# Program:

package exception.handling;

import java.util.Scanner;

import java.lang.Exception;

class OverFlowException extends Exception

{

String msg=null;

public OverFlowException(String msg)

{

this.msg=msg;

}

}

class UnderFlowException extends Exception

{

String msg=null;

public UnderFlowException(String msg)

{

this.msg=msg;

}

}

class ExceptioninQueue

{

public static void main(String[] args) throws UnderFlowException, OverFlowException

{

Scanner in=new Scanner(System.in);

Queue q = new Queue();

int ch, val, op=1;

System.out.println("Note: Queue Size is 10");

do

{

System.out.println("\nEnter Your Choice");

System.out.println("1.Enqueue\n2.Dequeue\n3.Display\n4.Exit");

ch=in.nextInt();

switch(ch)

{

case 1 :

q.Add();

break;

case 2 :

val=q.Delete();

if(val!=0)

System.out.println("\nDeleted Element: "+val);

break;

case 3 :

q.display();

break;

case 4:

op=0;

break;

default:

System.out.println("\nInvalid Option! Try Again...");

}

}while(op==1);

}

}

public class Queue

{

Scanner in=new Scanner(System.in);

int MAX=10;

int arr[]=new int[MAX];

int front=-1, rear=-1;

void Add() throws OverFlowException

{

try

{

if(rear == MAX-1)

throw new OverFlowException("Queue Full");

rear++;

System.out.println("\nEnter Element");

arr[rear] = in.nextInt();

if( front == -1 )

front = 0;

}

catch(OverFlowException e)

{

System.out.println("Exception Received: "+e.msg);

}

}

int Delete() throws UnderFlowException

{

int data = 0;

try

{

if(front == -1)

throw new UnderFlowException("Queue Empty");

data = arr[front];

if( front == rear)

front = rear = -1;

else

front++;

}

catch(UnderFlowException e)

{

System.out.println("Exception Received: "+e.msg);

}

return data;

}

void display() throws UnderFlowException

{

try

{

int i;

if(front == -1)

throw new UnderFlowException("Queue Empty");

System.out.println("The Elements In Queue Are: ");

for(i=front;i<=rear;i++)

System.out.println(arr[i]+" ");

}

catch(UnderFlowException e)

{

System.out.println("Exception Received: "+e.msg);

}

}

}

# Output:

Note: Queue Size is 10

Enter Your Choice

1.Enqueue

2.Dequeue

3.Display

4.Exit

1

Enter Element

56

Enter Your Choice

1.Enqueue

2.Dequeue

3.Display

4.Exit

1

Enter Element

9

Enter Your Choice

1.Enqueue

2.Dequeue

3.Display

4.Exit

3

The Elements In Queue Are:

56

9

Enter Your Choice

1.Enqueue

2.Dequeue

3.Display

4.Exit

2

Deleted Element: 56

Enter Your Choice

1.Enqueue

2.Dequeue

3.Display

4.Exit

3

The Elements In Queue Are:

9

Enter Your Choice

1.Enqueue

2.Dequeue

3.Display

4.Exit

4

BUILD SUCCESSFUL (total time: 38 seconds)