Assignment of data values

Assign 20 to the integer variable i
Assign 150.45 to the floating point variable j
Assign "AkiSoft" to the string variable str1
Assign 300 to integer variables var1 and var2 in one line
Assign 1 to integer variable var3, 2 to integer variable var4 and "AkiSoft" to string variable var5 in one line
Print i, j, str1, var1, var2, var3, var4 and var5

String and List manipulation

Assign 'This is my second Python program!' to string variable str
Assign 'And I love it!' to string variable stradd
Print str
Print the first character of str
Print the 3rd to 6th character of str
Print str starting from the 3rd character
Print str two times
Parse str and place the substrings in an array (list)
Print str concatenated with the string "TEST"
Print str concatenated with stradd using the join function

Tuples

Assign the values 'MON', 'TUE', 'WED', 3 to a tuple variable called tuple1
Assign the values 'THR', 'FRI', 2 to a tuple variable called tuple 2
Assign 1, 4, 8, 12 to a list variable called list1
Print tuple1
Print the 1st element of tuple1
Print the elements of tuple1 starting from 2nd until 3rd
Print the elements of tuple1 starting from the 3rd element
Print the concatenated values of tuple1 and tuple2
Append 16 to list1 and print the list

Dictionaries

Create an empty dictionary called dictAge
Create a key called 'Wayne' and assign a value of 20 to dictAge
Create a key called 'David' and assign a value of 40 to dictAge
Create a dictionary called dictJason and assign the following:
    The key 'name' holds the value 'Wayne'
    The key 'employeecode' holds the value 6734
    The key 'dept' holds the value 'DCIT'
Print the value for 'Wayne' key
Print the value for 'David' key
Print the complete dictJason dictionary
Print all the keys of the dictJason dictionary
Print all the values of the dictJason dictionary

Selection structure

Create a list with the values: 'TCP', 'UDP', 'SMTP'
Write code to do the following:
    If the value in the list is 'TCP' the string, "TCP Transport will be used" is printed.
    If the value in the list is 'UDP' the string, "UDP Transport will be used" is printed.
    If the value in the list is 'SMTP' the string, "Unknown Transport" is printed

Repetition structure

Create a list with the values: 'TCP', 'UDP', 'SMTP'
Print the list using a for loop
Add 'FTP' to the end of the list
Print the values in the list using a while loop

Input data

Write code to prompt the user for his/her name and stores the value into a variable called name
Write code to prompt the user for an employee code and stores this in a variable called employeecode
Write code to prompt the user for the department in which he/she works in and stores this in a variable called dept
Print name
Print employeecode
Print dept
Open the file 'employee.txt' for reading

Read each line of the file into a variable called line and print the contents of line. Do this until the end-of-file is reached
Parse the string line and place the substrings into a list variable called list1
Print the 1st item in list1
Print the 2nd item in list1
Print the 3rd item in list1

Function with Default Arguments

Write a Python function which accepts name, employeecode, dept='DCIT' and returns the values as a concatenated string
Call the function with the arguments: 'Wayne Goodridge", '3000' and print the returned string

Function with Required Arguments

Write a Python function which accepts name, employeecode and returns the values as a concatenated string
Call the function with the arguments: 'Wayne Goodridge", '3000' and print the returned string

Function with Key Arguments

Write a Python function which accepts name, employeecode and prints "Name: ", name_value and "Code: ", employeecode_value. The function returns no value.
Call the function with the arguments: name = 'Wayne Goodridge", employeecode = '3000'
Call the function with the arguments: employeecode = '3000', name = 'Wayne Goodridge"

Function with variable number of arguments

Create a function called varArgumentsFunc which accepts a variable number of arguments
Use a for loop to print out all the values of the arguments
Call the function with a value of 33
Call the function with values 50, 60, 70, 90, 77

Next Week --- Python Modules