class node

{

int data;

node next;

node(int d)

{

data=d;

next=null;

}

}

class queue

{

node front=null;

node rear=null;

boolean isempty()

{

return(rear==null&front==null);

}

void enqueue(int d)

{

node nnode=new node(d);

if(front==null&&rear==null)

{

front=nnode;

rear=nnode;

}

else

{

rear.next=nnode;

rear=nnode;

}

return;

}

int dequeue()

{

if(isempty())

{

System.out.println("Queue is empty");

return Integer.MIN\_VALUE;

}

else

{

int d=front.data;

front=front.next;

if(front==null)

rear=null;

return d;

}

}

void display()

{

if(isempty())

{

System.out.println("Queue is empty");

}

else

{

node temp=front;

while(temp!=null)

{

System.out.print(temp.data+" ");

temp=temp.next;

}

System.out.println("\n");

}

}

}

public class Main

{

public static void main(String args[])

{

queue q1=new queue();

q1.enqueue(45);

q1.enqueue(90);

q1.enqueue(21);

q1.display();

q1.dequeue();

q1.display();

q1.enqueue(78);

q1.display();

q1.dequeue();

q1.dequeue();

q1.dequeue();

q1.display();

q1.enqueue(45);

q1.enqueue(90);

q1.enqueue(21);

q1.display();

q1.dequeue();

q1.display();

}

}