**Level 1: Basic ASCII Coding**

1. Research the "ASCII Code"
   1. Explain what ASCII stands for.

American Standard Code for Information Interchange.

* 1. Explain how to convert a letter into an ASCII coded number

You will need to understand and interpret where that character (char) falls under in category of either a decimal, octal or hexadecimal.

* 1. Explain how to decode an ASCII number into a letter

You will need to understand and interpret where that decimal, octal or hexadecimal falls under in category with the various list of characters (char) which are based from the alphabet in lower and upper case forms.

1. Open a new Python Repl and run the sample program provided at the end of this module.
   1. Briefly summarize what the "asciiCodes" list does

This variable has sets of arrays that contain two types which are a character and decimal (Char, Dec). The character represents what it is converted to as a decimal in ASCII. This variable is and can be stored with many characters representing decimals.

* 1. Briefly summarize what the "textCoder" function does

The textCoder is a defined function in the program that is used to represent the letter of the decimal which is in proportion with the variable the called “textChar.” It simply converts the inputted character (char) into a decimal (dec).

* 1. Briefly summarize what the "textDeCoder" function does

The textDeCoder is a defined function in the program that is used to represent the decimal of the letter which is in proportion with the variable called “codedChar.” It simply decodes the inputted code (decimal) into a character (char).

* 1. Briefly summarize what the main program code does

Firstly, the user will be prompted to enter an input which is a password (character/char). That character will then go through the process of being matched with its decimal counterpart. Where it will be analyzed based on thecreated the “asciiCodes” variable that is and can be stored with many characters representing decimals. The program will then print and output the decimal. The user will then be asked to enter a password to decode or return to use the coded string. Where now it is the opposite and the user inputs a decimal number to get a printed output of a character.

1. Explain the main limitation of the program.

This program only supports characters “ABCD” and “abcd” to be inputted. Which limits all the rest of the lower and uppercase letters of the alphabet and even the special characters

(ex. &, $, #, %,&, etc..) including their decimal counterparts which are a part of the ASCII.

**Level 2: Extending The Program**

1. Modify the sample program to do the following (Still using the ASCII code):
   1. Code all of the uppercase and lowercase letters
   2. Code the digits 0 to 9
   3. Code at least 5 special characters (e.g. "1?$%&")
2. Verify that your program works for ***coding*** a message containing all of the basic and special characters.
   1. Provide a sample of your program output below.





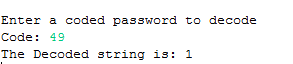
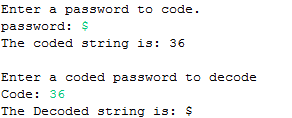




1. Verify that your program works for ***de-coding*** a message containing all of the basic and special characters.
   1. Provide a sample of your program output below.







1. List your program modifications below:

*The program is also posted in the B.7 folder under a sub folder called “Java Programs Created and Used”*

import java.util.Scanner;

public class ASCIIApendixCharDec {

public static void main(String[] args) {

Scanner sc = new Scanner (System.in);

System.out.println("Enter a password to code.");

System.out.print("password: ");

String character = sc.nextLine();

//Convert the string to a character

char password = character.charAt(0);

int codeOut = 0; //Create a variable to be returned in the method

int c = ChartoDec(password, codeOut); //Stores the method and its contents in a variable

System.out.println("The coded string is: " + c ); //Prints the method's returned decimal

//Repeats the same process above but is the opposite as we are decoding now

System.out.println("Enter a coded password to decode");

System.out.print("Code: ");

int codeIn = sc.nextInt();

char textOut = 0;

char t = DectoChar(codeIn, textOut);

System.out.println("The Decoded string is: " + t ); //Prints the method's returned character

}

public static int ChartoDec(char password, int codeOut) {

if (password == ' ')

{

codeOut = ' ';

}

else if (password == 'A')

{

codeOut = 65;

}

else if (password == 'B')

{

codeOut = 66;

}

else if (password == 'C')

{

codeOut = 67;

}

else if (password == 'D')

{

codeOut = 68;

}

else if (password == 'a')

{

codeOut = 97;

}

else if (password == 'b')

{

codeOut = 98;

}

else if (password == 'c')

{

codeOut = 99;

}

else if (password == 'd')

{

codeOut = 100;

}

else if (password == '1')

{

codeOut = 49;

}

else if (password == '?')

{

codeOut = 63;

}

else if (password == '$')

{

codeOut = 36;

}

else if (password == '%')

{

codeOut = 37;

}

else if (password == '&')

{

codeOut = 38;

}

return codeOut;

}

public static char DectoChar(int codeIn, char textOut) {

if (codeIn == 32)

{

textOut = ' ';

}

else if (codeIn == 65)

{

textOut = 'A';

}

else if (codeIn == 66)

{

textOut = 'B';

}

else if (codeIn == 67)

{

textOut = 'C';

}

else if (codeIn == 68)

{

textOut = 'D';

}

else if (codeIn == 97)

{

textOut = 'a';

}

else if (codeIn == 98)

{

textOut = 'b';

}

else if (codeIn == 99)

{

textOut = 'c';

}

else if (codeIn == 100)

{

textOut = 'd';

}

else if (codeIn == 49)

{

textOut = '1';

}

else if (codeIn == 63)

{

textOut = '?';

}

else if (codeIn == 36)

{

textOut = '$';

}

else if (codeIn == 37)

{

textOut = '%';

}

else if (codeIn == 38)

{

textOut = '&';

}

return textOut;

}

}

**Level 3: Creating A Secret Code**

1. Modify the sample program to create your own secret code that is different from the ASCII code:
   1. Work with a partner to create a secret code that codes letters and characters into different letters and characters.
   2. Your program should be able to create a coded message that   
      you can give to your partner
   3. Your program should be able to de-code a coded message that   
      you get from your partner

The program is based on the concept below:



1. Provide a sample of your program output below.
   1. Show how your program codes a secret message









* 1. Show how your program de-codes a secret message



1. List your program modifications below:

*The program is also posted in the B.7 folder under a sub folder called “Java Programs Created and Used”*

// A simple program that represents all the 26 uppercase letters of the alphabet.

// If a user enters 'A' that would be outputted back as 1 cause A is the first letter of the alphabet.

import java.util.Scanner;

public class AlphabetsLettersToNumbers {

public static void main(String[] args) {

Scanner sc = new Scanner (System.in);

System.out.println("Enter a password to code.");

System.out.print("password: ");

String character = sc.nextLine();

//Converts the string to a character

char password = character.charAt(0);

int codeOut = 0; //Create a variable to be returned in the method

int c = LettertoNum(password, codeOut);

System.out.println("The coded string is: " + c ); //Prints the method's returned decimal

//Repeats the same process above but is the opposite as we are decoding now

System.out.println("Enter a coded password to decode");

System.out.print("Code: ");

int codeIn = sc.nextInt();

char textOut = 0;

char t = NumtoLetter(codeIn, textOut);

System.out.println("The Decoded string is: " + t ); //Prints the methods returned character

}

public static int LettertoNum(char password, int codeOut) {

if (password == 'A')

{

codeOut = 1;

}

else if (password == 'B')

{

codeOut = 2;

}

else if (password == 'C')

{

codeOut = 3;

}

else if (password == 'D')

{

codeOut = 4;

}

else if (password == 'E')

{

codeOut = 5;

}

else if (password == 'F')

{

codeOut = 6;

}

else if (password == 'G')

{

codeOut = 7;

}

else if (password == 'H')

{

codeOut = 8;

}

else if (password == 'I')

{

codeOut = 9;

}

else if (password == 'J')

{

codeOut = 10;

}

else if (password == 'K')

{

codeOut = 11;

}

else if (password == 'L')

{

codeOut = 12;

}

else if (password == 'M')

{

codeOut = 13;

}

else if (password == 'N')

{

codeOut = 14;

}

else if (password == 'O')

{

codeOut = 15;

}

else if (password == 'P')

{

codeOut = 16;

}

else if (password == 'Q')

{

codeOut = 17;

}

else if (password == 'R')

{

codeOut = 18;

}

else if (password == 'S')

{

codeOut = 19;

}

else if (password == 'T')

{

codeOut = 20;

}

else if (password == 'U')

{

codeOut = 21;

}

else if (password == 'V')

{

codeOut = 22;

}

else if (password == 'W')

{

codeOut = 23;

}

else if (password == 'X')

{

codeOut = 24;

}

else if (password == 'Y')

{

codeOut = 25;

}

else if (password == 'Z')

{

codeOut = 26;

}

return codeOut;

}

public static char NumtoLetter(int codeIn, char textOut) {

if (codeIn == 1)

{

textOut = 'A';

}

else if (codeIn == 2)

{

textOut = 'B';

}

else if (codeIn == 3)

{

textOut = 'C';

}

else if (codeIn == 4)

{

textOut = 'D';

}

else if (codeIn == 5)

{

textOut = 'E';

}

else if (codeIn == 6)

{

textOut = 'F';

}

else if (codeIn == 7)

{

textOut = 'G';

}

else if (codeIn == 8)

{

textOut = 'H';

}

else if (codeIn == 9)

{

textOut = 'I';

}

else if (codeIn == 10)

{

textOut = 'J';

}

else if (codeIn == 11)

{

textOut = 'K';

}

else if (codeIn == 12)

{

textOut = 'L';

}

else if (codeIn == 13)

{

textOut = 'M';

}

else if (codeIn == 14)

{

textOut = 'N';

}

else if (codeIn == 15)

{

textOut = 'O';

}

else if (codeIn == 16)

{

textOut = 'P';

}

else if (codeIn == 17)

{

textOut = 'Q';

}

else if (codeIn == 18)

{

textOut = 'R';

}

else if (codeIn == 19)

{

textOut = 'S';

}

else if (codeIn == 20)

{

textOut = 'T';

}

else if (codeIn == 21)

{

textOut = 'U';

}

else if (codeIn == 22)

{

textOut = 'V';

}

else if (codeIn == 23)

{

textOut = 'W';

}

else if (codeIn == 24)

{

textOut = 'X';

}

else if (codeIn == 25)

{

textOut = 'Y';

}

else if (codeIn == 26)

{

textOut = 'Z';

}

return textOut;

}

}