Python - Queue

We are familiar with queue in our day to day life as we wait for a service. The queue data structure aslo means the same where the data elements are arranged in a queue. The uniqueness of queue lies in the way items are added and removed. The items are allowed at on end but removed form the other end. So it is a First-in-First out method. An queue can be implemented using python list where we can use the insert() and pop() methods to add and remove elements. Their is no insertion as data elements are always added at the end of the queue.

Adding Elements to a Queue

In the below example we create a queue class where we implement the First-in-First-Out method. We use the in-built insert method for adding data elements.

```
class Queue:

  def __init__(self):
      self.queue = list()

  def addtoq(self,dataval):
# Insert method to add element
      if dataval not in self.queue:
          self.queue.insert(0,dataval)
          return True
      return False

  def size(self):
      return len(self.queue)

TheQueue = Queue()
TheQueue.addtoq("Mon")
TheQueue.addtoq("Tue")
TheQueue.addtoq("Wed")
print(TheQueue.size())
```

When the above code is executed, it produces the following result −

```
3
```

Removing Element from a Queue

In the below example we create a queue class where we insert the data and then remove the data using the in-built pop method.

```
class Queue:

  def __init__(self):
```

```
      self.queue = list()

  def addtoq(self,dataval):
# Insert method to add element
      if dataval not in self.queue:
          self.queue.insert(0,dataval)
          return True
      return False
# Pop method to remove element
  def removefromq(self):
      if len(self.queue)>0:
          return self.queue.pop()
      return ("No elements in Queue!")

TheQueue = Queue()
TheQueue.addtoq("Mon")
TheQueue.addtoq("Tue")
TheQueue.addtoq("Wed")
print(TheQueue.removefromq())
print(TheQueue.removefromq())
```

When the above code is executed, it produces the following result −

```
Mon
Tue
```