 101 108 108 111**
2. How would you code the string “127” in ASCII?  
     
   **049 050 055**
3. What is the difference between 127 and “127”?

**One is a number and the other is a piece of ASCII code**

**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?

**Because a computer can’t read 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?

**Unicode uses UTF to translate symbols and emojis into numbers for people who don’t speak english**

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

**print(ord('5'))**

**print(ord('A'))**

**print(ord('$'))**

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

**numb = 0x03a9**

**print(chr(numb))**

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.

letter = str(input("Enter a letter: "))

symbol = ord(letter)

print(ord(letter))

newSymbol = symbol + 3

print(chr(newSymbol))

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.

letter = str(input("Enter a letter (\*WARNING, IT CAN NOT BE A NUMBER!\*): "))

symbol = ord(letter)

if (symbol >= 97) and (symbol <= 122) :

print (symbol)

elif (symbol >= 65) and (symbol <= 90) :

print (symbol)

elif (symbol < 97) or (symbol > 122) :

print ("\*")

elif (symbol < 65) or (symbol > 90) :

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