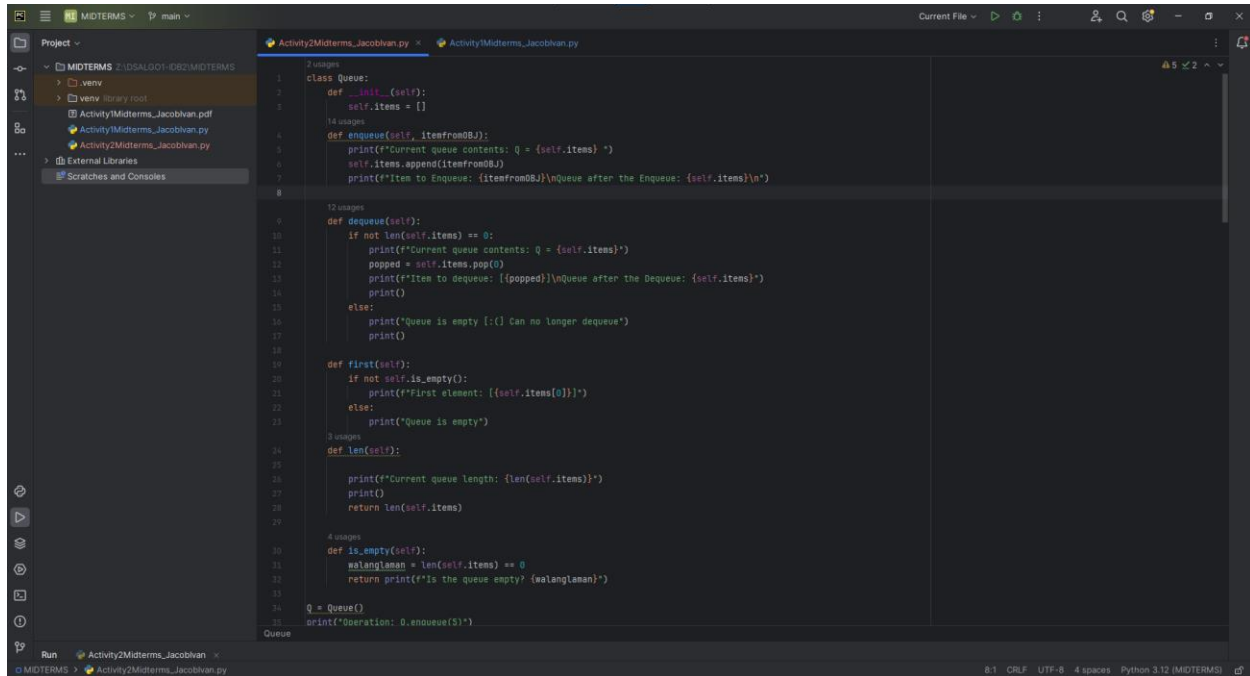
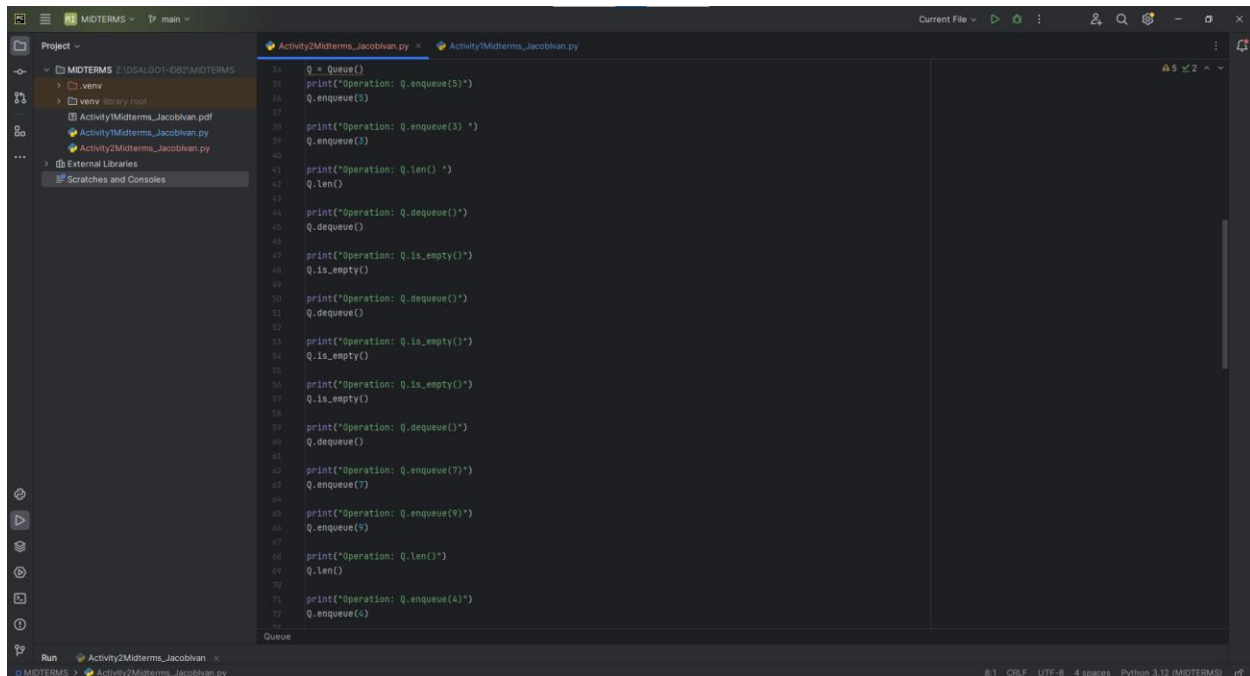


LABORATORY ACTIVITY (2)

CODE :



```
1 class Queue:
2     def __init__(self):
3         self.items = []
4
5     def enqueue(self, itemfromOBJ):
6         print(f"Current queue contents: Q = {self.items}")
7         self.items.append(itemfromOBJ)
8         print(f"Item to Enqueue: {itemfromOBJ}\nQueue after the Enqueue: {self.items}")
9
10    def dequeue(self):
11        if not len(self.items) >= 0:
12            print(f"Current queue contents: Q = {self.items}")
13            popped = self.items.pop(0)
14            print(f"Item to dequeue: [{popped}]\nQueue after the Dequeue: {self.items}")
15            print()
16        else:
17            print("Queue is empty :: Can no longer dequeue")
18            print()
19
20    def first(self):
21        if not self.is_empty():
22            print(f"First element: {self.items[0]}")
23        else:
24            print("Queue is empty")
25
26    def len(self):
27        print(f"Current queue length: {len(self.items)}")
28        print()
29        return len(self.items)
30
31    def is_empty(self):
32        malanglaman = len(self.items) == 0
33        return print(f"Is the queue empty? {malanglaman}")
34
35 Q = Queue()
36 print("Operation: Q.enqueue(5)")
```



```
37 Q = Queue()
38 print("Operation: Q.enqueue(5)")
39 Q.enqueue(5)
40
41 print("Operation: Q.enqueue(3)")
42 Q.enqueue(3)
43
44 print("Operation: Q.len()")
45 Q.len()
46
47 print("Operation: Q.dequeue()")
48 Q.dequeue()
49
50 print("Operation: Q.is_empty()")
51 Q.is_empty()
52
53 print("Operation: Q.dequeue()")
54 Q.dequeue()
55
56 print("Operation: Q.is_empty()")
57 Q.is_empty()
58
59 print("Operation: Q.dequeue()")
60 Q.dequeue()
61
62 print("Operation: Q.enqueue(7)")
63 Q.enqueue(7)
64
65 print("Operation: Q.enqueue(9)")
66 Q.enqueue(9)
67
68 print("Operation: Q.len()")
69 Q.len()
70
71 print("Operation: Q.enqueue(4)")
72 Q.enqueue(4)
```

The screenshot displays a Jupyter Notebook environment within a code editor. The left sidebar shows the project structure, including a folder named 'MIDTERMS' and a file named 'Activity2Midterms_Jacobvian.py'. The main editor area shows the code for 'Activity2Midterms_Jacobvian.py', which implements a queue simulation. The code includes comments for 'PART TWO SIMULATION' and uses a 'Queue' class to manage operations. The right sidebar shows the 'Run' button and the 'Queue' output.

```
70 print("Operation: Q.enqueue(4)")
71 Q.enqueue(4)
72
73 print("Operation: Q.len()")
74 Q.len()
75
76
77 print("Operation: Q.dequeue()")
78 Q.dequeue()
79
80 print(" * =====> End of Simulation 1 <===== *")
81 "PART TWO SIMULATION"
82
83 Q = Queue()
84 print("Operation: Q2.enqueue(5) ")
85 Q2.enqueue(5)
86
87 print("Operation: Q2.enqueue(3) ")
88 Q2.enqueue(3)
89
90 print("Operation: Q2.dequeue() ")
91 Q2.dequeue()
92
93 print("Operation: Q2.enqueue(2) ")
94 Q2.enqueue(2)
95
96 print("Operation: Q2.enqueue(8) ")
97 Q2.enqueue(8)
98
99 print("Operation: Q2.dequeue()")
100 Q2.dequeue()
101
102 print("Operation: Q2.dequeue()")
103 Q2.dequeue()
104
105 print("Operation: Q2.enqueue(9)")
106 Q2.enqueue(9)
107
108 print("Operation: Q2.enqueue(1)")
109 Q2.enqueue(1)
```

Queue

```
Project ▾ Activity2Midterms_Jacoblvn.py
Run ▾ Activity2Midterms_Jacoblvn.py
Z:\DSAL001-1082\MIDTERMS\venv\Scripts\python.exe Z:\DSAL001-1082\MIDTERMS\Activity2Midterms_Jacoblvn.py
:
Operation: Q.enqueue(5)
Current queue contents: Q = []
Item to Enqueue: 5
Queue after the Enqueue: [5]

Operation: Q.enqueue(3)
Current queue contents: Q = [5]
Item to Enqueue: 3
Queue after the Enqueue: [5, 3]

Operation: Q.len()
Current queue length: 2

Operation: Q.dequeue()
Current queue contents: Q = [5, 3]
Item to dequeue: [5]
Queue after the Dequeue: [3]

Operation: Q.is_empty()
Is the queue empty? False
Operation: Q.dequeue()
Current queue contents: Q = [3]
Item to dequeue: [3]
Queue after the Dequeue: []

Operation: Q.is_empty()
Is the queue empty? True
Operation: Q.is_empty()
Is the queue empty? True
Operation: Q.dequeue()
Queue is empty [::] Can no longer dequeue

Operation: Q.enqueue(7)
Current queue contents: Q = []
Item to Enqueue: 7
Queue after the Enqueue: [7]

Operation: Q.enqueue(9)
```

```
Project ▾ Activity2Midterms_Jacoblvn.py
Run ▾ Activity2Midterms_Jacoblvn.py
:
Operation: Q.enqueue(9)
Current queue contents: Q = [7]
Item to Enqueue: 9
Queue after the Enqueue: [7, 9]

Operation: Q.len()
Current queue length: 2

Operation: Q.enqueue(4)
Current queue contents: Q = [7, 9]
Item to Enqueue: 4
Queue after the Enqueue: [7, 9, 4]

Operation: Q.len()
Current queue length: 3

Operation: Q.dequeue()
Current queue contents: Q = [7, 9, 4]
Item to dequeue: [7]
Queue after the Dequeue: [9, 4]

* ***** End of Simulation 1 ***** *
Operation: Q2.enqueue(5)
Current queue contents: Q = []
Item to Enqueue: 5
Queue after the Enqueue: [5]

Operation: Q2.enqueue(3)
Current queue contents: Q = [5]
Item to Enqueue: 3
Queue after the Enqueue: [5, 3]

Operation: Q2.dequeue()
Current queue contents: Q = [5, 3]
Item to dequeue: [5]
Queue after the Dequeue: [3]

Operation: Q2.enqueue(2)
Current queue contents: Q = [3]
```

