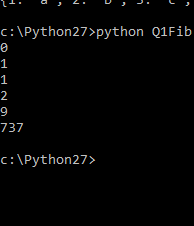
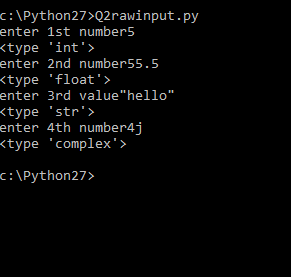
**Answers for Assignment**

Q1.



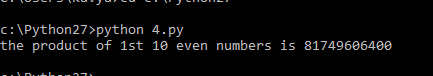
Q2.



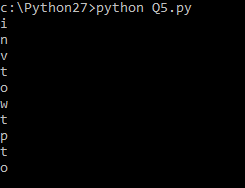
Q3



Q4.



Q5.



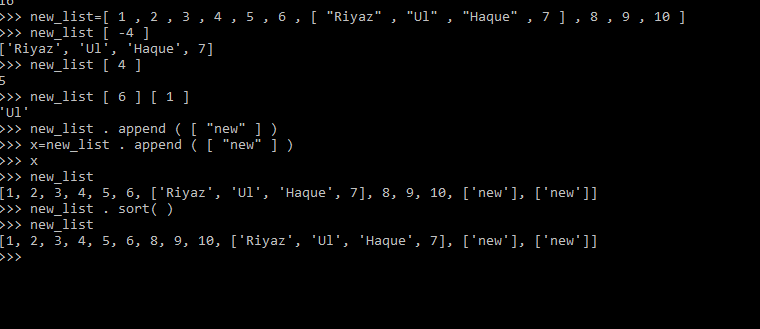
Q6



Q7



Q8



Q9



Q10

* **Negative indexing in python**

Used to index the last element i.e staring from last to first. Supports the list, tuple or any other container class which supports indexing

X= [1,3,4,10,100,35]

x[-2] =100

* **Packing and UnPacking**

We use \* (for tuples) and \*\* (for dictionaries).

**Packing**

When we don’t know how many arguments need to be passed to a python function, we can use Packing to pack all arguments in a tuple.

**def** friends(\*names):

**for** name **in** names:

**print**(name)

friends('Tom')

friends('Tom', 'Jerry')

Tom , Tom , jerry

**Unpacking**  
We can use **\*** to unpack the list so that all elements of it can be passed as different parameters.

**def** fun(a,b,c):

**print**('{} {} {}'.format(a, b, c))

myList = ['Tom', '&', 'Jerry']

fun(\*myList)

Tom & Jerry

**Mutable and Immutable**

Everything in Python is an object. A mutable object can be changed after it is created, and an immutable object can't.

Ex: List is mutable and tuple is immutable

**Append and Extend**

[Append](https://www.geeksforgeeks.org/list-methods-python/)**:** Adds its argument as a single element to the end of a list. The length of the list increases by one.

Ex: List =[1,2,3,33,4,6]

List2 =[10,11,13]

X=list.append(list2)

X=[1,2,3,33,4,6, [10,11,13] ]

Extend : It iterates the addition on each element. It individually adds i.e length increses by number of new elements added

X=List.extend(list2)

X=[1,2,3,33,4,6,10,11,13]

**Pickling and Unpickling**

Pickling: Converting data object into byte stream

Import pickle

Pickle\_out

Unpickling: Byte stream to data object

Import pickle

Pickle\_in

Q11

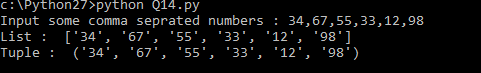


Q12



Q13What is split function in python?

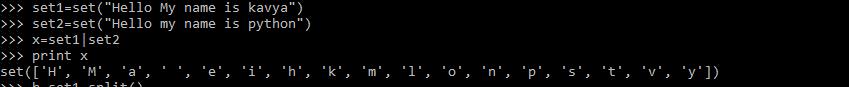
Q14



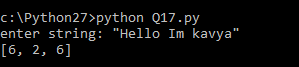
Q15



Q16



Q17



Q18



Q19 What is memory Management in Python?

Memory management in Python involves a private heap containing all Python objects and data structures. Python memory manager manages the private heap. It deals with dynamic storage which involves sharing, segmentation, referencing ,preallocation or caching.

Q20. What is the difference between range and xrange function

range () –

* Give the list of numbers in a specified range
* Can be sliced
* Storage space is more

xrange() –

* It’s a function the loops itself and returns exact same thing value and function
* Can’t be sliced
* Storage space is relatively less

Python 3:

range () – Give the list f numbers in a specified range

* There is no xrange() function in Python 3
* range() function acts as xrange() in Python 2