Information
Technology
Institute

2022

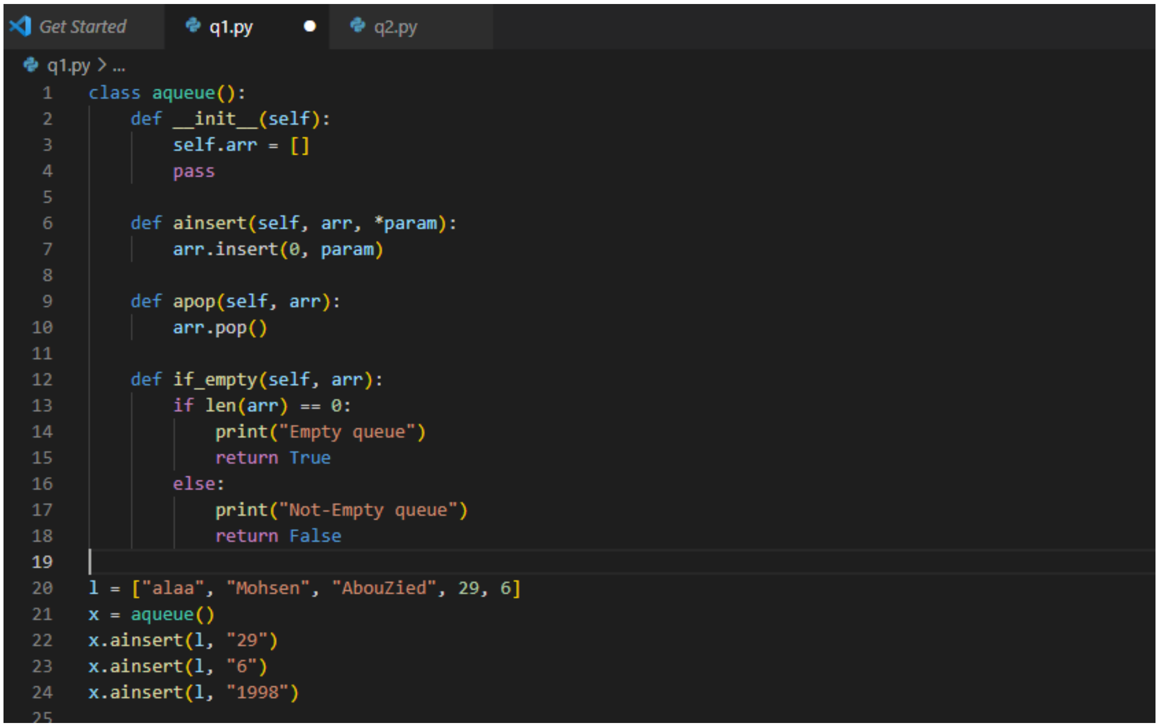
Python

Task: 4

Name: - Alaa Mohsen Mohammed Ahmed AbouZied

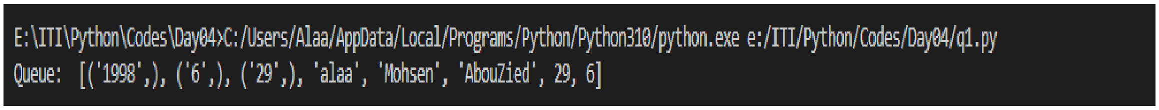
Instructor: - Eng. Noha

(1)



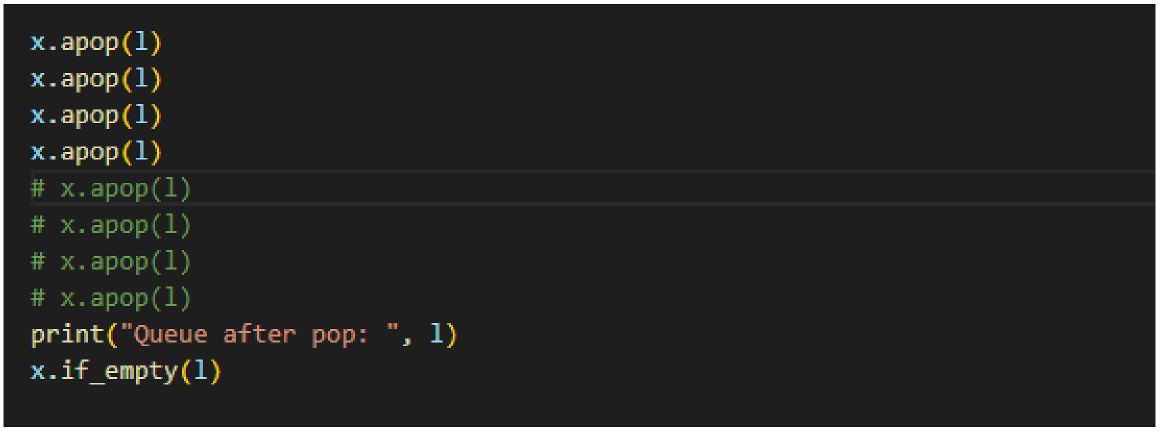
```python
class aqueue():
    def __init__(self):
        self.arr = []
        pass

    def ainsert(self, arr, *param):
        arr.insert(0, param)

    def apop(self, arr):
        arr.pop()

    def if_empty(self, arr):
        if len(arr) == 0:
            print("Empty queue")
            return True
        else:
            print("Not-Empty queue")
            return False

l = ["alaa", "Mohsen", "AbouZied", 29, 6]
x = aqueue()
x.ainsert(l, "29")
x.ainsert(l, "6")
x.ainsert(l, "1998")
```

Run after insert



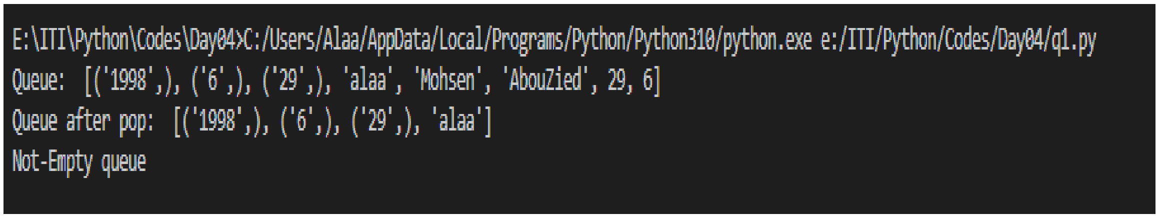
```
E:\ITI\Python\Codes\Day04>C:/Users/Alaa/AppData/Local/Programs/Python/Python310/python.exe e:/ITI/Python/Codes/Day04/q1.py
Queue:  [('1998',), ('6',), ('29',), 'alaa', 'Mohsen', 'AbouZied', 29, 6]
```

(1 cont.)



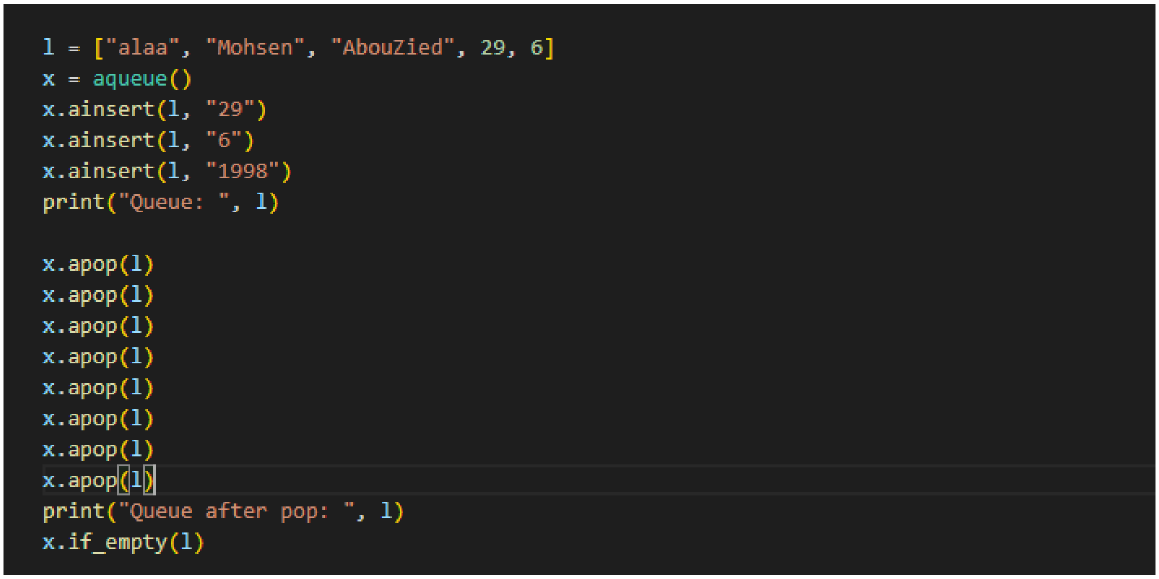
```python
x.apop(l)
x.apop(l)
x.apop(l)
x.apop(l)
# x.apop(l)
# x.apop(l)
# x.apop(l)
# x.apop(l)
print("Queue after pop: ", l)
x.if_empty(l)
```

Run after pop some values



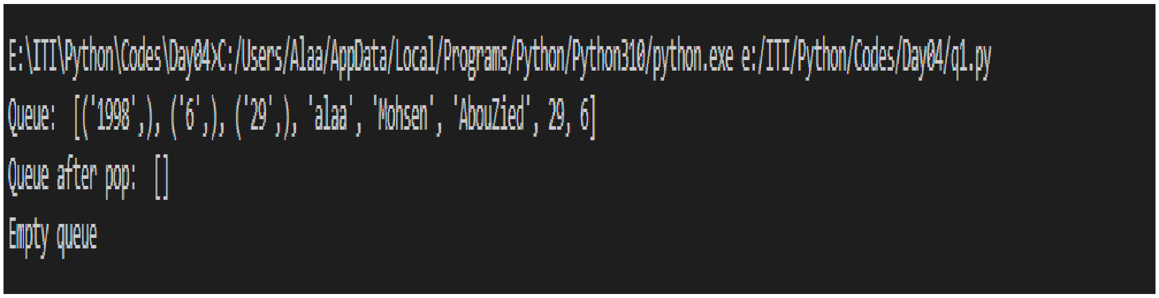
```
E:\ITI\Python\Codes\Day04>C:/Users/Alaa/AppData/Local/Programs/Python/Python310/python.exe e:/ITI/Python/Codes/Day04/q1.py
Queue:  [('1998',), ('6',), ('29',), 'alaa', 'Mohsen', 'AbouZied', 29, 6]
Queue after pop:  [('1998',), ('6',), ('29',), 'alaa']
Not-Empty queue
```

(1 cont.)



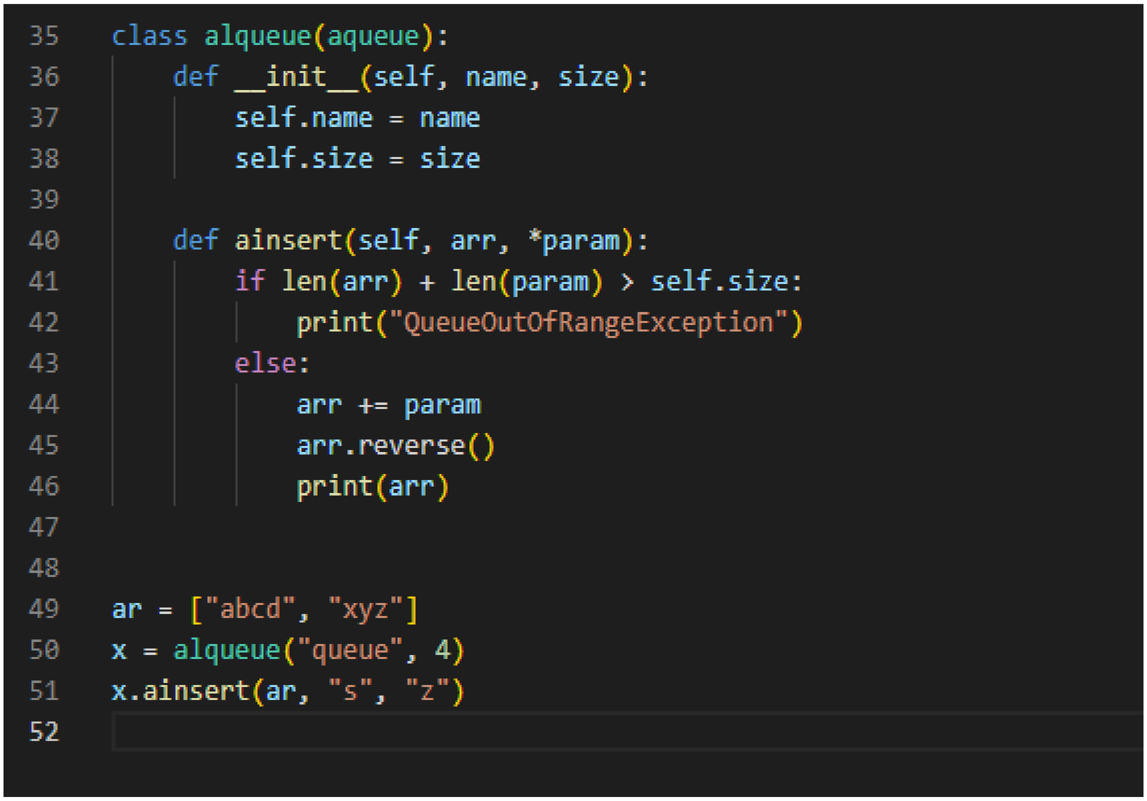
```
l = ["alaa", "Mohsen", "AbouZied", 29, 6]
x = aqueue()
x.ainsert(l, "29")
x.ainsert(l, "6")
x.ainsert(l, "1998")
print("Queue: ", l)

x.apop(l)
x.apop(l)
x.apop(l)
x.apop(l)
x.apop(l)
x.apop(l)
x.apop(l)
x.apop(l)
print("Queue after pop: ", l)
x.if_empty(l)
```

Run after pop all values



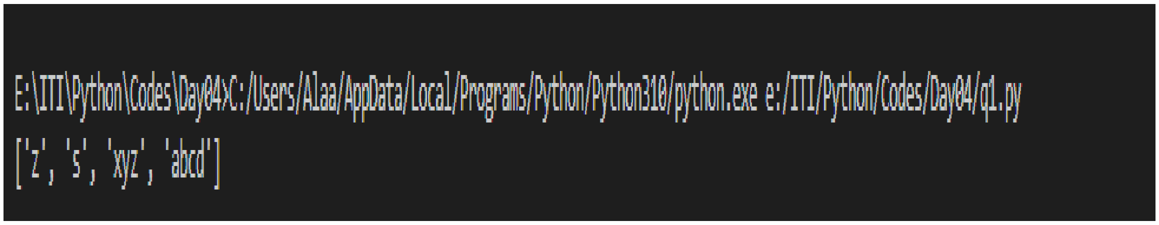
```
E:\ITI\Python\Codes\Day04>C:/Users/Alaa/AppData/Local/Programs/Python/Python310/python.exe e:/ITI/Python/Codes/Day04/q1.py
Queue:  [('1998',), ('6',), ('29',), 'alaa', 'Mohsen', 'AbouZied', 29, 6]
Queue after pop:  []
Empty queue
```

(2)



```python
class alqueue(aqueue):
    def __init__(self, name, size):
        self.name = name
        self.size = size

    def ainsert(self, arr, *param):
        if len(arr) + len(param) > self.size:
            print("QueueOutOfRangeException")
        else:
            arr += param
            arr.reverse()
            print(arr)


ar = ["abcd", "xyz"]
x = alqueue("queue", 4)
x.ainsert(ar, "s", "z")
```

Run



```
E:\ITI\Python\Codes\Day04>C:/Users/Alaa/AppData/Local/Programs/Python/Python310/python.exe e:/ITI/Python/Codes/Day04/q1.py
['z', 's', 'xyz', 'abcd']
```