Muhammad Rizwan Khalid

BSCD-6A

180459

Lab:02: Introduction to Python

Computer Networks

Submission: 21st September, 2018

Lab Engineer: Sir Kaleem

In [1]:
```
#user input program
name = input("What is your name? ")
print("Its our pleasure to greet you Mr. "+name+"!")
age =  input("what is your age? ")
print("You are quite mature having "+age+" years of age")
```

```
What is your name? Rizwan Khalid
Its our pleasure to greet you Mr. Rizwan Khalid!
what is your age? 21
You are quite mature having 21 years of age
```

In [2]:
```
#python Strings
str = 'Muhammad Rizwan Khalid'
print('str = ', str)

#first character
print('str[0] = ', str[0])

#last character
print('str[-1] = ', str[-1])

#slicing 2nd to 5th character
print('str[1:5] = ', str[1:5])

#slicing 6th to 2nd last character

print('str[5:-2] = ', str[5:-2])
```

```
str =  Muhammad Rizwan Khalid
str[0] =  M
str[-1] =  d
str[1:5] =  uham
str[5:-2] =  mad Rizwan Khal
```

In [3]:
```
#following script cover tuple concepts
# empty tuple
my_tuple = ()
print(my_tuple)

# tuple having integers
my_tuple = (1, 2, 3)
print(my_tuple)

# tuple with mixed datatypes
my_tuple = (1, "Hello", 3.14)
print(my_tuple)

# nested tuple
my_tuple = ("Animals", [1, 2, 3], (4, 5, 6))
print(my_tuple)

# tuple can be created without parentheses
# also called tuple packing
my_tuple = 1, 2.3, "Animals"
print(my_tuple)

# tuple unpacking is also possible
a, b, c = my_tuple
print(a)
print(b)
print(c)
```

```
()
(1, 2, 3)
(1, 'Hello', 3.14)
('Animals', [1, 2, 3], (4, 5, 6))
(1, 2.3, 'Animals')
1
2.3
```

In [4]:
```
#list concepts
list = [ 'abcd', 786 , 2.23, 'john', 70.2 ]
tinylist = [123, 'john']
print(list) # Prints complete list
print(list[0]) # Prints first element of the list
print(list[1:3]) # Prints elements starting from 2nd till 3rd
print(list[2:]) # Prints elements starting from 3rd element
print(tinylist * 2) # Prints list two times
print(list + tinylist) # Prints concatenated lists
```

```
['abcd', 786, 2.23, 'john', 70.2]
abcd
[786, 2.23]
[2.23, 'john', 70.2]
[123, 'john', 123, 'john']
['abcd', 786, 2.23, 'john', 70.2, 123, 'john']
```

In [5]:
```
#this script covers nested for loop, functions, list and conditions
def bubbleSort(arr):
    n = len(arr)
    for i in range(n):
        for j in range(0, n-i-1):
            if arr[j] > arr[j+1] :
                arr[j], arr[j+1] = arr[j+1], arr[j]
    return


arr = [64, 34, 25, 12, 22, 11, 90]

bubbleSort(arr)

print ("Sorted array is:")
for i in range(len(arr)):
    print ("%d" %arr[i]),
```

```
Sorted array is:
11
12
22
25
34
64
90
```

In [6]:
```
#this script uses while loop
print ("calculate an average of first n natural numbers")
n = input("Enter Number ")
n = int (n)
average = 0
sum = 0
num = 0
while(num < n+1):
    sum = sum+num;
    num += 1
average = sum / n
print("Average of first ", n, "number is: ", average)
```

```
calculate an average of first n natural numbers
Enter Number 10
Average of first  10 number is:  5.5
```

In [7]:
```
#this script uses dictionary to compute the net amount of stock
#dictionary, for loop variables
prices={}
prices["banana"] = 5
prices["apple"]= 30
prices["orange"]= 10
prices["pear"]= 20
stock={}
stock["banana"]= 6
stock["apple"]= 0
stock["orange"] = 32
stock["pear"]= 15
for food in prices:
    print(food)
    print("price: %s" % prices[food])
    print("stock: %s" % stock[food])
total=0
for price in prices:
    money= prices[price]*stock[price]
    total=total +money
print("The total worth of stock is %.1f" % total)
```

```
banana
price: 5
stock: 6
apple
price: 30
stock: 0
orange
price: 10
stock: 32
pear
price: 20
stock: 15
The total worth of stock is 650.0
```

In [8]:
```
#this script creates and write into a file
with open("test.txt",'w',encoding = 'utf-8') as f:
    f.write("my first file\n")
    f.write("This file\n")
    f.write("contains three lines")
    f.close()
```

In [9]:
```
#this script reads the content of file
with open("test.txt",'r',encoding = 'utf-8') as f:
    print(f.read())
    f.close()
```

```
my first file
This file
contains three lines
```