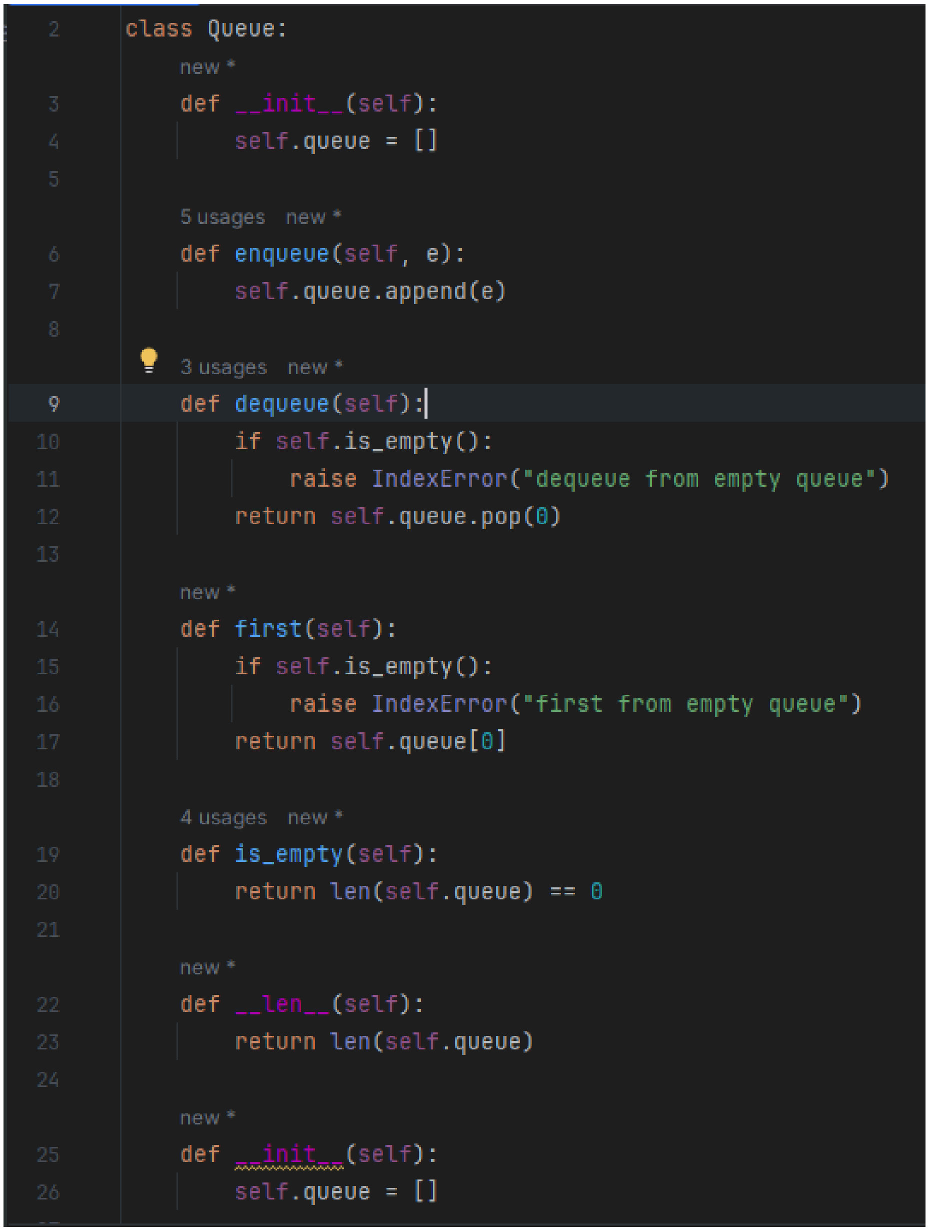
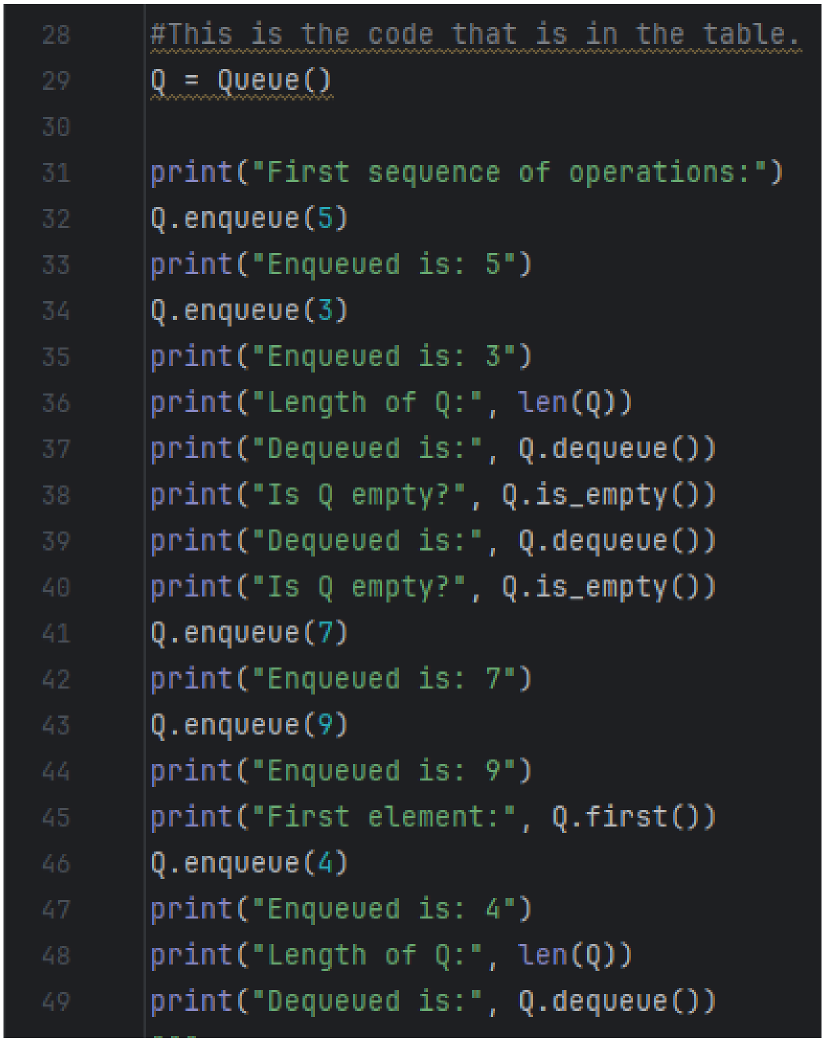
Activity #2 (Midterm)

Create an implementation of the Queue Data Structure:

Code:

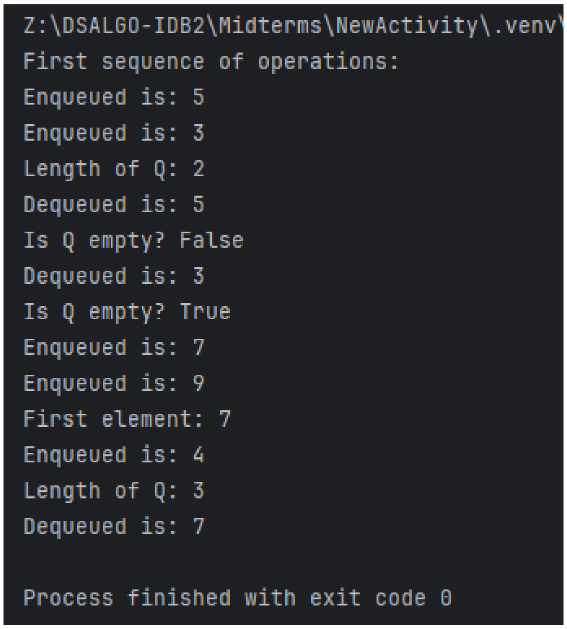


```python
class Queue:
    def __init__(self):
        self.queue = []

    def enqueue(self, e):
        self.queue.append(e)

    def dequeue(self):
        if self.is_empty():
            raise IndexError("dequeue from empty queue")
        return self.queue.pop(0)

    def first(self):
        if self.is_empty():
            raise IndexError("first from empty queue")
        return self.queue[0]

    def is_empty(self):
        return len(self.queue) == 0

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

    def __init__(self):
        self.queue = []
```



```python
#This is the code that is in the table.
Q = Queue()

print("First sequence of operations:")
Q.enqueue(5)
print("Enqueued is: 5")
Q.enqueue(3)
print("Enqueued is: 3")
print("Length of Q:", len(Q))
print("Dequeued is:", Q.dequeue())
print("Is Q empty?", Q.is_empty())
print("Dequeued is:", Q.dequeue())
print("Is Q empty?", Q.is_empty())
Q.enqueue(7)
print("Enqueued is: 7")
Q.enqueue(9)
print("Enqueued is: 9")
print("First element:", Q.first())
Q.enqueue(4)
print("Enqueued is: 4")
print("Length of Q:", len(Q))
print("Dequeued is:", Q.dequeue())
```

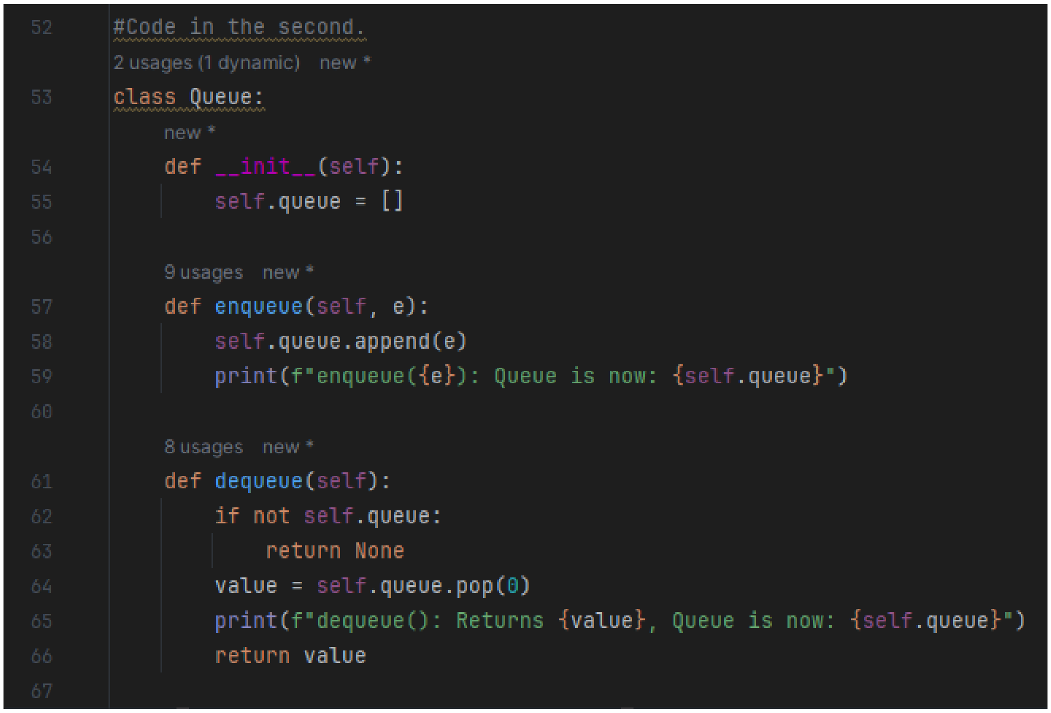
Output:



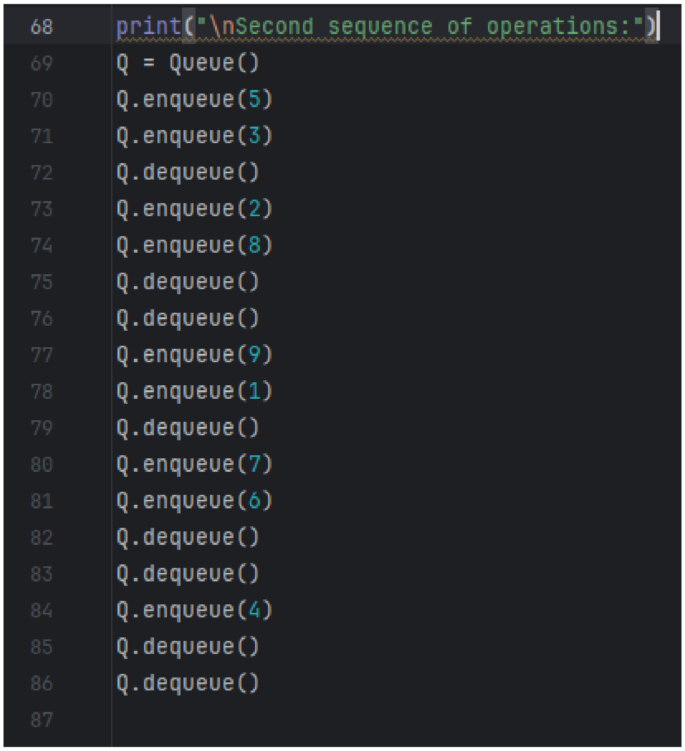
```
Z:\DSALGO-IDB2\Midterms\NewActivity\.venv\
First sequence of operations:
Enqueued is: 5
Enqueued is: 3
Length of Q: 2
Dequeued is: 5
Is Q empty? False
Dequeued is: 3
Is Q empty? True
Enqueued is: 7
Enqueued is: 9
First element: 7
Enqueued is: 4
Length of Q: 3
Dequeued is: 7

Process finished with exit code 0
```

What values are returned during the following sequence of queue operations, if executed on an initially: empty queue? enqueue(5), enqueue(3), dequeue(), enqueue(2), enqueue(8), dequeue(), dequeue(), enqueue(9), enqueue(1), dequeue(), enqueue(7), enqueue(6), dequeue(), dequeue(), enqueue(4), dequeue(), dequeue().

Code:

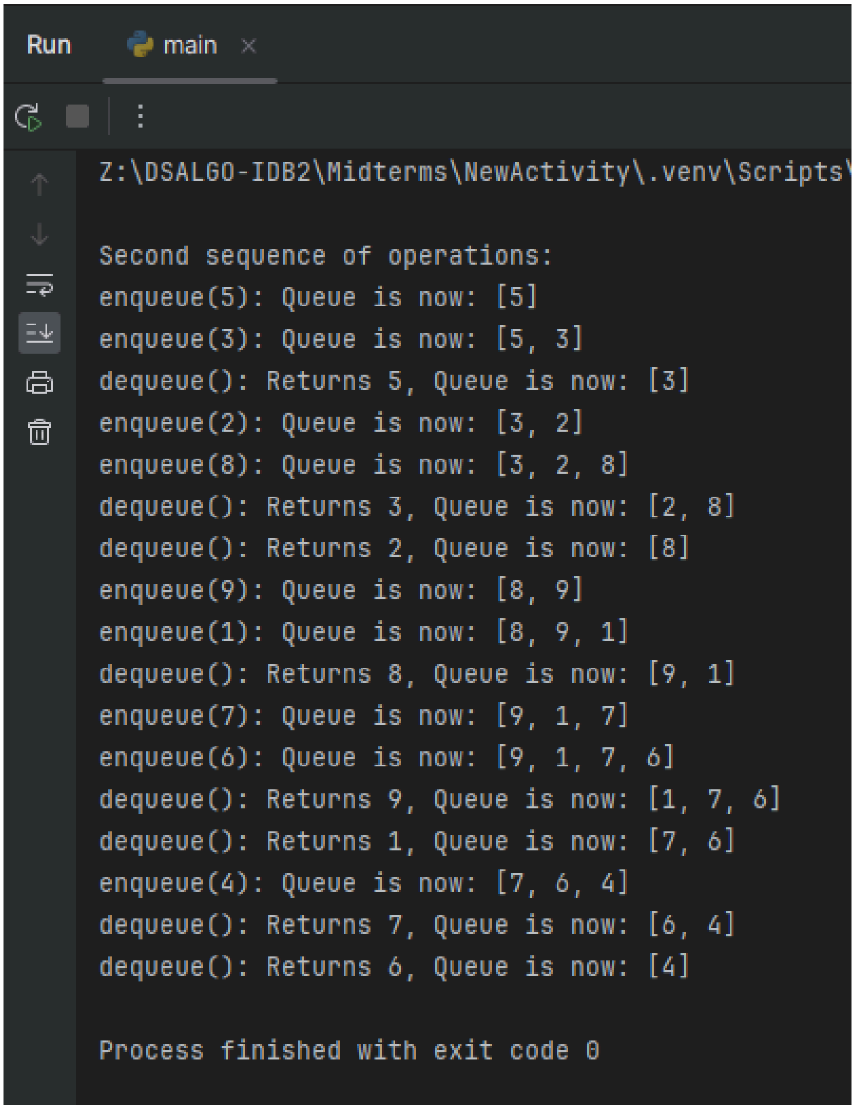


```python
#Code in the second.
class Queue:
    def __init__(self):
        self.queue = []

    def enqueue(self, e):
        self.queue.append(e)
        print(f"enqueue({e}): Queue is now: {self.queue}")

    def dequeue(self):
        if not self.queue:
            return None
        value = self.queue.pop(0)
        print(f"dequeue(): Returns {value}, Queue is now: {self.queue}")
        return value

```



```python
print("\nSecond sequence of operations:")
Q = Queue()
Q.enqueue(5)
Q.enqueue(3)
Q.dequeue()
Q.enqueue(2)
Q.enqueue(8)
Q.dequeue()
Q.dequeue()
Q.enqueue(9)
Q.enqueue(1)
Q.dequeue()
Q.enqueue(7)
Q.enqueue(6)
Q.dequeue()
Q.dequeue()
Q.enqueue(4)
Q.dequeue()
Q.dequeue()

```

Output:



```
Z:\DSALGO-IDB2\Midterms\NewActivity\.venv\Scripts\

Second sequence of operations:
enqueue(5): Queue is now: [5]
enqueue(3): Queue is now: [5, 3]
dequeue(): Returns 5, Queue is now: [3]
enqueue(2): Queue is now: [3, 2]
enqueue(8): Queue is now: [3, 2, 8]
dequeue(): Returns 3, Queue is now: [2, 8]
dequeue(): Returns 2, Queue is now: [8]
enqueue(9): Queue is now: [8, 9]
enqueue(1): Queue is now: [8, 9, 1]
dequeue(): Returns 8, Queue is now: [9, 1]
enqueue(7): Queue is now: [9, 1, 7]
enqueue(6): Queue is now: [9, 1, 7, 6]
dequeue(): Returns 9, Queue is now: [1, 7, 6]
dequeue(): Returns 1, Queue is now: [7, 6]
enqueue(4): Queue is now: [7, 6, 4]
dequeue(): Returns 7, Queue is now: [6, 4]
dequeue(): Returns 6, Queue is now: [4]

Process finished with exit code 0
```