INFORMATICS INSTITUTE OF TECHNOLOGY

Software Development II

Coursework Report

Name-S. Chamodi Kavindhya De Silva

UoW-w1898939

IIT ID-20211323

Contents

Task 01 – Source Code

w1898939_java_cw_part1.java

```
import java.io.*;
import java.util.Scanner;
import static java.lang.System.exit;

public class w1898939_java_cw_part1 {
    static String [] queue1 = new String[6];
    static String [] queue2 = new String[6];
    static String [] queue3 = new String[6];
    static int x1=10;
    static int x2=10;
    static int x3=10;
    static int one=0;
    static int two=0;
    static int three=0;
    static int tot=0;
    static int tot2=0;
    static int tot3=0;
    static int x,y,z,e,size,j,size1,size2,p;
    static int c=0;
    static int t=0;
    static int v=0;
    static int w=0;
    static String temp,temp1,temp2;
    static int fuel=6600;
    static int remain=0;
    public static void Menu(){
        //Display the Menu
        System.out.println("\n");
        System.out.println("------------------Fuel Management System-
--------------\n");
        System.out.println("\t100 or VFQ: View all Fuel Queues");
        System.out.println("\t101 or VEQ: View all Empty Queues");
        System.out.println("\t102 or ACQ: Add customer to a Queue");
        System.out.println("\t103 or RCQ: Remove a customer from a
Queue");
        System.out.println("\t104 or PCQ: Remove a served customer");
        System.out.println("\t105 or VCS: View Customers Sorted in
alphabetical order");
        System.out.println("\t106 or SPD: Store Program Data into
file");
        System.out.println("\t107 or LPD: Load Program Data from
file");
        System.out.println("\t108 or STK: View Remaining Fuel
Stock");
        System.out.println("\t109 or AFS: Add Fuel Stock");
        System.out.println("\t999 or EXT: Exit the Program\n");

    }


    public static void Repeat() {
        System.out.print("Enter 1=Menu  0=Exit: ");
        Scanner no = new Scanner(System.in);
        String view = no.next();
        switch (view) {
            case "1":
                start();
```

```
            case "0":
                exit(0);

        }
    }
    public static void ViewFuelQueue()
    {
        //View all fuel queues details
        x=one;
        System.out.println("Queue 1:");
        for (x = 0; x <6; x++) {

            if(queue1[x]!="Empty") {

                System.out.println("Customer id:" + x+ " " +
"Customer Name:" + queue1[x]);
            }
        }
        System.out.println("Total liters= " + tot);

        y=two;
        System.out.println("Queue 2:");
        for (y = 0; y < 6; y++) {
            if(queue2[y]!="Empty") {
                System.out.println("Customer id:" + y + " " +
"Customer Name:" + queue2[y]);
            }
        }
        System.out.println("Total liters= " + tot2);

        z=three;
        System.out.println("Queue 3:");
        for (z = 0; z < 6; z++) {
            if(queue3[z]!="Empty") {
                System.out.println("Customer id:" + z + " " +
"Customer Name:" + queue3[z]);
            }
        }
        System.out.println("Total liters= " + tot3);
        System.out.println("\n");
        Repeat();
    }
    public static void ViewEmptyQueue()
    {//Display empty queues
        if(tot==0)
        {
            System.out.println("Queue1 is empty");
        }
        if(tot2==0)
        {
            System.out.println("Queue2 is empty");
        }
        if(tot3==0)
        {
            System.out.println("Queue3 is empty");
        }
        Repeat();
    }

    public static void add()
    {//add customer to the queues
```

```
        System.out.print("Do you want to add a customer(Y/N):");
        Scanner customer=new Scanner(System.in);
        String answer1=customer.next();
        String answer=answer1.toUpperCase();
        switch (answer){
            case "Y":
                System.out.print("Enter the Queue (1 or 2 or 3): ");
                Scanner add=new Scanner(System.in);
                String num=add.next();
                switch (num){
                    case "1":
                        while (one<=5) {
                            System.out.print("Enter Name: ");
                            Scanner name1=new Scanner(System.in);
                            String Cname=name1.next();
                            queue1[one] = Cname;
                            tot = tot + x1;
                            System.out.println(Cname+" is
successfully added to Queue1"+ " " + "\t\t\tTotal liters= "+tot);
                            if(one==5){
                                System.out.println("Queue 1 is
full");
                                add();
                            }else {
                                one = one + 1;
                                add();
                            }
                        }

                    case "2":
                        while (two<=5) {
                            System.out.print("Enter Name: ");
                            Scanner name2=new Scanner(System.in);
                            String Cname2=name2.next();
                            queue2[two] = Cname2;
                            tot2 = tot2 + x2;
                            System.out.println(Cname2+" is
successfully added to Queue2"+" " +"\t\t\tTotal Litres= "+ tot2);
                            if(two==5){
                                System.out.println("Queue 2 is
full");
                                add();

                            }else {
                                two = two + 1;
                                add();
                            }
                        }


                    case "3":
                        while (three<=5) {
                            System.out.print("Enter Name: ");
                            Scanner name3=new Scanner(System.in);
                            String Cname3=name3.next();
                            queue3[three] = Cname3;
                            tot3 = tot3 + x3;
                            System.out.println(Cname3+" is
successfully added to Queue3"+ " " +"\t\t\tTotal Liters= "+ tot3);
                            if(three==5){
                                System.out.println("Queue 3 is
```

```
full");
                                add();
                            }else {
                                three = three + 1;
                                add();
                            }
                        }


                }
            case "N":
                Repeat();


        }


    }

    public static void RemoveCustomer()
    {// remove customers from the queues
        //E,X1,X2,X3
        System.out.print("Enter the Queue no:");
        Scanner que=new Scanner(System.in);
        String QueName=que.next();
        System.out.print("Enter the ID of Customer:");
        Scanner id=new Scanner(System.in);
        String idCustomer=id.next();
        e=Integer.parseInt(idCustomer);
        switch (QueName)
        {
            case "1":
                System.out.println("Queue 01:");
                System.out.println("CustomerID:" + e+ " CustomerName:
" + queue1[e]+" is successfully removed");
                queue1[e] ="Empty";
                for(int y=e;y<6;y++){
                    if(queue1[e+1]!="Empty"){

                        queue1[e]=queue1[e+1];
                        queue1[e+1] = "Empty";

                    }}


                tot = tot - x1;
                System.out.println("\n");
                Repeat();
            case "2":
                System.out.println("Queue 02:");
                System.out.println("CustomerID:"+e+" CustomerName:
"+queue2[e]+" is successfully removed");
                queue2[e]= "Empty";
                for(int y=e;y<6;y++){
                    if(queue2[e+1]!="Empty"){

                        queue2[e]=queue2[e+1];
                        queue2[e+1] = "Empty";

                    }}
```

```
                tot2=tot2-x2;
                System.out.println("\n");
                Repeat();
            case "3":
                System.out.println("Queue 03:");
                System.out.println("CustomerID:"+e+" CustomeName:
"+queue3[e]+" is successfully removed");
                queue3[e]= "Empty";
                for(int y=e;y<6;y++){
                    if(queue3[e+1]!="Empty"){

                        queue3[e]=queue3[e+1];
                        queue3[e+1] = "Empty";

                    }}

                tot3=tot3-x3;
                System.out.println("\n");
                Repeat();

        }

    }

    public static void RemoveServed()
    {
        //remove the first customer of the queue
        System.out.print("Enter the Queue no:");
        Scanner queNo=new Scanner(System.in);
        String QName=queNo.next();
        switch (QName)
        {
            case "1":

                System.out.println("Queue1-CustomerId:"+p+"
CustomerName: "+queue1[p]+" is successfully removed");
                queue1[p]= "Empty";
                for(int e=0;e<5;e++){
                    if(queue1[e+1]!="Empty"){

                        queue1[e]=queue1[e+1];
                        queue1[e+1] = "Empty";

                    }}

               // p=p+1;
                Repeat();
            case "2":

                System.out.println("Queue2-CustomerId:"+t+"
CustomerName: "+queue2[t]+" is successfully removed");
                queue2[t]="Empty";
                for(int e=0;e<5;e++){
                    if(queue1[e+1]!="Empty"){

                        queue1[e]=queue1[e+1];
                        queue1[e+1] = "Empty";

                    }}
               // t=t+1;
                Repeat();
```

```
            case "3":

                System.out.println("Queue3-CustomerId:"+v+"
CustomerName: "+queue3[v]+" is successfully removed");
                queue3[v]="Empty";
                for(int e=0;e<5;e++){
                    if(queue1[e+1]!="Empty"){

                        queue1[e]=queue1[e+1];
                        queue1[e+1] = "Empty";

                    }}
                //v=v+1;
                Repeat();


        }


    }

    public static void ViewCustomerAlphabeticOrder()
    {//Display the customers name in alphabetical order

        size = queue1.length;

        for (int a = 0; a < size - 1; a++) {

            for (j = a + 1; j < queue1.length; j++) {
                if (queue1[a].compareTo(queue1[j]) > 0) {
                    temp = queue1[a];
                    queue1[a] = queue1[j];
                    queue1[j] = temp;
                }
            }

        }
        System.out.println("Queue 1:");
        for (int r=0;r<6;r++) {
            if(queue1[r]!="Empty") {
                System.out.print(queue1[r]+" ");
            }
        }



        size1 = queue2.length;

        for (int a = 0; a < size1 - 1; a++) {

            for (j = a + 1; j < queue2.length; j++) {
                if (queue2[a].compareTo(queue2[j]) > 0) {
                    temp1 = queue2[a];
                    queue2[a] = queue2[j];
                    queue2[j] = temp1;
                }
            }

        }
        System.out.println("\nQueue 2:");
```

```
        for (int r=0;r<6;r++) {
            if(queue2[r]!="Empty") {
                System.out.print(queue2[r]+" ");
            }
        }
        size2 = queue3.length;

        for (int a = 0; a < size2 - 1; a++) {

            for (j = a + 1; j < queue3.length; j++) {
                if (queue3[a].compareTo(queue3[j]) > 0) {
                    temp2 = queue3[a];
                    queue3[a] = queue3[j];
                    queue3[j] = temp2;
                }
            }

        }

        System.out.println("\nQueue 3:");

        for(int r=0;r<6;r++) {
            if (queue3[r] != "Empty") {
                System.out.print(queue3[r] + " ");
            }
        }
        System.out.println("\n");
        Repeat();

    }
    public static void ViewRemaining()
    {
        //display the remaining fuel stock
        remain=fuel-(tot+tot2+tot3);
        if(remain==500) {
            System.out.println("Remaining Fuel is  500L" );
            Repeat();
        }
        else {
            System.out.println("Remaining Fuel is " + remain+"L");
            Repeat();
        }
        tot=0;tot2=0;tot3=0;
    }
    public static void AddFuel()
    {
        //display total fuel stock after adding fuel
        System.out.print("How many Literes Do you add? : ");
        Scanner adding=new Scanner(System.in);
        int add1=adding.nextInt();
        fuel=remain+add1;
        System.out.println("Total fuel stock: "+fuel);
        Repeat();
    }

    public static void StoreDta()
    {//store the data to a text file
        try {
            BufferedWriter writer= new BufferedWriter(new
FileWriter("filename.txt")); //Create a file
            writer.write("---Customers Details---");
```

```
            writer.write("\nQueue 1:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue1[c]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue1[c]);
                }
            }
            writer.write("\nQueue 2:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue2[c]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue2[c]);
                }
            }writer.write("\nQueue 3:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue3[c]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue3[c]);
                }

            }
            writer.close();
            Repeat();
        }catch (IOException e){  //handling errors
            e.printStackTrace();
            Repeat();
        }


    }
    public static void LoadData()
    {//load data from a text file
        try {
            BufferedReader reader= new BufferedReader(new
FileReader("filename.txt"));
            String line;
            while ((line=reader.readLine()) != null){  //Read the
file line by line
                System.out.println(line);
            }
            reader.close();
            Repeat();
        }catch (IOException e){       //handling errors
            e.printStackTrace();
            Repeat();
        }

    }



    public static void start()
    {

        Menu();
        System.out.print("Enter your choice: ");
        Scanner choice=new Scanner(System.in);
        String name1=choice.next();
        String name=name1.toUpperCase();
        switch (name)
        {
```

```
            case "100":
            case "VFQ":
                ViewFuelQueue();
            case "101":
            case "VEQ":
                ViewEmptyQueue();
            case "102":
            case "ACQ":
                add();
            case "103":
            case "RCQ":
                RemoveCustomer();
            case "104":
            case "PCQ":
                RemoveServed();
            case "105":
            case "VCS":
                ViewCustomerAlphabeticOrder();
            case "106":
            case "SPD":
                StoreDta();
            case "107":
            case "LPD":
                LoadData();
            case "108":
            case "STK":
                ViewRemaining();
            case "109":
            case "AFS":
                AddFuel();
            case "999":
            case "EXT":
                exit(0);

        }
    }

    public static void main(String[] args) {
        for(int h=0;h<6;h++)
        {
            queue1[h]="Empty";
            queue2[h]="Empty";
            queue3[h]="Empty";
        }

        start();

    }
}
```

Task 02 – Source Code

Passenger.java

```
package com.example.w1898939_task2_task3_task4;

import java.util.Scanner;

public class Passenger extends FuelQueue{
   private String Cname2,Cname,Cname3,Cname4;
    public void setDetails(String Fname,String Sname,String
Vno,String litres){
        this.Cname=Fname;
        this.Cname2=Sname;
        this.Cname3=Vno;
        this.Cname4=litres;
    }
    public String getCname()
    {
        return Cname;

    }
    public String getCname2(){
        return Cname2;
    }
    public String getCname3()
    {
        return Cname3;
    }
    public String getCname4(){
        return Cname4;
    }
    public void adding(){

        System.out.print("Enter First Name: ");
        Scanner name1=new Scanner(System.in);
        Cname=name1.next();
        System.out.print("Enter Second Name: ");
        Scanner name2=new Scanner(System.in);
        Cname2=name2.next();
        System.out.print("Enter Vehicle Number: ");
        Scanner name3=new Scanner(System.in);
        Cname3=name3.next();
        System.out.print("Enter No. Of Liters required: ");
        Scanner name4=new Scanner(System.in);
        Cname4=name4.next();
    }

    public static void main(String[] args) {
    Passenger NewObj=new Passenger();
    NewObj.setDetails(NewObj.Cname, NewObj.Cname2, NewObj.Cname3,
NewObj.Cname4);









    }
}
```

FuelQueue.java

```
package com.example.w1898939_task2_task3_task4;

import java.io.*;
import java.util.Scanner;

import static java.lang.System.exit;

public class FuelQueue {
    String [][] queue1 = new String[6][4];
    String [][] queue2 = new String[6][4];
    String [][] queue3 = new String[6][4];
    String [][] queue4=new String[6][4];
    String [][] queue5=new String[6][4];

    static int x1;
    static int x2=10;
    static int x3=10;
    int x4=10;
    int x5=10;
    static int one=0;
    static int two=0;
    static int three=0;
    int four=0;
    int five=0;
    static int tot=0;
    static int tot2=0;
    static int tot3=0;
    int tot4=0;
    int tot5=0;
    static int x,y,z,e,size,size1,size2,size3,size4,p,u,m;
    static int c=0;
    static int t=0;
    static int v=0;
    static int w=0;
    static int s=0;
    static int d=0;
    static String temp,temp1,temp2,j;
    static int fuel=6600;
    static int remain=0;
   int length1=0;
   int length2=0;
   int length3=0;
   int length4=0;
   int length5=0;

    public  void Menu(){
        //Display the Menu
        System.out.println("\n");
        System.out.println("------------------Fuel Management System-
--------------\n");
        System.out.println("\t100 or VFQ: View all Fuel Queues");
        System.out.println("\t101 or VEQ: View all Empty Queues");
        System.out.println("\t102 or ACQ: Add customer to a Queue");
        System.out.println("\t103 or RCQ: Remove a customer from a
Queue");
        System.out.println("\t104 or PCQ: Remove a served customer");
        System.out.println("\t105 or VCS: View Customers Sorted in
alphabetical order");
        System.out.println("\t106 or SPD: Store Program Data into
```

```
file");
        System.out.println("\t107 or LPD: Load Program Data from
file");
        System.out.println("\t108 or STK: View Remaining Fuel
Stock");
        System.out.println("\t109 or AFS: Add Fuel Stock");
        System.out.println("\t110 or IFQ: Print Income Of Each Fuel
Queue");
        System.out.println("\t999 or EXT: Exit the Program\n");

    }
    public void start()
    {
        Menu();
        System.out.print("Enter your choice: ");
        Scanner choice=new Scanner(System.in);
        String name1=choice.next();
        String name=name1.toUpperCase();
        switch (name)
        {
            case "100":
            case "VFQ":
                ViewFuelQueue();
            case "101":
            case "VEQ":
                ViewEmptyQueue();
            case "102":
            case "ACQ":
                add();
            case "103":
            case "RCQ":
                RemoveCustomer();
            case "104":
            case "PCQ":
                RemoveServed();
            case "105":
            case "VCS":
                ViewCustomerAlphabeticOrder();
            case "106":
            case "SPD":
               StoreDta();
            case "107":
            case "LPD":
                LoadData();
            case "108":
            case "STK":
               ViewRemaining();
            case "109":
            case "AFS":
                AddFuel();
            case "110":
            case "IFQ":
                PrintIncome();
            case "999":
            case "EXT":
                exit(0);

        }
    }
```

```
    public void first()
    {
        for(int h=0;h<6;h++)
        {
            for(int y=0;y<4;y++){
            queue1[h][y]="Empty";
            queue2[h][y]="Empty";
            queue3[h][y]="Empty";
            queue4[h][y]="Empty";
            queue5[h][y]="Empty";
        }}
    }
    public void Repeat() {
        System.out.print("Enter 1=Menu  0=Exit: ");
        Scanner no = new Scanner(System.in);
        String view = no.next();
        switch (view) {
            case "1":
                start();
            case "0":
                exit(0);

        }
    }



    public void add() {//add customer to the queues

        System.out.print("Do you want to add a customer(Y/N):");
        Scanner customer = new Scanner(System.in);
        String answer1 = customer.next();
        String answer = answer1.toUpperCase();
        switch (answer) {
            case "Y":

                if ((length1 < length2) && (length1 <
length3)&&(length1<length4)&&(length1<length5)) {
                    while (one <= 5) {
                        Passenger obj3 = new Passenger();
                        obj3.adding();

                        queue1[one][0] = obj3.getCname();
                        queue1[one][1] = obj3.getCname2();
                        queue1[one][2] = obj3.getCname3();
                        queue1[one][3] = obj3.getCname4();

                        tot = tot+Integer.parseInt(queue1[one][3]);
                        System.out.println(obj3.getCname() + " is
successfully added to Queue1" + " " + "\t\t\tTotal liters= " + tot);
                        if (one == 5) {
                            System.out.println("Queue 1 is full");

                        } else {
                            one = one + 1;
                            length1=length1+1;
                            add();
                        }
                    }
```

```
                }



                else if ((length2 < length1) && (length2 <
length3)&&(length2<length4)&&(length2<length5)) {
                    while (two <= 5) {
                        Passenger obj4 = new Passenger();
                        obj4.adding();
                        queue2[two][0] = obj4.getCname();
                        queue2[two][1] = obj4.getCname2();
                        queue2[two][2] = obj4.getCname3();
                        queue2[two][3] = obj4.getCname4();
                        tot2 =tot2+ Integer.parseInt(queue2[two][3]);
                        System.out.println(obj4.getCname() + " is
successfully added to Queue2" + " " + "\t\t\tTotal Litres= " + tot2);
                        if (two == 5) {
                            System.out.println("Queue 2 is full");


                        } else {
                            two = two + 1;
                            length2=length2+1;
                            add();
                        }
                    }
                }



                else if ((length3 < length1) && (length3 <
length2)&&(length3<length4)&&(length3<length5)) {
                    while (three <= 5) {
                        Passenger obj5 = new Passenger();
                        obj5.adding();
                        queue3[three][0] = obj5.getCname();
                        queue3[three][1] = obj5.getCname2();
                        queue3[three][2] = obj5.getCname3();
                        queue3[three][3] = obj5.getCname4();
                        tot3 = tot3
+Integer.parseInt(queue3[three][3]) ;
                        System.out.println(obj5.getCname() + " is
successfully added to Queue3" + " " + "\t\t\tTotal Liters= " + tot3);
                        if (three == 5) {
                            System.out.println("Queue 3 is full");

                        } else {
                            three = three + 1;
                            length3 = length3 + 1;
                            add();
                        }
                    }}

                else if ((length4 < length1) && (length4 <
length2)&&(length4<length3)&&(length4<length5)) {
                    while (four <= 5) {
                        Passenger obj6 = new Passenger();
                        obj6.adding();
                        queue4[four][0] = obj6.getCname();
                        queue4[four][1] = obj6.getCname2();
                        queue4[four][2] = obj6.getCname3();
```

```
                        queue4[four][3] = obj6.getCname4();
                        tot4 = tot4
+Integer.parseInt(queue4[four][3]);
                        System.out.println(obj6.getCname() + " is
successfully added to Queue4" + " " + "\t\t\tTotal Liters= " + tot4);
                        if (four == 5) {
                            System.out.println("Queue 4 is full");

                        } else {
                            four = four + 1;
                            length4=length4+1;
                            add();
                        }
                    }
                }

                else if ((length5 < length1) && (length5 <
length2)&&(length5<length3)&&(length5<length4)) {
                    while (five <= 5) {
                        Passenger obj6 = new Passenger();
                        obj6.adding();
                        queue5[five][0] = obj6.getCname();
                        queue5[five][1] = obj6.getCname2();
                        queue5[five][2] = obj6.getCname3();
                        queue5[five][3] = obj6.getCname4();
                        tot5 = tot5
+Integer.parseInt(queue5[five][3]) ;
                        System.out.println(obj6.getCname() + " is
successfully added to Queue5" + " " + "\t\t\tTotal Liters= " + tot5);
                        if (five == 5) {
                            System.out.println("Queue 5 is full");

                        } else {
                            five = five + 1;
                            length5=length5+1;
                            add();
                        }
                    }
                }

                else if ((length1 == length2) && (length1 ==
length3)&&(length1==length4)&&(length1==length5)) {

                    while (one <= 5) {
                        Passenger obj3 = new Passenger();
                        obj3.adding();

                        queue1[one][0] = obj3.getCname();
                        queue1[one][1] = obj3.getCname2();
                        queue1[one][2] = obj3.getCname3();
                        queue1[one][3] = obj3.getCname4();

                        tot = tot+Integer.parseInt(queue1[one][3]);
                        System.out.println( queue1[one][0]+ " is
successfully added to Queue1" + " " + "\t\t\tTotal liters= " + tot);
                        if (one == 5) {
                            System.out.println("Queue 1 is full");
```

```
                        } else {
                            one = one + 1;
                            length1=length1+1;
                            add();
                        }
                    }
                } else if ((length2 == length3)&&(length2==length4))
{

                    while (two <= 5) {
                        Passenger obj4 = new Passenger();
                        obj4.adding();
                        queue2[two][0] = obj4.getCname();
                        queue2[two][1] = obj4.getCname2();
                        queue2[two][2] = obj4.getCname3();
                        queue2[two][3] = obj4.getCname4();
                        tot2 = tot2 +Integer.parseInt(queue2[two][3])
;
                        System.out.println(obj4.getCname() + " is
successfully added to Queue2" + " " + "\t\t\tTotal Litres= " + tot2);
                        if (two == 5) {
                            System.out.println("Queue 2 is full");


                        } else {
                            two = two + 1;
                            length2=length2+1;
                            add();
                        }
                    }
                } else if (length3==length4) {
                    while (three <= 5) {
                        Passenger obj5 = new Passenger();
                        obj5.adding();
                        queue3[three][0] = obj5.getCname();
                        queue3[three][1] = obj5.getCname2();
                        queue3[three][2] = obj5.getCname3();
                        queue3[three][3] = obj5.getCname4();
                        tot3 = tot3 +
Integer.parseInt(queue3[three][3]);
                        System.out.println(obj5.getCname() + " is
successfully added to Queue3" + " " + "\t\t\tTotal Liters= " + tot3);
                        if (three == 5) {
                            System.out.println("Queue 3 is full");

                        } else {
                            three = three + 1;
                            length3=length3+1;
                            add();
                        }
                }}
                else if (length4==length5) {
                    while (four <= 5) {
                        Passenger obj6 = new Passenger();
                        obj6.adding();
                        queue4[four][0] = obj6.getCname();
                        queue4[four][1] = obj6.getCname2();
                        queue4[four][2] = obj6.getCname3();
                        queue4[four][3] = obj6.getCname4();
                        tot4 = tot4 +
```

```
Integer.parseInt(queue4[four][3]);
                        System.out.println(obj6.getCname() + " is
successfully added to Queue4" + " " + "\t\t\tTotal Liters= " + tot4);
                        if (four == 5) {
                            System.out.println("Queue 4 is full");

                        } else {
                            four = four + 1;
                            length4=length4+1;
                            add();
                        }
                    }}



            case "N":
                Repeat();

        }

     return;
    }




    public void ViewFuelQueue()
    {

        //View all fuel queues details
        x=one;
        System.out.println("Queue 1:");
        for (x = 0; x <6; x++) {
                    if(queue1[x][0]!="Empty") {
                        System.out.println("Customer id:" + x + " " +
"Customer Name:" + queue1[x][0]);
                    }
        }
        System.out.println("Total liters= " + tot);

        y=two;
        System.out.println("Queue 2:");
        for (y = 0; y < 6; y++) {
            if(queue2[y][0]!="Empty") {
                System.out.println("Customer id:" + y + " " +
"Customer Name:" + queue2[y][0]);
            }
        }
        System.out.println("Total liters= " + tot2);

        z=three;
        System.out.println("Queue 3:");
        for (z = 0; z < 6; z++) {
            if(queue3[z][0]!="Empty") {
                System.out.println("Customer id:" + z + " " +
"Customer Name:" + queue3[z][0]);
            }
        }
        System.out.println("Total liters= " + tot3);
```

```
        u=four;
        System.out.println("Queue 4:");
        for (u = 0; u < 6; u++) {
            if(queue4[u][0]!="Empty") {
                System.out.println("Customer id:" + u + " " +
"Customer Name:" + queue4[u][0]);
            }
        }
        System.out.println("Total liters= " + tot4);

        m=five;
        System.out.println("Queue 5:");
        for (m = 0; m < 6; m++) {
            if(queue5[m][0]!="Empty") {
                System.out.println("Customer id:" + m + " " +
"Customer Name:" + queue5[m][0]);
            }
        }
        System.out.println("Total liters= " + tot5);

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

    public  void ViewEmptyQueue()
    {//Display empty queues

        if(tot==0)
        {
            System.out.println("Queue1 is empty");
        }
        if(tot2==0)
        {
            System.out.println("Queue2 is empty");
        }
        if(tot3==0)
        {
            System.out.println("Queue3 is empty");
        }
        if(tot4==0){
            System.out.println("Queue4 is empty");
        }
        if(tot5==0){
            System.out.println("Queue5 is empty");
        }
        Repeat();
    }

    public void RemoveCustomer()
    {// remove customers from the queues
        //E,X1,X2,X3
        System.out.print("Enter the Queue no:");
        Scanner que=new Scanner(System.in);
        String QueName=que.next();
        System.out.print("Enter the ID of Customer:");
        Scanner id=new Scanner(System.in);
        String idCustomer=id.next();
        e=Integer.parseInt(idCustomer);
        switch (QueName)
        {
            case "1":
```

```
                    try {
                        x1 = Integer.parseInt(queue1[e][3]);
                        tot = tot - x1;
                        System.out.println("Queue 01:");
                        System.out.println("CustomerID:" + e + "
CustomerName: " + queue1[e][0] + " is successfully removed");
                        for (int r = 0; r < 4; r++) {
                            queue1[e][r] = "Empty";
                        }
                        for(int y=e;y<4;y++){
                        if(queue1[e+1][0]!="Empty"){

                            queue1[e][0]=queue1[e+1][0];
                            queue1[e][1]=queue1[e+1][1];
                            queue1[e][2]=queue1[e+1][2];
                            queue1[e][3]=queue1[e+1][3];

                            for (int r = 0; r < 4; r++) {
                                queue1[e+1][r] = "Empty";
                            }

                        }}
                        System.out.println("\n");
                        Repeat();
                    }catch (Exception e){
                        System.out.println(e);
                        Repeat();
                    }

            case "2":
                try {
                    tot2 = tot2 - Integer.parseInt(queue2[e][3]);
                    System.out.println("Queue 02:");
                    System.out.println("CustomerID:" + e + "
CustomerName: " + queue2[e][0] + " is successfully removed");
                    for (int r = 0; r < 4; r++) {
                        queue2[e][r] = "Empty";
                    }
                    for(int y=e;y<4;y++){
                        if(queue2[e+1][0]!="Empty"){

                            queue2[e][0]=queue2[e+1][0];
                            queue2[e][1]=queue2[e+1][1];
                            queue2[e][2]=queue2[e+1][2];
                            queue2[e][3]=queue2[e+1][3];

                            for (int r = 0; r < 4; r++) {
                                queue2[e+1][r] = "Empty";
                            }

                        }}

                    System.out.println("\n");
                    Repeat();
                }catch (Exception e){
                    System.out.println(e);
                    Repeat();
                }
            case "3":
                try {
                    tot3 = tot3 - Integer.parseInt(queue3[e][3]);
```

```
                    System.out.println("Queue 03:");
                    System.out.println("CustomerID:" + e + "
CustomeName: " + queue3[e][0] + " is successfully removed");
                    for (int r = 0; r < 4; r++) {
                        queue3[e][r] = "Empty";
                    }
                    for(int y=e;y<4;y++){
                        if(queue3[e+1][0]!="Empty"){

                            queue3[e][0]=queue3[e+1][0];
                            queue3[e][1]=queue3[e+1][1];
                            queue3[e][2]=queue3[e+1][2];
                            queue3[e][3]=queue3[e+1][3];

                            for (int r = 0; r < 4; r++) {
                                queue3[e+1][r] = "Empty";
                            }

                        }}
                    System.out.println("\n");
                    Repeat();
                }catch (Exception e){
                    System.out.println(e);
                    Repeat();
                }

            case "4":
                try {
                    tot4 = tot4 - Integer.parseInt(queue4[e][3]);
                    System.out.println("Queue 04:");
                    System.out.println("CustomerID:" + e + "
CustomeName: " + queue4[e][0] + " is successfully removed");
                    for (int r = 0; r < 4; r++) {
                        queue4[e][r] = "Empty";
                    }
                    for(int y=e;y<4;y++){
                        if(queue4[e+1][0]!="Empty"){

                            queue4[e][0]=queue1[e+1][0];
                            queue4[e][1]=queue1[e+1][1];
                            queue4[e][2]=queue1[e+1][2];
                            queue4[e][3]=queue1[e+1][3];

                            for (int r = 0; r < 4; r++) {
                                queue4[e+1][r] = "Empty";
                            }

                        }}
                    System.out.println("\n");
                    Repeat();
                }catch (Exception e){
                    System.out.println(e);
                    Repeat();
                }
            case "5":
                try {
                    tot5 = tot5 - Integer.parseInt(queue5[e][3]);
                    System.out.println("Queue 05:");
                    System.out.println("CustomerID:" + e + "
CustomeName: " + queue5[e][0] + " is successfully removed");
                    for (int r = 0; r < 4; r++) {
```

```
                        queue5[e][r] = "Empty";
                    }
                    for(int y=e;y<4;y++){
                        if(queue1[e+1][0]!="Empty"){

                            queue5[e][0]=queue1[e+1][0];
                            queue5[e][1]=queue1[e+1][1];
                            queue5[e][2]=queue1[e+1][2];
                            queue5[e][3]=queue1[e+1][3];

                            for (int r = 0; r < 4; r++) {
                                queue5[e+1][r] = "Empty";
                            }

                        }}
                    System.out.println("\n");
                    Repeat();
                }catch (Exception e){
                    System.out.println(e);
                    Repeat();
                }


        }

    }

    public void RemoveServed()
    {
        //remove the first customer of the queue
        System.out.print("Enter the Queue no:");
        Scanner queNo=new Scanner(System.in);
        String QName=queNo.next();
        switch (QName)
        {
            case "1":

                    System.out.println("Queue1-CustomerId: 0"  + "
CustomerName: " + queue1[0][0] + " is successfully removed");
                    for (int r = 0; r < 4; r++) {
                        queue1[0][r] = "Empty";
                    }
                    for(int e=0;e<4;e++){
                    if(queue1[e+1][0]!="Empty"){

                        queue1[e][0]=queue1[e+1][0];
                        queue1[e][1]=queue1[e+1][1];
                        queue1[e][2]=queue1[e+1][2];
                        queue1[e][3]=queue1[e+1][3];

                        for (int r = 0; r < 4; r++) {
                            queue1[e+1][r] = "Empty";
                        }

                    }}

                    Repeat();

            case "2":

                System.out.println("Queue2-CustomerId: 0"+"
```

```
CustomerName: "+queue2[0][0]+" is successfully removed");
                for (int r=0;r<4;r++) {
                    queue2[0][r]= "Empty";
                }
                for(int e=0;e<4;e++){
                    if(queue2[e+1][0]!="Empty"){

                        queue2[e][0]=queue2[e+1][0];
                        queue2[e][1]=queue2[e+1][1];
                        queue2[e][2]=queue2[e+1][2];
                        queue2[e][3]=queue2[e+1][3];

                        for (int r = 0; r < 4; r++) {
                            queue2[e+1][r] = "Empty";
                        }

                    }}

                Repeat();
            case "3":

                System.out.println("Queue3-CustomerId:0 "+"
CustomerName: "+queue3[0][0]+" is successfully removed");
                for (int r=0;r<4;r++) {
                    queue3[v][r] = "Empty";
                }
                for(int e=0;e<4;e++){
                    if(queue3[e+1][0]!="Empty"){

                        queue3[e][0]=queue3[e+1][0];
                        queue3[e][1]=queue3[e+1][1];
                        queue3[e][2]=queue3[e+1][2];
                        queue3[e][3]=queue3[e+1][3];

                        for (int r = 0; r < 4; r++) {
                            queue3[e+1][r] = "Empty";
                        }

                    }}
                Repeat();
            case "4":

                System.out.println("Queue4-CustomerId:0 "+"
CustomerName: "+queue4[0][0]+" is successfully removed");
                for (int r=0;r<4;r++) {
                    queue4[0][r] = "Empty";
                }
                for(int e=0;e<4;e++){
                    if(queue4[e+1][0]!="Empty"){

                        queue4[e][0]=queue4[e+1][0];
                        queue4[e][1]=queue4[e+1][1];
                        queue4[e][2]=queue4[e+1][2];
                        queue4[e][3]=queue4[e+1][3];

                        for (int r = 0; r < 4; r++) {
                            queue4[e+1][r] = "Empty";
                        }

                    }}
                Repeat();
```

```
            case "5":

                System.out.println("Queue4-CustomerId:0 "+"
CustomerName: "+queue5[0][0]+" is successfully removed");
                for (int r=0;r<4;r++) {
                    queue5[0][r] = "Empty";
                }
                for(int e=0;e<4;e++){
                    if(queue5[e+1][0]!="Empty"){

                        queue5[e][0]=queue5[e+1][0];
                        queue5[e][1]=queue5[e+1][1];
                        queue5[e][2]=queue5[e+1][2];
                        queue5[e][3]=queue5[e+1][3];

                        for (int r = 0; r < 4; r++) {
                            queue5[e+1][r] = "Empty";
                        }

                    }}
                Repeat();


        }


    }

    public void ViewCustomerAlphabeticOrder()
    {//Display the customers name in alphabetical order
//one
        size = queue1.length;

        for (int a = 0; a < size - 1; a++) {

            for (int j = a + 1; j < queue1.length; j++) {
                if (queue1[a][0].compareTo(queue1[j][0]) > 0) {
                    temp = queue1[a][0];
                    queue1[a][0]= queue1[j][0];
                    queue1[j][0] = temp;
                }
            }

        }
        System.out.println("Queue 1:");
        for (int r=0;r<6;r++) {
            if (queue1[r][0] != "Empty") {

                    System.out.print(queue1[r][0]+ " ");

            }
        }
        //two

        size1 = queue2.length;

        for (int a = 0; a < size1 - 1; a++) {

            for (int j = a + 1; j < queue2.length; j++) {
                if (queue2[a][0].compareTo(queue2[j][0]) > 0) {
```

```
                temp1 = queue2[a][0];
                queue2[a][0] = queue2[j][0];
                queue2[j][0] = temp1;
            }
        }

    }
    System.out.println("\nQueue 2:");

    for (int r=0;r<6;r++) {
        if(queue2[r][0]!="Empty") {
            System.out.print(queue2[r][0]+" ");
        }
    }
    //three
    size2 = queue3.length;

    for (int a = 0; a < size2 - 1; a++) {

        for (int j = a + 1; j < queue3.length; j++) {
            if (queue3[a][0].compareTo(queue3[j][0]) > 0) {
                temp2 = queue3[a][0];
                queue3[a][0] = queue3[j][0];
                queue3[j][0] = temp2;
            }
        }

    }

    System.out.println("\nQueue 3:");

    for(int r=0;r<6;r++) {
        if (queue3[r][0]!= "Empty") {
            System.out.print(queue3[r][0] + " ");
        }
    }
    //four
    size3 = queue4.length;

    for (int a = 0; a < size2 - 1; a++) {

        for (int j = a + 1; j < queue4.length; j++) {
            if (queue4[a][0].compareTo(queue4[j][0]) > 0) {
                temp2 = queue4[a][0];
                queue4[a][0] = queue4[j][0];
                queue4[j][0] = temp2;
            }
        }

    }

    System.out.println("\nQueue 4:");

    for(int r=0;r<6;r++) {
        if (queue4[r][0]!= "Empty") {
            System.out.print(queue4[r][0] + " ");
        }
    }
    //five
    size4 = queue5.length;
```

```
        for (int a = 0; a < size2 - 1; a++) {

            for (int j = a + 1; j < queue5.length; j++) {
                if (queue5[a][0].compareTo(queue5[j][0]) > 0) {
                    temp2 = queue5[a][0];
                    queue5[a][0] = queue5[j][0];
                    queue5[j][0] = temp2;
                }
            }

        }

        System.out.println("\nQueue 5:");

        for(int r=0;r<6;r++) {
            if (queue5[r][0]!= "Empty") {
                System.out.print(queue5[r][0] + " ");
            }
        }

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

    }
    public void ViewRemaining()
    {
        //display the remaining fuel stock
        remain=fuel-(tot+tot2+tot3+tot4+tot5);
        if(remain==500) {
            System.out.println("Remaining Fuel is  500L" );
            Repeat();
        }
        else {
            System.out.println("Remaining Fuel is " + remain+"L");
            Repeat();
        }
        tot=0;tot2=0;tot3=0;tot4=0;tot5=0;
    }

    public void AddFuel()
    {
        //display total fuel stock after adding fuel
        System.out.print("How many Literes Do you add? : ");
        Scanner adding=new Scanner(System.in);
        int add1=adding.nextInt();
        fuel=remain+add1;
        System.out.println("Total fuel stock: "+fuel);
        Repeat();
    }

    public void PrintIncome(){
        System.out.println("Queue1 income is Rs."+(430*tot));
        System.out.println("Queue2 income is Rs."+(430*tot2));
        System.out.println("Queue3 income is Rs."+(430*tot3));
        System.out.println("Queue4 income is Rs."+(430*tot4));
        System.out.println("Queue5 income is Rs."+(430*tot5));
        Repeat();
    }

    public void StoreDta()
    {//store the data to a text file
```

```
        try {
            BufferedWriter writer= new BufferedWriter(new
FileWriter("filename.txt")); //Create a file
            writer.write("---Customers Details---");
            writer.write("\nQueue 1:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue1[c][0]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue1[c][0]);
                }
            }
            writer.write("\nQueue 2:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue2[c][0]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue2[c][0]);
                }
            }
            writer.write("\nQueue 3:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue3[c][0]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue3[c][0]);
                }
            }
            writer.write("\nQueue 4:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue4[c][0]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue4[c][0]);
                }
            }
            writer.write("\nQueue 5:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue5[c][0]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue5[c][0]);
                }
            }
            writer.close();
            Repeat();
        }catch (IOException e){  //handling errors
            e.printStackTrace();
            Repeat();
        }
    }

    public void LoadData()
    {//load data from a text file
        try {
            BufferedReader reader= new BufferedReader(new
FileReader("filename.txt"));
            String line;
            while ((line=reader.readLine()) != null){  //Read the
file line by line
                System.out.println(line);
            }
            reader.close();
            Repeat();
        }catch (IOException e){       //handling errors
            e.printStackTrace();
```

```
                Repeat();
            }

    }

    public static void main(String[] args) {


    }


}
```

Driver.java

```
package com.example.w1898939_task2_task3_task4;

public class Driver extends Passenger{

    public static void main(String[] args) {
        Driver obj=new Driver();
        obj.first();
        obj.start();
    }
}
```

Task 03 – Source Code

Task3_Passenger.java

```
package com.example.w1898939_task2_task3_task4;

import java.util.Scanner;

public class Task3_Passenger extends Task3_FuelQueue{
    private String Cname2,Cname,Cname3,Cname4;
    public void setDetails(String Fname,String Sname,String
Vno,String litres){
        this.Cname=Fname;
        this.Cname2=Sname;
        this.Cname3=Vno;
        this.Cname4=litres;
    }
    public String getCname()
    {
        return Cname;

    }
    public String getCname2(){
        return Cname2;
    }
    public String getCname3()
    {
        return Cname3;
    }
    public String getCname4(){
        return Cname4;
    }
    public void adding(){

        System.out.print("Enter First Name: ");
        Scanner name1=new Scanner(System.in);
        Cname=name1.next();
        System.out.print("Enter Second Name: ");
        Scanner name2=new Scanner(System.in);
        Cname2=name2.next();
        System.out.print("Enter Vehicle Number: ");
        Scanner name3=new Scanner(System.in);
        Cname3=name3.next();
        System.out.print("Enter No. Of Liters required: ");
        Scanner name4=new Scanner(System.in);
        Cname4=name4.next();
    }

    public static void main(String[] args) {
        Passenger NewObj1=new Passenger();




    }

}
```

Task3_FuelQueue.java

```
package com.example.w1898939_task2_task3_task4;

import java.io.*;
import java.util.Scanner;

import static java.lang.System.exit;

public class Task3_FuelQueue {
    String[][] queue1 = new String[6][4];
    String[][] queue5 = new String[6][4];
    String[][] WaitQueue1 = new String[10][4];
    String[][] WaitQueue2 = new String[10][4];
    String[][] WaitQueue3 = new String[10][4];
    String[][] WaitQueue4 = new String[10][4];
    String[][] WaitQueue5 = new String[100][4];

    static int x1;
    String[][] queue2 = new String[6][4];
    String[][] queue3 = new String[6][4];
    String[][] queue4 = new String[6][4];
    static int x2 = 10;
    static int x3 = 10;
    int x4 = 10;
    int x5 = 10;
    static int one = 0;
    static int two = 0;
    static int three = 0;
    int four = 0;
    int five = 0;
    static int tot = 0;
    static int tot2 = 0;
    static int tot3 = 0;
    int tot4 = 0;
    int tot5 = 0;
    static int x, y, z, e, size, size1, size2, size3, size4, p, u, m;
    static int c = 0;
    static int t = 0;
    static int v = 0;
    static int w = 0;
    static int s = 0;
    static int d = 0;
    static String temp, temp1, temp2, j;
    static int fuel = 6600;
    static int remain = 0;
    int length1 = 0;
    int length2 = 0;
    int length3 = 0;
    int length4 = 0;
    int length5 = 0;
    int i = 0;
    int front = 0;
    int front2 = 0;
    int front3 = 0;
    int front4 = 0;
    int front5 = 0;

    public void Menu() {
        //Display the Menu
        System.out.println("\n");
        System.out.println("------------------Fuel Management System-
```

```
--------------\n");
        System.out.println("\t100 or VFQ: View all Fuel Queues");
        System.out.println("\t101 or VEQ: View all Empty Queues");
        System.out.println("\t102 or ACQ: Add customer to a Queue");
        System.out.println("\t103 or RCQ: Remove a customer from a
Queue");
        System.out.println("\t104 or PCQ: Remove a served customer");
        System.out.println("\t105 or VCS: View Customers Sorted in
alphabetical order");
        System.out.println("\t106 or SPD: Store Program Data into
file");
        System.out.println("\t107 or LPD: Load Program Data from
file");
        System.out.println("\t108 or STK: View Remaining Fuel
Stock");
        System.out.println("\t109 or AFS: Add Fuel Stock");
        System.out.println("\t110 or IFQ: Print Income Of Each Fuel
Queue");
        System.out.println("\t999 or EXT: Exit the Program\n");

    }

    public void start() {
        Menu();
        System.out.print("Enter your choice: ");
        Scanner choice = new Scanner(System.in);
        String name1 = choice.next();
        String name = name1.toUpperCase();
        switch (name) {
            case "100":
            case "VFQ":
                ViewFuelQueue();
            case "101":
            case "VEQ":
                ViewEmptyQueue();
            case "102":
            case "ACQ":
                add();
            case "103":
            case "RCQ":
                RemoveCustomer();
            case "104":
            case "PCQ":
                RemoveServed();
            case "105":
            case "VCS":
                ViewCustomerAlphabeticOrder();
            case "106":
            case "SPD":
                StoreDta();
            case "107":
            case "LPD":
                LoadData();
            case "108":
            case "STK":
                ViewRemaining();
            case "109":
            case "AFS":
                AddFuel();
            case "110":
            case "IFQ":
```

```
                PrintIncome();
            case "999":
            case "EXT":
                exit(0);

        }
    }


    public void first() {
        for (int h = 0; h < 6; h++) {
            for (int y = 0; y < 4; y++) {
                queue1[h][y] = "Empty";
                queue2[h][y] = "Empty";
                queue3[h][y] = "Empty";
                queue4[h][y] = "Empty";
                queue5[h][y] = "Empty";
            }
        }
    }

    public void Repeat() {
        System.out.print("Enter 1=Menu  0=Exit: ");
        Scanner no = new Scanner(System.in);
        String view = no.next();
        switch (view) {
            case "1":
                start();
            case "0":
                exit(0);

        }
    }


    public void add() {//add customer to the queues

        System.out.print("Do you want to add a customer(Y/N):");
        Scanner customer = new Scanner(System.in);
        String answer1 = customer.next();
        String answer = answer1.toUpperCase();
        switch (answer) {
            case "Y":

                if ((length1 < length2) && (length1 < length3) &&
(length1 < length4) && (length1 < length5)) {
                    while (one <= 5) {
                        Passenger obj3 = new Passenger();
                        obj3.adding();

                        queue1[one][0] = obj3.getCname();
                        queue1[one][1] = obj3.getCname2();
                        queue1[one][2] = obj3.getCname3();
                        queue1[one][3] = obj3.getCname4();

                        tot = tot + Integer.parseInt(queue1[one][3]);
                        System.out.println(obj3.getCname() + " is
successfully added to Queue1" + " " + "\t\t\tTotal liters= " + tot);
                            one = one + 1;
                            length1=length1+1;
                            add();
```

```
                            if(one==5){
                                System.out.println("Queue1 is full.
Added to the waiting queue");
                                waiting();


                            }

                        }




                    }


                else if ((length2 < length1) && (length2 < length3)
&& (length2 < length4) && (length2 < length5)) {
                    while (two <= 5) {
                        Passenger obj4 = new Passenger();
                        obj4.adding();
                        queue2[two][0] = obj4.getCname();
                        queue2[two][1] = obj4.getCname2();
                        queue2[two][2] = obj4.getCname3();
                        queue2[two][3] = obj4.getCname4();
                        tot2 = tot2 +
Integer.parseInt(queue2[two][3]);
                        System.out.println(obj4.getCname() + " is
successfully added to Queue2" + " " + "\t\t\tTotal Litres= " + tot2);

                        if (two== 5) {
                            System.out.println("Queue 2 is full");
                            waiting2();


                        } else {
                            two = two + 1;
                            length2=length2+1;
                            add();
                        }


                    }


                } else if ((length3 < length1) && (length3 < length2)
&& (length3 < length4) && (length3 < length5)) {
                    while (three <= 5) {
                        Passenger obj5 = new Passenger();
                        obj5.adding();
                        queue3[three][0] = obj5.getCname();
                        queue3[three][1] = obj5.getCname2();
                        queue3[three][2] = obj5.getCname3();
                        queue3[three][3] = obj5.getCname4();
                        tot3 = tot3 +
Integer.parseInt(queue3[three][3]);
                        System.out.println(obj5.getCname() + " is
successfully added to Queue3" + " " + "\t\t\tTotal Liters= " + tot3);
```

```
                        if (three == 5) {
                            System.out.println("Queue 3 is full");
                            waiting3();


                        } else {
                            three = three + 1;
                            length3 = length3 + 1;
                            add();
                        }

                    }


                } else if ((length4 < length1) && (length4 < length2)
&& (length4 < length3) && (length4 < length5)) {
                    while (four <= 5) {
                        Passenger obj6 = new Passenger();
                        obj6.adding();
                        queue4[four][0] = obj6.getCname();
                        queue4[four][1] = obj6.getCname2();
                        queue4[four][2] = obj6.getCname3();
                        queue4[four][3] = obj6.getCname4();
                        tot4 = tot4 +
Integer.parseInt(queue4[four][3]);
                        System.out.println(obj6.getCname() + " is
successfully added to Queue4" + " " + "\t\t\tTotal Liters= " + tot4);


                        if (four==5) {
                            System.out.println("Queue 4 is full");
                            waiting4();




                        } else {
                            four = four + 1;
                            length4=length4+1;
                            add();
                        }


                    }


                } else if ((length5 < length1) && (length5 < length2)
&& (length5 < length3) && (length5 < length4)) {
                    while (five <= 5) {
                        Passenger obj78 = new Passenger();
                        obj78.adding();
                        queue5[five][0] = obj78.getCname();
                        queue5[five][1] = obj78.getCname2();
                        queue5[five][2] = obj78.getCname3();
                        queue5[five][3] = obj78.getCname4();
                        tot5 = tot5 +
Integer.parseInt(queue5[five][3]);
                        System.out.println(obj78.getCname() + " is
successfully added to Queue5" + " " + "\t\t\tTotal Liters= " + tot5);
```

```
                        if (five == 5) {
                            System.out.print("Queue 5 is full. ");
                            waiting5();




                        } else {
                            five = five + 1;
                            length5=length5+1;
                            add();
                        }

                    }



                } else if ((length1 == length2) && (length1 ==
length3) && (length1 == length4) && (length1 == length5)) {

                    while (one <= 5) {
                        Passenger obj3 = new Passenger();
                        obj3.adding();

                        queue1[one][0] = obj3.getCname();
                        queue1[one][1] = obj3.getCname2();
                        queue1[one][2] = obj3.getCname3();
                        queue1[one][3] = obj3.getCname4();

                        tot = tot + Integer.parseInt(queue1[one][3]);
                        System.out.println(queue1[one][0] + " is
successfully added to Queue1" + " " + "\t\t\tTotal liters= " + tot);

                        if (one==5) {
                            System.out.print("Queue 1 is full. Added
to the waiting Queue");
                            waiting();

                                }


                            else{
                                one = one + 1;
                                length1 = length1 + 1;
                                add();
                            }

                         }

                } else if ((length2 == length3) && (length2 ==
length4)) {

                    while (two <= 5) {
                        Passenger obj4 = new Passenger();
                        obj4.adding();
                        queue2[two][0] = obj4.getCname();
                        queue2[two][1] = obj4.getCname2();
                        queue2[two][2] = obj4.getCname3();
                        queue2[two][3] = obj4.getCname4();
```

```
                        tot2 = tot2 +
Integer.parseInt(queue2[two][3]);
                        System.out.println(obj4.getCname() + " is
successfully added to Queue2" + " " + "\t\t\tTotal Litres= " + tot2);

                        if (two== 5) {
                            System.out.print("Queue 2 is full. Added
to the waiting Queue");
                            waiting2();




                        } else {
                            two = two + 1;
                            length2=length2+1;
                            add();
                        }
                    }


                } else if (length3 == length4) {
                    while (three <= 5) {
                        Passenger obj5 = new Passenger();
                        obj5.adding();
                        queue3[three][0] = obj5.getCname();
                        queue3[three][1] = obj5.getCname2();
                        queue3[three][2] = obj5.getCname3();
                        queue3[three][3] = obj5.getCname4();
                        tot3 = tot3 +
Integer.parseInt(queue3[three][3]);
                        System.out.println(obj5.getCname() + " is
successfully added to Queue3" + " " + "\t\t\tTotal Liters= " + tot3);

                        if (three == 5) {
                            System.out.print("Queue 3 is full. Added
to the waiting Queue");
                            waiting3();




                        } else {
                            three = three + 1;
                            length3=length3+1;
                            add();
                        }


                    }

                }

                else if (length4 == length5) {
            while (four <= 5) {
                Passenger obj6 = new Passenger();
                obj6.adding();
                queue4[four][0] = obj6.getCname();
                queue4[four][1] = obj6.getCname2();
```

```
                queue4[four][2] = obj6.getCname3();
                queue4[four][3] = obj6.getCname4();
                tot4 = tot4 + Integer.parseInt(queue4[four][3]);
                System.out.println(obj6.getCname() + " is
successfully added to Queue4" + " " + "\t\t\tTotal Liters= " + tot4);
                if (four == 5) {
                    System.out.print("Queue 4 is full. Added to the
waiting Queue");
                    waiting4();




                } else {
                    four = four + 1;
                    length4=length4+1;
                    add();
                }

            }



        }


        case "N":
        Repeat();

    }}
    int l=0;
public void waiting()
{

    Passenger objnew=new Passenger();
    objnew.adding();
    WaitQueue1[l][0]= objnew.getCname();
    WaitQueue1[l][1]=objnew.getCname2();
    WaitQueue1[l][2]=objnew.getCname3();
    WaitQueue1[l][3]=objnew.getCname4();
    l=l+1;
    Repeat();

}
    public void waiting2()
    {

        Passenger objnew=new Passenger();
        objnew.adding();
        WaitQueue2[l][0]= objnew.getCname();
        WaitQueue2[l][1]=objnew.getCname2();
        WaitQueue2[l][2]=objnew.getCname3();
        WaitQueue2[l][3]=objnew.getCname4();
        l=l+1;
        Repeat();

    }
    public void waiting3()
    {
```

```
        Passenger objnew=new Passenger();
        objnew.adding();
        WaitQueue3[l][0]= objnew.getCname();
        WaitQueue3[l][1]=objnew.getCname2();
        WaitQueue3[l][2]=objnew.getCname3();
        WaitQueue3[l][3]=objnew.getCname4();
        l=l+1;
        Repeat();

    }
    public void waiting4()
    {

        Passenger objnew=new Passenger();
        objnew.adding();
        WaitQueue4[l][0]= objnew.getCname();
        WaitQueue4[l][1]=objnew.getCname2();
        WaitQueue4[l][2]=objnew.getCname3();
        WaitQueue4[l][3]=objnew.getCname4();
        l=l+1;
        Repeat();

    }
    public void waiting5()
    {

        Passenger objnew=new Passenger();
        objnew.adding();
        WaitQueue5[l][0]= objnew.getCname();
        WaitQueue5[l][1]=objnew.getCname2();
        WaitQueue5[l][2]=objnew.getCname3();
        WaitQueue5[l][3]=objnew.getCname4();
        l=l+1;
        Repeat();

    }



    public void ViewFuelQueue()
    {

        //View all fuel queues details
        x=one;
        System.out.println("Queue 1:");
        for (x = 0; x <6; x++) {
            if(queue1[x][0]!="Empty") {
                System.out.println("Customer id:" + x + " " +
"Customer Name:" + queue1[x][0]);
            }
        }
        System.out.println("Total liters= " + tot);

        y=two;
        System.out.println("Queue 2:");
        for (y = 0; y < 6; y++) {
            if(queue2[y][0]!="Empty") {
                System.out.println("Customer id:" + y + " " +
"Customer Name:" + queue2[y][0]);
            }
        }
```

```
        System.out.println("Total liters= " + tot2);

        z=three;
        System.out.println("Queue 3:");
        for (z = 0; z < 6; z++) {
            if(queue3[z][0]!="Empty") {
                System.out.println("Customer id:" + z + " " +
"Customer Name:" + queue3[z][0]);
            }
        }
        System.out.println("Total liters= " + tot3);

        u=four;
        System.out.println("Queue 4:");
        for (u = 0; u < 6; u++) {
            if(queue4[u][0]!="Empty") {
                System.out.println("Customer id:" + u + " " +
"Customer Name:" + queue4[u][0]);
            }
        }
        System.out.println("Total liters= " + tot4);

        m=five;
        System.out.println("Queue 5:");
        for (m = 0; m < 6; m++) {
            if(queue5[m][0]!="Empty") {
                System.out.println("Customer id:" + m + " " +
"Customer Name:" + queue5[m][0]);
            }
        }
        System.out.println("Total liters= " + tot5);

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

    public  void ViewEmptyQueue()
    {//Display empty queues

        if(tot==0)
        {
            System.out.println("Queue1 is empty");
        }
        if(tot2==0)
        {
            System.out.println("Queue2 is empty");
        }
        if(tot3==0)
        {
            System.out.println("Queue3 is empty");
        }
        if(tot4==0){
            System.out.println("Queue4 is empty");
        }
        if(tot5==0){
            System.out.println("Queue5 is empty");
        }
        Repeat();
    }

    public void RemoveCustomer()
```

```
    {// remove customers from the queues
        //E,X1,X2,X3
        System.out.print("Enter the Queue no:");
        Scanner que=new Scanner(System.in);
        String QueName=que.next();
        System.out.print("Enter the ID of Customer:");
        Scanner id=new Scanner(System.in);
        String idCustomer=id.next();
        e=Integer.parseInt(idCustomer);
        switch (QueName)
        {
            case "1":
                try {
                    x1 = Integer.parseInt(queue1[e][3]);
                    tot = tot - x1;
                    System.out.println("Queue 01:");
                    System.out.println("CustomerID:" + e + "
CustomerName: " + queue1[e][0] + " is successfully removed");
                    for (int r = 0; r < 4; r++) {
                        queue1[e][r] = "Empty";
                    }
                    for(int y=e;y<4;y++){
                        if(queue1[e+1][0]!="Empty"){

                            queue1[e][0]=queue1[e+1][0];
                            queue1[e][1]=queue1[e+1][1];
                            queue1[e][2]=queue1[e+1][2];
                            queue1[e][3]=queue1[e+1][3];

                            for (int r = 0; r < 4; r++) {
                                queue1[e+1][r] = "Empty";
                            }


                        }}
                    System.out.println("\n");
                    Repeat();
                }catch (Exception e){
                    System.out.println(e);
                    Repeat();
                }

            case "2":
                try {
                    tot2 = tot2 - Integer.parseInt(queue2[e][3]);
                    System.out.println("Queue 02:");
                    System.out.println("CustomerID:" + e + "
CustomerName: " + queue2[e][0] + " is successfully removed");
                    for (int r = 0; r < 4; r++) {
                        queue2[e][r] = "Empty";
                    }
                    for(int y=e;y<4;y++){
                        if(queue2[e+1][0]!="Empty"){

                            queue2[e][0]=queue2[e+1][0];
                            queue2[e][1]=queue2[e+1][1];
                            queue2[e][2]=queue2[e+1][2];
                            queue2[e][3]=queue2[e+1][3];

                            for (int r = 0; r < 4; r++) {
                                queue2[e+1][r] = "Empty";
```

```
                            }

                        }}

                        System.out.println("\n");
                        Repeat();
                    }catch (Exception e){
                        System.out.println(e);
                        Repeat();
                    }
                case "3":
                    try {
                        tot3 = tot3 - Integer.parseInt(queue3[e][3]);
                        System.out.println("Queue 03:");
                        System.out.println("CustomerID:" + e + "
CustomeName: " + queue3[e][0] + " is successfully removed");
                        for (int r = 0; r < 4; r++) {
                            queue3[e][r] = "Empty";
                        }
                        for(int y=e;y<4;y++){
                            if(queue3[e+1][0]!="Empty"){

                                queue3[e][0]=queue3[e+1][0];
                                queue3[e][1]=queue3[e+1][1];
                                queue3[e][2]=queue3[e+1][2];
                                queue3[e][3]=queue3[e+1][3];

                                for (int r = 0; r < 4; r++) {
                                    queue3[e+1][r] = "Empty";
                                }

                            }}
                        System.out.println("\n");
                        Repeat();
                    }catch (Exception e){
                        System.out.println(e);
                        Repeat();
                    }

                case "4":
                    try {
                        tot4 = tot4 - Integer.parseInt(queue4[e][3]);
                        System.out.println("Queue 04:");
                        System.out.println("CustomerID:" + e + "
CustomeName: " + queue4[e][0] + " is successfully removed");
                        for (int r = 0; r < 4; r++) {
                            queue4[e][r] = "Empty";
                        }
                        for(int y=e;y<4;y++){
                            if(queue4[e+1][0]!="Empty"){

                                queue4[e][0]=queue1[e+1][0];
                                queue4[e][1]=queue1[e+1][1];
                                queue4[e][2]=queue1[e+1][2];
                                queue4[e][3]=queue1[e+1][3];

                                for (int r = 0; r < 4; r++) {
                                    queue4[e+1][r] = "Empty";
                                }

                            }}
```

```
                    System.out.println("\n");
                    Repeat();
                }catch (Exception e){
                    System.out.println(e);
                    Repeat();
                }
            case "5":
                try {
                    tot5 = tot5 - Integer.parseInt(queue5[e][3]);
                    System.out.println("Queue 05:");
                    System.out.println("CustomerID:" + e + "
CustomeName: " + queue5[e][0] + " is successfully removed");
                    for (int r = 0; r < 4; r++) {
                        queue5[e][r] = "Empty";
                    }
                    for(int y=e;y<4;y++){
                        if(queue1[e+1][0]!="Empty"){

                            queue5[e][0]=queue1[e+1][0];
                            queue5[e][1]=queue1[e+1][1];
                            queue5[e][2]=queue1[e+1][2];
                            queue5[e][3]=queue1[e+1][3];

                            for (int r = 0; r < 4; r++) {
                                queue5[e+1][r] = "Empty";
                            }

                        }}
                    System.out.println("\n");
                    Repeat();
                }catch (Exception e){
                    System.out.println(e);
                    Repeat();
                }


        }

    }

    public void RemoveServed()
    {
        //remove the first customer of the queue
        System.out.print("Enter the Queue no:");
        Scanner queNo=new Scanner(System.in);
        String QName=queNo.next();
        switch (QName)
        {
            case "1":

                System.out.println("Queue1-CustomerId: 0"  + "
CustomerName: " + queue1[0][0] + " is successfully removed");
                for (int r = 0; r < 4; r++) {
                    queue1[0][r] = "Empty";
                }
                for(int e=0;e<4;e++){
                    if(queue1[e+1][0]!="Empty"){

                        queue1[e][0]=queue1[e+1][0];
                        queue1[e][1]=queue1[e+1][1];
                        queue1[e][2]=queue1[e+1][2];
```

```
                        queue1[e][3]=queue1[e+1][3];

                        for (int r = 0; r < 4; r++) {
                            queue1[e+1][r] = "Empty";
                        }

                    }}

                Repeat();

            case "2":

                System.out.println("Queue2-CustomerId: 0"+"
CustomerName: "+queue2[0][0]+" is successfully removed");
                for (int r=0;r<4;r++) {
                    queue2[0][r]= "Empty";
                }
                for(int e=0;e<4;e++){
                    if(queue2[e+1][0]!="Empty"){

                        queue2[e][0]=queue2[e+1][0];
                        queue2[e][1]=queue2[e+1][1];
                        queue2[e][2]=queue2[e+1][2];
                        queue2[e][3]=queue2[e+1][3];

                        for (int r = 0; r < 4; r++) {
                            queue2[e+1][r] = "Empty";
                        }

                    }}

                Repeat();
            case "3":

                System.out.println("Queue3-CustomerId:0 "+"
CustomerName: "+queue3[0][0]+" is successfully removed");
                for (int r=0;r<4;r++) {
                    queue3[v][r] = "Empty";
                }
                for(int e=0;e<4;e++){
                    if(queue3[e+1][0]!="Empty"){

                        queue3[e][0]=queue3[e+1][0];
                        queue3[e][1]=queue3[e+1][1];
                        queue3[e][2]=queue3[e+1][2];
                        queue3[e][3]=queue3[e+1][3];

                        for (int r = 0; r < 4; r++) {
                            queue3[e+1][r] = "Empty";
                        }

                    }}
                Repeat();
            case "4":

                System.out.println("Queue4-CustomerId:0 "+"
CustomerName: "+queue4[0][0]+" is successfully removed");
                for (int r=0;r<4;r++) {
                    queue4[0][r] = "Empty";
                }
                for(int e=0;e<4;e++){
```

```
                    if(queue4[e+1][0]!="Empty"){

                        queue4[e][0]=queue4[e+1][0];
                        queue4[e][1]=queue4[e+1][1];
                        queue4[e][2]=queue4[e+1][2];
                        queue4[e][3]=queue4[e+1][3];

                        for (int r = 0; r < 4; r++) {
                            queue4[e+1][r] = "Empty";
                        }

                    }}
                Repeat();

            case "5":

                System.out.println("Queue4-CustomerId:0 "+"
CustomerName: "+queue5[0][0]+" is successfully removed");
                for (int r=0;r<4;r++) {
                    queue5[0][r] = "Empty";
                }
                for(int e=0;e<4;e++){
                    if(queue5[e+1][0]!="Empty"){

                        queue5[e][0]=queue5[e+1][0];
                        queue5[e][1]=queue5[e+1][1];
                        queue5[e][2]=queue5[e+1][2];
                        queue5[e][3]=queue5[e+1][3];

                        for (int r = 0; r < 4; r++) {
                            queue5[e+1][r] = "Empty";
                        }

                    }}
                Repeat();


        }


    }

    public void ViewCustomerAlphabeticOrder()
    {//Display the customers name in alphabetical order
//one
        size = queue1.length;

        for (int a = 0; a < size - 1; a++) {

            for (int j = a + 1; j < queue1.length; j++) {
                if (queue1[a][0].compareTo(queue1[j][0]) > 0) {
                    temp = queue1[a][0];
                    queue1[a][0]= queue1[j][0];
                    queue1[j][0] = temp;
                }
            }

        }
        System.out.println("Queue 1:");
        for (int r=0;r<6;r++) {
            if (queue1[r][0] != "Empty") {
```

```
                System.out.print(queue1[r][0]+ " ");

            }
        }
        //two

        size1 = queue2.length;

        for (int a = 0; a < size1 - 1; a++) {

            for (int j = a + 1; j < queue2.length; j++) {
                if (queue2[a][0].compareTo(queue2[j][0]) > 0) {
                    temp1 = queue2[a][0];
                    queue2[a][0] = queue2[j][0];
                    queue2[j][0] = temp1;
                }
            }

        }
        System.out.println("\nQueue 2:");

        for (int r=0;r<6;r++) {
            if(queue2[r][0]!="Empty") {
                System.out.print(queue2[r][0]+" ");
            }
        }
        //three
        size2 = queue3.length;

        for (int a = 0; a < size2 - 1; a++) {

            for (int j = a + 1; j < queue3.length; j++) {
                if (queue3[a][0].compareTo(queue3[j][0]) > 0) {
                    temp2 = queue3[a][0];
                    queue3[a][0] = queue3[j][0];
                    queue3[j][0] = temp2;
                }
            }

        }

        System.out.println("\nQueue 3:");

        for(int r=0;r<6;r++) {
            if (queue3[r][0]!= "Empty") {
                System.out.print(queue3[r][0] + " ");
            }
        }
        //four
        size3 = queue4.length;

        for (int a = 0; a < size2 - 1; a++) {

            for (int j = a + 1; j < queue4.length; j++) {
                if (queue4[a][0].compareTo(queue4[j][0]) > 0) {
                    temp2 = queue4[a][0];
                    queue4[a][0] = queue4[j][0];
                    queue4[j][0] = temp2;
                }
            }
```

```
        }

        System.out.println("\nQueue 4:");

        for(int r=0;r<6;r++) {
            if (queue4[r][0]!= "Empty") {
                System.out.print(queue4[r][0] + " ");
            }
        }
        //five
        size4 = queue5.length;

        for (int a = 0; a < size2 - 1; a++) {

            for (int j = a + 1; j < queue5.length; j++) {
                if (queue5[a][0].compareTo(queue5[j][0]) > 0) {
                    temp2 = queue5[a][0];
                    queue5[a][0] = queue5[j][0];
                    queue5[j][0] = temp2;
                }
            }

        }

        System.out.println("\nQueue 5:");

        for(int r=0;r<6;r++) {
            if (queue5[r][0]!= "Empty") {
                System.out.print(queue5[r][0] + " ");
            }
        }

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

    }
    public void ViewRemaining()
    {
        //display the remaining fuel stock
        remain=fuel-(tot+tot2+tot3+tot4+tot5);
        if(remain==500) {
            System.out.println("Remaining Fuel is  500L" );
            Repeat();
        }
        else {
            System.out.println("Remaining Fuel is " + remain+"L");
            Repeat();
        }
        tot=0;tot2=0;tot3=0;tot4=0;tot5=0;
    }

    public void AddFuel()
    {
        //display total fuel stock after adding fuel
        System.out.print("How many Literes Do you add? : ");
        Scanner adding=new Scanner(System.in);
        int add1=adding.nextInt();
        fuel=remain+add1;
        System.out.println("Total fuel stock: "+fuel);
        Repeat();
```

```
    }

    public void PrintIncome(){
        System.out.println("Queue1 income is Rs."+(430*tot));
        System.out.println("Queue2 income is Rs."+(430*tot2));
        System.out.println("Queue3 income is Rs."+(430*tot3));
        System.out.println("Queue4 income is Rs."+(430*tot4));
        System.out.println("Queue5 income is Rs."+(430*tot5));
        Repeat();
    }

    public void StoreDta()
    {//store the data to a text file
        try {
            BufferedWriter writer= new BufferedWriter(new
FileWriter("filename.txt")); //Create a file
            writer.write("---Customers Details---");
            writer.write("\nQueue 1:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue1[c][0]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue1[c][0]);
                }
            }
            writer.write("\nQueue 2:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue2[c][0]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue2[c][0]);
                }
            }
            writer.write("\nQueue 3:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue3[c][0]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue3[c][0]);
                }
            }
            writer.write("\nQueue 4:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue4[c][0]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue4[c][0]);
                }
            }
            writer.write("\nQueue 5:");//Input data into file
            for (int c = 0; c<6; c++){
                if(queue5[c][0]!="Empty") {
                    writer.write("\nCustomer Id: " + c + " " +
"Customer Name: " + queue5[c][0]);
                }
            }
            writer.close();
            Repeat();
        }catch (IOException e){  //handling errors
            e.printStackTrace();
            Repeat();
        }
    }

    public void LoadData()
```

```
    {//load data from a text file
        try {
            BufferedReader reader= new BufferedReader(new
FileReader("filename.txt"));
            String line;
            while ((line=reader.readLine()) != null){  //Read the
file line by line
                System.out.println(line);
            }
            reader.close();
            Repeat();
        }catch (IOException e){       //handling errors
            e.printStackTrace();
            Repeat();
        }

    }


    public static void main(String[] args) {


    }

}
```

Task3_Driver.java

```
package com.example.w1898939_task2_task3_task4;

public class Task3_Driver extends Task3_Passenger{
    public static void main(String[] args) {
        Driver obj=new Driver();
        obj.first();
        obj.start();
    }
}
```

Task 04 – Source Code

Hello-view.fxml

```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.layout.VBox?>

<VBox alignment="CENTER" spacing="20.0"
xmlns:fx="http://javafx.com/fxml/1"
xmlns="http://javafx.com/javafx/18"
fx:controller="com.example.w1898939_task2_task3_task4.HelloController
">
    <padding>
        <Insets bottom="20.0" left="20.0" right="20.0" top="20.0" />
    </padding>

    <Label fx:id="welcomeText" />
    <Button onAction="#onHelloButtonClick" text="View" />
</VBox>
```

HelloApplication.java

```
package com.example.w1898939_task2_task3_task4;

import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.stage.Stage;

import java.io.IOException;

public class HelloApplication extends Application {
    @Override
    public void start(Stage stage) throws IOException {
        FXMLLoader fxmlLoader = new
FXMLLoader(HelloApplication.class.getResource("hello-view.fxml"));
        Scene scene = new Scene(fxmlLoader.load(), 320, 240);
        stage.setTitle("Hello!");
        stage.setScene(scene);
        stage.show();
    }

    public static void main(String[] args) {
        launch();
    }
}
```

HelloController.java

```
package com.example.w1898939_task2_task3_task4;

import javafx.fxml.FXML;
import javafx.scene.control.Label;

public class HelloController extends Task3_Passenger {
    @FXML
    private Label welcomeText;

    @FXML
    protected void onHelloButtonClick() {
        HelloController obj=new HelloController();
        welcomeText.setText("Customer 1:"+obj.queue1[0][0]);
    }
}
```

Task 04 – Testing

Test Case	Expected Result	Actual Result	Pass/Fail
Fuel Queue Initialized Correctly After program starts, 100 or VFQ	Displays 'empty' for all queue	Displays 'empty' for all Queues	Pass
Enter 101 or VEQ	Display all the empty queues	Display all the empty queues	Pass
Add passenger "Jane" to Queue 2 102 or ACQ Enter Queue: 2 Enter Name: Jane	Display 'Jane added to the queue 2 successfully"	Display 'Jane added to the queue 2 successfully"	Pass
Remove Passenger: Enter Queue no:2 Enter ID of passenger:2	Display "CustomerID:2 CustomerName: Amali is successfully removed"	Display "CustomerID:2 CustomerName: Amali is successfully removed"	Pass
Remove Served Passenger: Enter Queue no:2	Display "Queue1-CustomerId:0 CustomerName: Chamodi is successfully removed"	Display "Queue1-CustomerId:0 CustomerName: Chamodi is successfully removed"	Pass
Enter 105 or VCS	View Customers Sorted in alphabetical order for each queue	View Customers Sorted in alphabetical order for each queue	Pass
Enter 106 or SPD	Data sore to the text file called "filename.txt"	Store the customer id and name of each customer of separate queues	Pass
Enter 107 or LPD	Load data from text file to console	Load data from text file to console	Pass
Enter 108 or STK	View the remaining fuel stock	Remaining Fuel is 6580L	Pass
Enter 109 or AFS	Add fuel stock	Display "How many Literes Do you add? : 200 Total fuel stock: 6780 "	Pass
Enter 999 or EXT	Exit the program	Exit the program	Pass
Add Passenger with First Name, Second Name, Vehicle No ,No. of liters required in task 2.	Display "Enter FirstName:, Enter second name:, Enter vehicle no:,enter required liters:"	Display "Enter FirstName:, Enter second name:, Enter vehicle no:,enter required liters:"	Pass

add customer to the Fuel queue (102 or ACQ) option must select the queue with the minimum length in Task 2.	Automatically place the data for minimum length queues.	Automatically place the data for minimum length queues	Pass
Enter 110 or IFQ	Print the income of each fuel queue	Display ”Queue1 income:Rs.8600,Queue2 income:Rs.8600,Queue3 income:Rs.0”	Pass
After the fuel queues are full “Display full queues full”	After the fuel queues are full “Display full queues full” in each queue and go for waiting queue	After the fuel queues are full “Display full queues full” and go for waiting queue	Pass
View the waiting list of queue	Display names of waiting list	Customer name:null	Fail

Task 04 – Testing – Discussion

First, I initialized each queue as empty. Next, I programmed add customer method. Then I run it and test errors and when a customer adds, does it add to the array. Then I created the view menu method. After I looked for what are errors. And then I created the view empty queues. Then I thought How to remove a customer. I initialized the customer data to empty and then I checked to increase the position of each customer after one is removed. And then I implemented the removed a served customer. There I removed the first customer of the queue. And again, to increase the position of each customer after one is removed. Then I go to implement the customer's names in alphabetical order for each queue. Then I created view remaining fuel stock and add fuel stock functions. At the end I implemented the store program data into a text file called "filename.txt" and load the data form the file to the console. I used 1D arrays for task1. Because this program has a small database. I wanted to store only customer name. And I used 2D arrays for task 2. Because this program has usually a large database. further I used inheritance, encapsulation as OOP concept in task2. And also, if, else-if to check conditions. Then I used while and for loops for repetitions. I think version 2 is the best solution for this task.

Self-Evaluation form

Criteria	Component marks	Expected Marks
Task 1 2.5 marks for each option (10 options) Menu works correctly	25 5	25 5
Task 2 Fuel Queue class correctly implemented. Passenger class correctly implemented. Income of each Fuel Queue correctly implemented.	**10** **8** **7**	10 8 7
Task 3 Waiting list queue implementation “102 or ACQ’: Add” works correctly “104 or PCQ’: Delete” works correctly Circular queue implementation	**5** **5** **5** **5**	**5** **5** **0**
Task 4 JavaFX (10) GUI for Viewing the Fuel Queue 5 GUI for Search the passenger	**5** **5**	**2** **0**
Task 6 Test case coverage and reasons	**10**	**10**
Coding Style (Comments, indentation, style)	**5**	**5**
Total	**100**	**82**