Array Implementation:

SIZE = 5

queue = [None] \* SIZE

front = -1

rear = -1

def isEmpty():

return front == -1 or front > rear

def isFull():

return rear == SIZE – 1

def enqueue(value):

global rear, front

if isFull():

print("Queue is FULL!!! Insertion not possible!")

return

if front == -1:

front = 0

rear += 1

queue[rear] = value

print(f"Inserted {value} into queue.")

def dequeue():

global front, rear

if isEmpty():

print("Queue is EMPTY!!!")

return

print(f"Deleted element: {queue[front]}")

front += 1

if front > rear:

front = rear = -1

def size():

if isEmpty():

return 0

return rear - front + 1

def show():

if isEmpty():

print("Queue is EMPTY!!!")

else:

print("Queue elements are:", end=" ")

for i in range(front, rear + 1):

print(queue[i], end=" ")

print()

while True:

print("\n=== Queue Operations ===")

print("1. Enqueue")

print("2. Dequeue")

print("3. Check if Queue is Empty")

print("4. Get Queue Size")

print("5. Show Queue")

print("6. Exit")

choice = input("Enter your choice (1-6): ")

if choice == '1':

value = input("Enter the value to insert: ")

enqueue(value)

elif choice == '2':

dequeue()

elif choice == '3':

print("Queue is Empty" if isEmpty() else "Queue is NOT Empty")

elif choice == '4':

print(f"Queue size = {size()}")

elif choice == '5':

show()

elif choice == '6':

print("Exiting program.")

break

else:

print("Invalid choice! Please enter between 1-6.")

OUTPUT:

=== Queue Operations ===

1. Enqueue

2. Dequeue

3. Check if Queue is Empty

4. Get Queue Size

5. Show Queue

6. Exit

Enter your choice (1-6): 1

Enter the value to insert: PORS7777

Inserted PORS7777 into queue.

=== Queue Operations ===

1. Enqueue

2. Dequeue

3. Check if Queue is Empty

4. Get Queue Size

5. Show Queue

6. Exit

Enter your choice (1-6): 1

Enter the value to insert: RR4567

Inserted RR4567 into queue.

=== Queue Operations ===

1. Enqueue

2. Dequeue

3. Check if Queue is Empty

4. Get Queue Size

5. Show Queue

6. Exit

Enter your choice (1-6): 1

Enter the value to insert: BENZ0000

Inserted BENZ0000 into queue.

=== Queue Operations ===

1. Enqueue

2. Dequeue

3. Check if Queue is Empty

4. Get Queue Size

5. Show Queue

6. Exit

Enter your choice (1-6): 3

Queue is NOT Empty

=== Queue Operations ===

1. Enqueue

2. Dequeue

3. Check if Queue is Empty

4. Get Queue Size

5. Show Queue

6. Exit

Enter your choice (1-6): 4

Queue size = 3

=== Queue Operations ===

1. Enqueue

2. Dequeue

3. Check if Queue is Empty

4. Get Queue Size

5. Show Queue

6. Exit

Enter your choice (1-6): 5

Queue elements are: PORS7777 RR4567 BENZ0000

=== Queue Operations ===

1. Enqueue

2. Dequeue

3. Check if Queue is Empty

4. Get Queue Size

5. Show Queue

6. Exit

Enter your choice (1-6): 2

Deleted element: PORS7777

=== Queue Operations ===

1. Enqueue

2. Dequeue

3. Check if Queue is Empty

4. Get Queue Size

5. Show Queue

6. Exit

Enter your choice (1-6): 6

Exiting program.

Linked List Implementation:

class Node:

def \_\_init\_\_(self, data):

self.data = data

self.next = None

class Queue:

def \_\_init\_\_(self):

self.front = None

self.rear = None

def enqueue(self, value):

newNode = Node(value)

if self.rear is None:

self.front = self.rear = newNode

else:

self.rear.next = newNode

self.rear = newNode

print(f"{value} enqueued to queue.")

def dequeue(self):

if self.front is None:

print("Queue is Empty!")

return

temp = self.front

self.front = self.front.next

if self.front is None:

self.rear = None

print(f"{temp.data} dequeued from queue.")

def display(self):

if self.front is None:

print("Queue is Empty!")

return

temp = self.front

print("Queue Elements:")

while temp:

print(f"{temp.data} --> ", end="")

temp = temp.next

print("NULL")

if \_\_name\_\_ == "\_\_main\_\_":

q = Queue()

while True:

print("\nMenu:")

print("1. Enqueue")

print("2. Dequeue")

print("3. Display")

print("4. Exit")

choice = input("Enter your choice: ")

if choice == '1':

value = input("Enter value to enqueue: ")

q.enqueue(value)

elif choice == '2':

q.dequeue()

elif choice == '3':

q.display()

elif choice == '4':

print("Exiting program.")

break

else:

print("Invalid choice. Please try again.")

OUTPUT:

Menu:

1. Enqueue

2. Dequeue

3. Display

4. Exit

Enter your choice: 1

Enter value to enqueue: PORS7777

PORS7777 enqueued to queue.

Menu:

1. Enqueue

2. Dequeue

3. Display

4. Exit

Enter your choice: 1

Enter value to enqueue: BENZ4567

BENZ4567 enqueued to queue.

Menu:

1. Enqueue

2. Dequeue

3. Display

4. Exit

Enter your choice: 1

Enter value to enqueue: RR0000

RR0000 enqueued to queue.

Menu:

1. Enqueue

2. Dequeue

3. Display

4. Exit

Enter your choice: 3

Queue Elements:

PORS7777 --> BENZ4567 --> RR0000 --> NULL

Menu:

1. Enqueue

2. Dequeue

3. Display

4. Exit

Enter your choice: 2

PORS7777 dequeued from queue.

Menu:

1. Enqueue

2. Dequeue

3. Display

4. Exit

Enter your choice: 4

Exiting program.