| Name: | Advait Dhakad |
| --- | --- |
| Roll No: | 1510 |
| Title of Program: | Stacks & Queues: Double ended Queues |
| Objective: | 1. EnqueueFront 2. EnqueueRea 3. DequeueFront 4. DequeueRear 5. PeekFront 6. PeekRear 7. Display |

**CODE:**

import java.util.Scanner;

class DQNode {

int data;

DQNode right;

DQNode left;

public DQNode(int d) {

data = d;

right = null;

left = null;

}

} // end of dequeue

class DQue {

DQNode front;

DQNode rear;

public DQue() {

front = null;

rear = null;

}

void EnqueueFront(int data) {

DQNode x = new DQNode(data);

if (front == null) {

front = x;

rear = x;

} else {

x.right = front;

front.left = x;

front = x;

}

}

void EnqueueRear(int data) {

DQNode x = new DQNode(data);

if (front == null) {

front = x;

rear = x;

} else {

x.left = rear;

rear.right = x;

rear = x;

}

} // end of Enqueue

public void DequeueFront() {

if (front == null) {

System.out.println("Queue UnderFlow !!!");

} else {

System.out.println("Element Removed: " + front.data);

if (front == rear) {

front = null;

rear = null;

} else {

front = front.right;

front.left = null;

}

}

}

// DequeueRear

public void DequeueRear() {

System.out.println("Element Removed: " + rear.data);

if (front == null) {

System.out.println("Queue UnderFlow !!!");

} else {

if (front == rear) {

front = null;

rear = null;

} else {

rear = rear.left;

rear.right = null;

}

}

}

// PeekFront

public void PeekFront() {

if (front == null) {

System.out.println("Queue UnderFlow !!!");

} else {

System.out.println(front.data);

}

}

// PeekRear

public void PeekRear() {

if (front == null) {

System.out.println("Queue UnderFlow !!!");

} else {

System.out.println(rear.data);

}

}

// Display

public void Display() {

if (front == null) {

System.out.println("Queue UnderFlow !!!");

} else {

DQNode tmp = front;

while (tmp != null) {

System.out.print(tmp.data + " | " + " ");

tmp = tmp.right;

}

System.out.println("");

}

}

}// end of DQue

class DQueue {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

DQue d = new DQue();

char ch;

do {

System.out.print("\033[H\033[2J");

System.out.flush();

System.out.println("\n\t\*\*\*\*\*\*\*\*\*\*Double Ended Queue\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("1. Enqueue Front");

System.out.println("2. Enqueue Rear");

System.out.println("3. Dequeue Front");

System.out.println("4. Dequeue Rear");

System.out.println("5. Peek Front");

System.out.println("6. Peek Rear");

System.out.println("7. Display");

System.out.print("Enter your Choice: ");

int choice = sc.nextInt();

switch (choice) {

case 1:

System.out.println("You selected Enqueue Front.");

System.out.println("Enter the Value: ");

int val = sc.nextInt();

d.EnqueueFront(val);

break;

case 2:

System.out.println("You selected Enqueue Rear.");

System.out.println("Enter the Value: ");

int val1 = sc.nextInt();

d.EnqueueRear(val1);

break;

case 3:

System.out.println("You selected Dequeue Front.");

d.DequeueFront();

d.Display();

break;

case 4:

System.out.println("You selected Dequeue Rear.");

d.DequeueRear();

d.Display();

break;

case 5:

System.out.println("You selected Peek Front. ");

d.PeekFront();

break;

case 6:

System.out.println("You selected Peek Rear.");

d.PeekRear();

break;

case 7:

System.out.println("You selected Display.");

d.Display();

break;

default:

System.out.println("Incorrect Choice !!!");

break;

}

System.out.print("Do you want to countinue(y/n): ");

ch = sc.next().charAt(0);

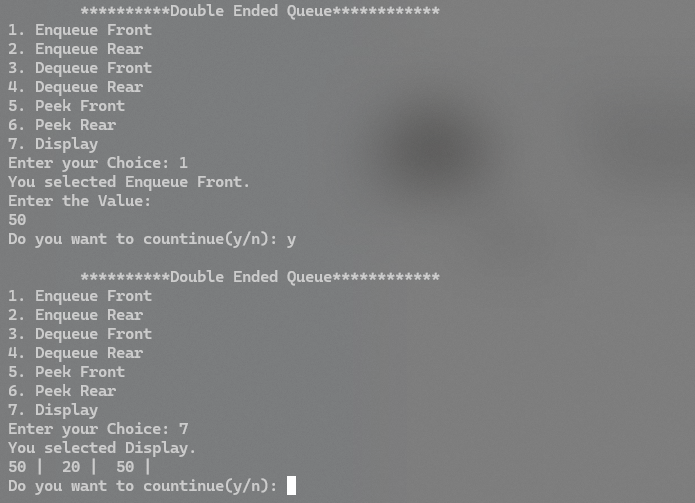
} while (ch == 'y' || ch == 'Y');// end of do while

}// end of main

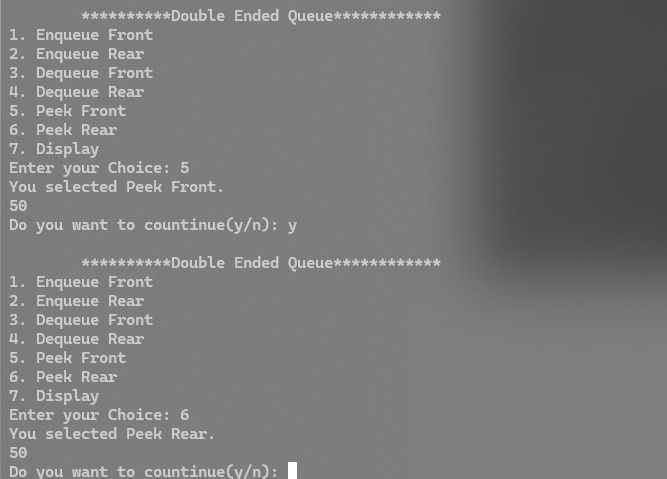
} // end of DQueue

**OUTPUT:**

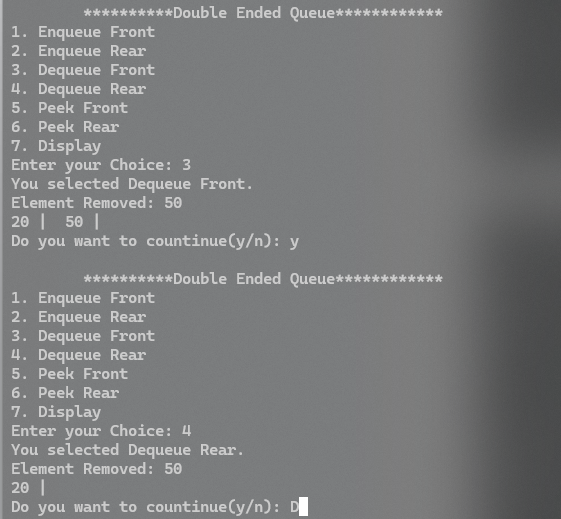
**Enqueue front and rear**

****

**Peek Front and rear**

****

**Dequeue front and rear:**

****