



**“UNIVERSIDAD DE LAS FUERZAS ARMADAS”**

***ESPE***

***SOFTWARE ENGINEERING***

***OBJECT - ORIENTED PROGRAMMING***

***TOPIC:***

---

*WORKSHOP 29 - design a table of your project in paper*

***TEACHER:***

*PHD. EDISON LASCANO*

***NRC:***

*4680*

***DATE:***

*July 14, 2022*

***SEDE MATRIZ SANGOLQUÍ***

***QUITO-ECUADOR***

***2022***

1. Arguello Espinosa Joel Daniel

0

Stylists					
Id	Name	Number	Payment	Appointment	Address
1	Joel	048787438	455	13/07/2022	Quito
2	GAbriel	09453654...	679	14/06/2022	Quito
3	Alonso	0934465464	786	19/07/2022	Quito

Projects   Files   Services

History

```
stylists = new ArrayList<>();
Stylist stylist;
//read from database
stylist = new Stylist("1724545876", "Joel", 983727652, 400, "17/08/2022", "Quito");
stylists.add(stylist);
stylist = new Stylist("1776578876", "Carlos", 983727652, 470, "19/08/2022", "Quito");
stylists.add(stylist);
stylist = new Stylist("1787879876", "Maria", 983727652, 800, "27/08/2022", "Quito");
stylists.add(stylist);
stylist = new Stylist("1787980378", "Confirma Cesar", 983727652, 200, "16/08/2022", "Quito");
stylists.add(stylist);

Object[][] objects = new Object[stylists.size()][6];

for (int i = 0; i < stylists.size(); i++) {
    objects[i][0] = stylists.get(i).getIdentificationCard();
    objects[i][1] = stylists.get(i).getName();
    objects[i][2] = stylists.get(i).getNumber();
    objects[i][3] = stylists.get(i).getPayment();
    objects[i][4] = stylists.get(i).getAppointment();
    objects[i][5] = stylists.get(i).getAddress();
}

tableStylists.setModel(new javax.swing.table.DefaultTableModel(
    objects,
    new String[]{

        "run-single",
        "run-single",
        "run-single",
        "run-single",
        "run-single",
        "run-single"
    }
));
```

Stylists

Id	Name	Number	Payment	Appointment	Address
1724545876	Joel	983727652	400.0	17/08/2022	Quito
1776578876	Carlos	983727652	470.0	19/08/2022	Quito
1787879876	Maria	983727652	800.0	27/08/2022	Quito
1787980378	Confirma ...	983727652	200.0	16/08/2022	Quito

Load

FrmTableStylist(), btrLoadActionPerformed(ActionEvent evt), initComponents(), loadStylistsTable(), main(String[] args), btnLoad, jLabel1, jScrollPane1

Output - Stylesirella (run-single) x

2. Bedon Guevara Roberto Alexander

10

NAME	STYLIST ADDRESS	NUMBER	PAYMENT
JUAN RODRIGUEZ	Av. PICHINCHA	0972295621	\$ 500
MARIA CHACON	Mariscal Sucre	0978144213	\$ 450

Stylists

Name	Adress	Number	Payment
Juan Rodriguez	Av. Pichincha	098547563	500
Maria Chacon	Mariscal Sucre	095412876	450

```

public void populateStylistTable() {
    ArrayList<Stylist> stylists;
    stylists = new ArrayList<>();
    Stylist stylist;

    stylist = new Stylist("Andrea Ruiz", "Chillogallo");
    stylists.add(stylist);
    stylist = new Stylist("Maria Montalvo", "Las casas");
    stylists.add(stylist);
    stylist = new Stylist("Luis Ojeda", "San Carlos");
    stylists.add(stylist);
    stylist = new Stylist("Antonio Calvache", "Chillogallo");
    stylists.add(stylist);

    Object[][] objects = new Object[stylists.size()][4];
    for (int i = 0; i < stylists.size(); i++) {
        objects[i][0] = stylists.get(i).getName(),
        objects[i][1] = stylists.get(i).getAddress(),
        objects[i][2] = stylists.get(i).getNumber(),
        objects[i][3] = stylists.get(i).getPayment();
    }

    tblStylist.setModel(new javax.swing.table.DefaultTableModel(
        objects,
        new String[]{
            "Name",
            "Address",
            "Number",
            "Payment"
        }
    ));
}

```

Stylists

Name	Address	Number	Payment
Andrea Ruiz	Chillogallo	0	500.0
Maria Montalvo	Las casas	2	600.0
Luis Ojeda	San Carlos	2	350.0
Antonio Calvache	Chillogallo	2	380.0

Load Stylist

3. Burbano Pacheco Luis Ariel 10

Stylist				
10 Name Number Payment Appointment Address				
1 Luis	09830229039	400	10/01/2021	Outo
2 Jorge	0981467177	600	14/06/2021	English
3 Gisela	0971475108	477	19/06/2021	Outo

Id	Name	Number	Payment	Appointment	Address
1	Luis	0983272037	400	13/07/2022	Quito
2	Jorge	09453623...	700	14/06/2022	Sangolqui
3	Gabriel	0934567633	450	19/07/2022	Quito

```

public void btnLoadActionPerformed(ActionEvent evt) {
    try {
        Connection conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/cooplatina", "root", "1234");
        Statement st = conn.createStatement();
        String sql = "SELECT * FROM stylists";
        ResultSet rs = st.executeQuery(sql);
        DefaultTableModel model = new DefaultTableModel();
        model.addColumn("Id");
        model.addColumn("Name");
        model.addColumn("Number");
        model.addColumn("Payment");
        model.addColumn("Appointment");
        model.addColumn("Address");
        while (rs.next()) {
            Vector v = new Vector();
            v.add(rs.getString("id"));
            v.add(rs.getString("name"));
            v.add(rs.getString("number"));
            v.add(rs.getDouble("payment"));
            v.add(rs.getDate("appointment"));
            v.add(rs.getString("address"));
            model.addRow(v);
        }
        tableStylists.setModel(model);
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, e);
    }
}

```

#### 4. Caiza Pullotasig Widinson Danilo 10

Stylist				
	Name	Id	Salary	Address
1.	Carlos Peir	665169	35,8	Quito
2.	Jhon Armas	15891	34,5	Quito

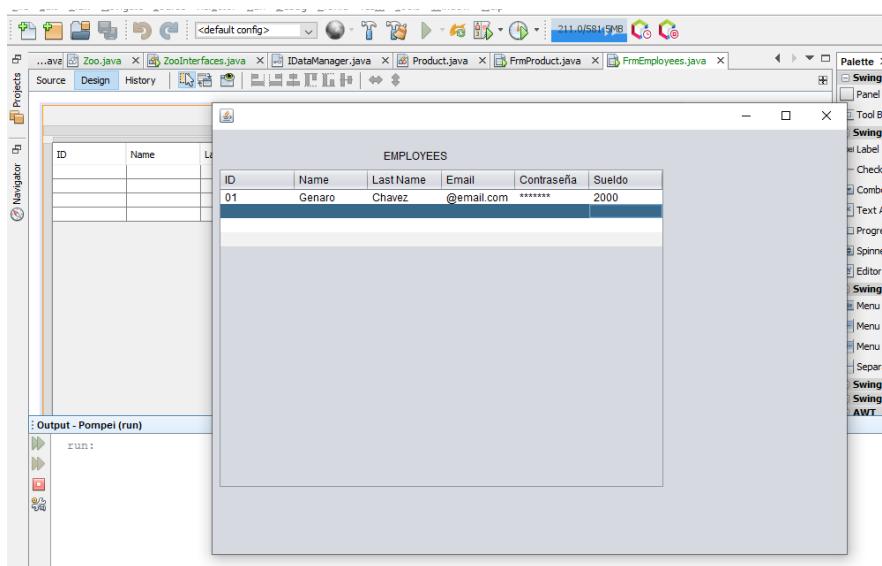
Customer				
	Name	Id	number	appointment
1.	Jhon Arun	0550180	1	17/7/22
2.	Carlos Jeir	011587	2	14/7/22

- □ ×

Stylists					
	Name	Id	Salary	Address	
1	Carlos Peir	6618	35.8	Quito	
2	Jhon Armas	15891	34.5	Quito	

5. Chavez Escudero Genaro David 10

Employee						
ID	Name	L. N.	Ext.	Prov.	Sueldo	
1	Genaro	Chavez	Qavlor	*** -	\$ 400	
2	Alejo	Condor	@out.com	-----	600	



```

    ...
    populateEmployeesTable();
}

public void populateEmployeesTable() {
    ArrayList<Employee> employees;
    employees = new ArrayList<>();
//read from database
    Employee employee = new Employee("Genaro", "Chavez");
    employees.add(employee);
    Employee employee2 = new Employee("Mateo", "Condor");
    employees.add(employee2);
    Employee employee3 = new Employee("Daniel", "Chiri");
    employees.add(employee3);

    Object[][] objects = new Object[employees.size()][6];

    for (int i = 0; i < employees.size(); i++) {
        objects[i][0] = i;
        objects[i][1] = employees.get(i).getName();
        objects[i][2] = employees.get(i).getLastName();
        objects[i][3] = employees.get(i).getEmail();
        objects[i][4] = employees.get(i).getPassword();
        objects[i][5] = "400";
    }
}

/* Creates new form FrmEmployees */
public FrmEmployees() {
    initComponents();
    populateEmployeesTable();
}

public void populateEmployeesTable() {
    ArrayList<Employee> employees;
    employees = new ArrayList<>();
//read from database
    Employee employee = new Employee("Genaro", "Chavez", "gchavez@gmail.com", "****");
    employees.add(employee);
    Employee employee2 = new Employee("Mateo", "Condor", "mat_condor@gmail.com", "****");
    employees.add(employee2);
    Employee employee3 = new Employee("Daniel", "Chiri", "dani_@gmail.com", "****");
    employees.add(employee3);

    Object[][] objects = new Object[employees.size()][6];

    for (int i = 0; i < employees.size(); i++) {
        objects[i][0] = i;
        objects[i][1] = employees.get(i).getName();
        objects[i][2] = employees.get(i).getLastName();
        objects[i][3] = employees.get(i).getEmail();
        objects[i][4] = employees.get(i).getPassword();
        objects[i][5] = "400";

        tblEmployees.setModel(new javax.swing.table.DefaultTableModel(
            objects,

```

6. Chicaiza Ortiz Frank Sebastian 0
7. Chiriboga Zurita Daniel Isai 10

*Customers*

<i>Id</i>	<i>Name</i>	<i>Phone</i>	<i>Adrees</i>
1	Mateo Flores	0996245331	Comite el pueblo
2	Steven Reyes	0987238159	Valle de los chillos.

*Customers*

<i>Id</i>	<i>Name</i>	<i>Phone</i>	<i>Adrees</i>
1	Mateo Flores	0996245331	Comite el pueblo
2	Steven Reyes	0987238159	Valle de los chillos.

[Return](#)

```

    ...
    populateCustomersTable();
}

public void populateCustomersTable() {
    ArrayList<Customer> customers;
    customers = new ArrayList<>();
    Customer customer;

    customer = new Customer(1, "Mateo Flores", 996245331, "Comite del pueblo");
    customers.add(customer);
    customer = new Customer(2, "Steven Reyes", 987238159, "Valle de los chillos");
    customers.add(customer);

    Object[][] objects = new Object[customers.size()][4];
    for (int i = 0; i < customers.size(); i++) {
        objects[i][0] = customers.get(i).getId();
        objects[i][1] = customers.get(i).getName();
        objects[i][2] = customers.get(i).getPhone();
        objects[i][3] = customers.get(i).getAdrees();
    }

    tblCustomer.setModel(new javax.swing.table.DefaultTableModel(
        objects,
        new String[]{
            "Id", "name", "phone", "adrees"
        }
    ));
}

```

Customers			
Id	name	phone	adrees
1	Mateo Flores	996245331	Comite del pueblo
2	Steven Reyes	987238159	Valle de los chillos

## 8. Condor Sosa Mateo Javier 10

Cantidad	descripción	PU	PT
4	Pan	0,15	0,60
6	Cachorro	0,12	0,72
2	envoltorio	0,14	0,28
		Total a pagar	1,70

cantidad	descripcion	pu	pt
1	pan	0.15	0.15
2	cachito	0.12	0.24

9. Granda Velasquez Carlos Javier 10

Design Table

Fruits				
Id	Name	Quantity	Cost.	Color
1	Apple	1 Lb	1,50 \$	Red
2	Pear	2 Lb	\$ 1,00	Green
3	Banana	3 Lb	\$ 1,25	Yellow

Id	Name	Quantity	Cost	Color
1	Apple	1 Lb	1.50	Red
2	Pear	2 Lb	1.00	Green
3	Banana	3 Lb	1.25	Yellow

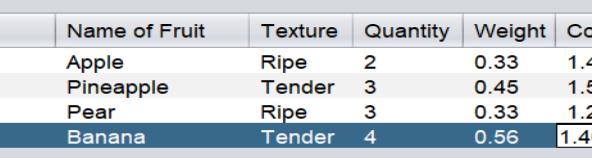
The screenshot shows the Apache NetBeans IDE interface. The top menu bar includes File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help, and a search bar. The left sidebar contains the Projects X tab, listing various Java files under the FruitApp project. Below it are Source Packages, Test Packages, and Libraries. The main workspace shows the Start Page with the FrmCustomers.java file open. The code editor highlights the populateCustomersTable() method, which creates a Customer object for each fruit and adds it to a list. The preview pane on the right displays a JTable titled "FRUIT" with the following data:

	Name	Quantity	Cost	Color
1	apple	1	0.90	red
2	pear	2	5.3	green
3	banana	3	10.2	yellow

A "Return" button is located at the bottom right of the preview pane.

10. Ibarra Gaona Ronny Joel 10

Fruit				
Name of Fruit	Color	Amount	Weight (lbs)	Cost (\$)
Apple	red	3	0.44	1.32
Pineapple	yellow	2	0.55	1.10
Pear	green	2	0.55	1.10



The screenshot shows a Java Swing application window titled "FRUITS". The window has a standard title bar with minimize, maximize, and close buttons. The main content area displays a table with the following data:

Id	Name of Fruit	Texture	Quantity	Weight	Cost
1	Apple	Ripe	2	0.33	1.40
2	Pineapple	Tender	3	0.45	1.50
3	Pear	Ripe	3	0.33	1.25
4	Banana	Tender	4	0.56	1.40

FRUITS

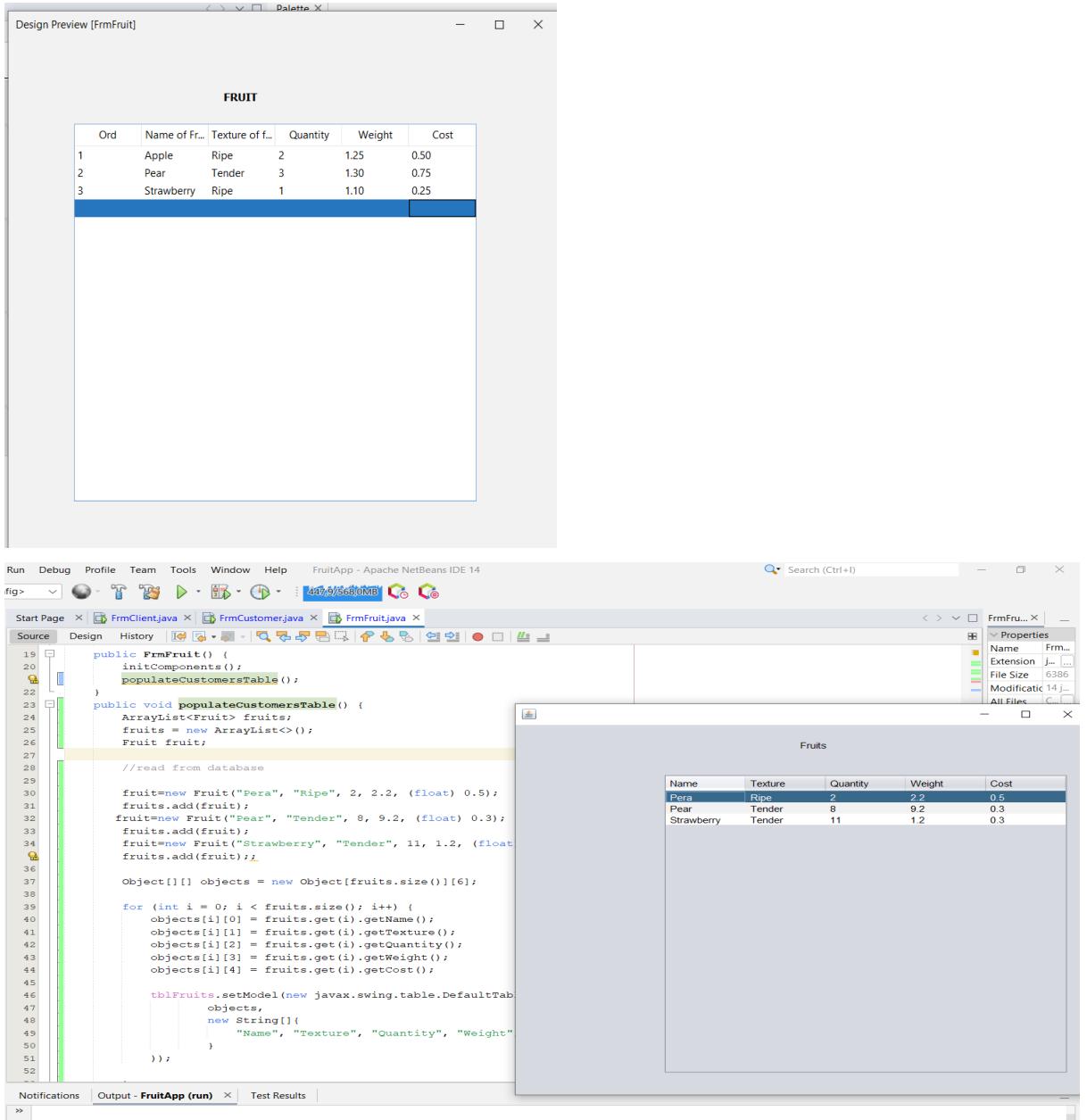
Name of Fruit	Texture	Quantity	Weight(Lb)	Cost(\$)
Apple	Ripe	2.0	0.33	1.5
Pineapple	Tender	3.0	0.45	1.5
Pear	Ripe	3.0	0.33	1.25
Banana	Tender	4.0	0.56	1.4

Load

### 11. Imbaquinga Guaña Jose Ricardo 9

Fruit

Ord	Name of Fruit	Texture of Fruit	Quantity	Weight	Cost
1	Apple	Ripe	2	1,25	0.50
2	Pear	Tender	3	1,30	0.75
3	Strawberry	Ripe	1	1,30	0.25



12. Loor Mercado Cesar Ignacio 10

Passenger		
Identification	Name	Email
0350599440	fabian	fab.tragmail
1751856945	Laura	Lmz@gmail.com
1301929582		
1753846982	Juan	Juan.Ln@gmail.com
1301500847		

CoopLatina

### Passengers

Identification	Name	Email
850599440	Cesar Loor	ciloor2@gmail.com
1751856945	Laura Ricarte	Lau.23@hotmail.com
1301929582	Bethy Sabando	bethy-sab@hotmail.com
1753846982	Fabian Lopez	fab.lop2@gmail.com
1301500847	Juan Perez	Juan-Perez38@hotmail.com

13. Mendoza Arteaga Aldric Mateo

0

Passengers		
ID	Name	Phonenumber
1003689321-2	Aldric	0986271212
0501822331-3	Jose	0976231456
1001998725-4	Maria	0998653417

Passenger passenger;

Passengers		
Id	Name	Phonenumber
1003689321	Aldric	0986271212
0501822331	Jose	0976231456
1001998725	Maria	0998653417

Passenger passenger;

Passengers		
Id	Name	Phonenumber
181213	Alejandro Tapia	0986271212
191723	Joan Ortega	0976231456
197634	Gabriel Alarcon	0998653417

```
passenger = new Passenger("181213", "Alejandro Tapia", "0986271212");
passengers.add(passenger);
passenger = new Passenger("191723", "Joan Ortega", "0976231456");
passengers.add(passenger);
passenger = new Passenger("197634", "Gabriel Alarcon", "0998653417");
passengers.add(passenger);
```

Full name	Email	Cell Phone	Route	PAY
Steven Molina steve@gmail.com		099871347	Carapungo	5,66\$
Diego Silva diego@yahoo.com		099712713	Calderón	7,54\$
Raul Quimbiaco raul@hotmail.com		0998134215	La Loma	8,99\$

— □ ×

### PASSENGERS

Full Name	Email	Cell Phone	Route	Pay
Gustavo Moli...	steve@gmail...	0998472619	Calderon	5,66\$
Juan Silva	juan@yahoo....	098971623	Carapungo	7,64\$
Raul Quimbi...	raul@hotmail...	098671261	La Loma	8,99\$

[Return to main menu](#)

PASSENGERS

Name	Address	e-mail	Cell phone number	Total pay (\$)
Steven Molina	carapungo	steve@gmail.com	09986212	7.54
Juan Silva	calderon	juan@yahoo.com	09871231	7.54
Raul Quimbiulco	La Loma	steve@hotmail.com	099871231	8.99
Yeremy Quesada	El centro	yeremy@gmail.com	09883211	19.99

[Return to main menu](#)

```

public void populatePassengersTable() {
    ArrayList<Passenger> passengers;
    passengers = new ArrayList<>();
    Passenger passenger;

    passenger = new Passenger("Steven Molina", "carapungo", "steve@gmail.com", "09986212", 7.54F);
    passengers.add(passenger);
    passenger = new Passenger("Juan Silva", "calderon", "juan@yahoo.com", "09871231", 7.54F);
    passengers.add(passenger);
    passenger = new Passenger("Raul Quimbiulco", "La Loma", "steve@hotmail.com", "099871231", 8.99F);
    passengers.add(passenger);
    passenger = new Passenger("Yeremy Quesada", "El centro", "yeremy@gmail.com", "09883211", 19.99F);
    passengers.add(passenger);

    Object[][] objects = new Object[passengers.size()][5];

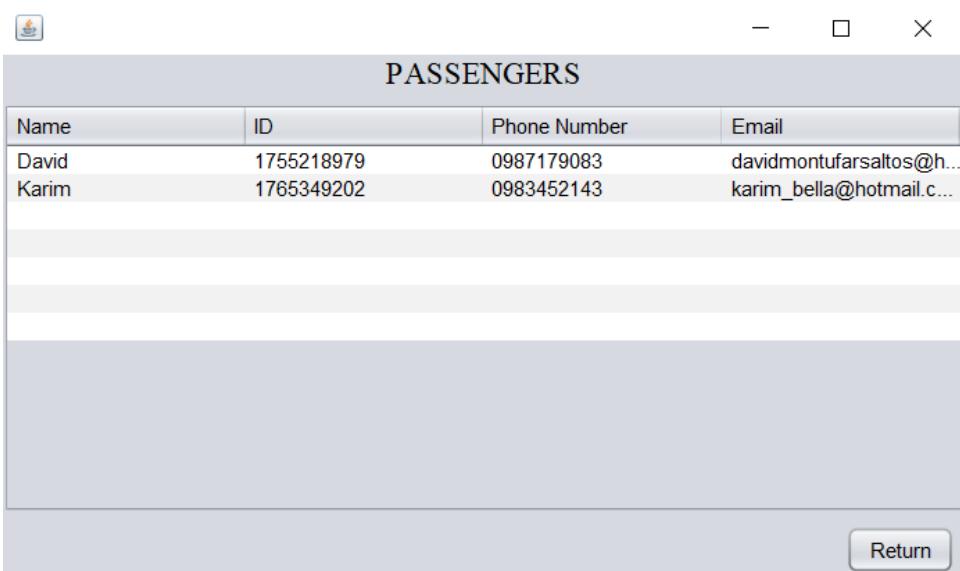
    for (int i = 0; i < passengers.size(); i++) {
        objects[i][0] = passengers.get(i).getName();
        objects[i][1] = passengers.get(i).getAddress();
        objects[i][2] = passengers.get(i).getEmail();
        objects[i][3] = passengers.get(i).getCellPhone();
        objects[i][4] = passengers.get(i).getPayment();
    }
}

ec.edu.espe.coopLatina.view.FrmPassengers > ● populatePassengersTable >
run: Test Results
Coop-Latina (run) × Coop-Latina (run) #2 × Coop-Latina (run) #3 × Coop-Latina (run) #4 × Coop-Latina (run) #5 ×
BUILD SUCCESSFUL (total time: 2 seconds)

```

15. Montufar Saltos David Ariel 9

David Monticelos		Día	Mes	Año
Name	ID	Phone Number	Email	
► David	1755218979	0987179083	► David.monticelosaltos@gmail.com	
► Karim	1708869891	0971469001	► Karim.02@hotmail.com.	



```
public void populateCustomersTable() {
    ArrayList<Customer> customers;
    customers = new ArrayList<>();

    Customer customer = new Customer("David Montufar", "1755218943", "0987179083", "davidmontufar@hotmail.com");
    customers.add(customer);
    customer = new Customer("Karim Cadena", "1756541894", "0986463483", "karim22@hotmail.com");
    customers.add(customer);
    customer = new Customer("Mateo Gallardo", "1759450543", "0965465465", "mateofr@hotmail.com");
    customers.add(customer);

    Object[][] objects = new Object[customers.size()][4];

    for (int i = 0; i < customers.size(); i++) {
        objects[i][0] = customers.get(i).getName();
        objects[i][0] = customers.get(i).getID();
        objects[i][0] = customers.get(i).getPhoneNumber();
        objects[i][0] = customers.get(i).getEmail();
    }
}
```

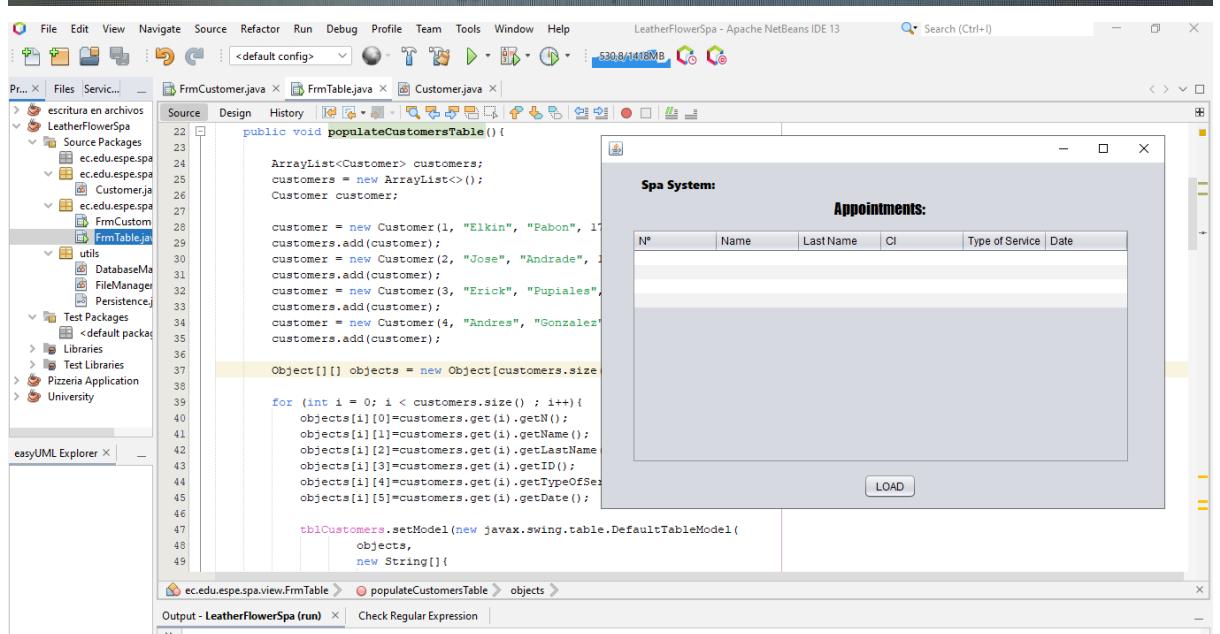
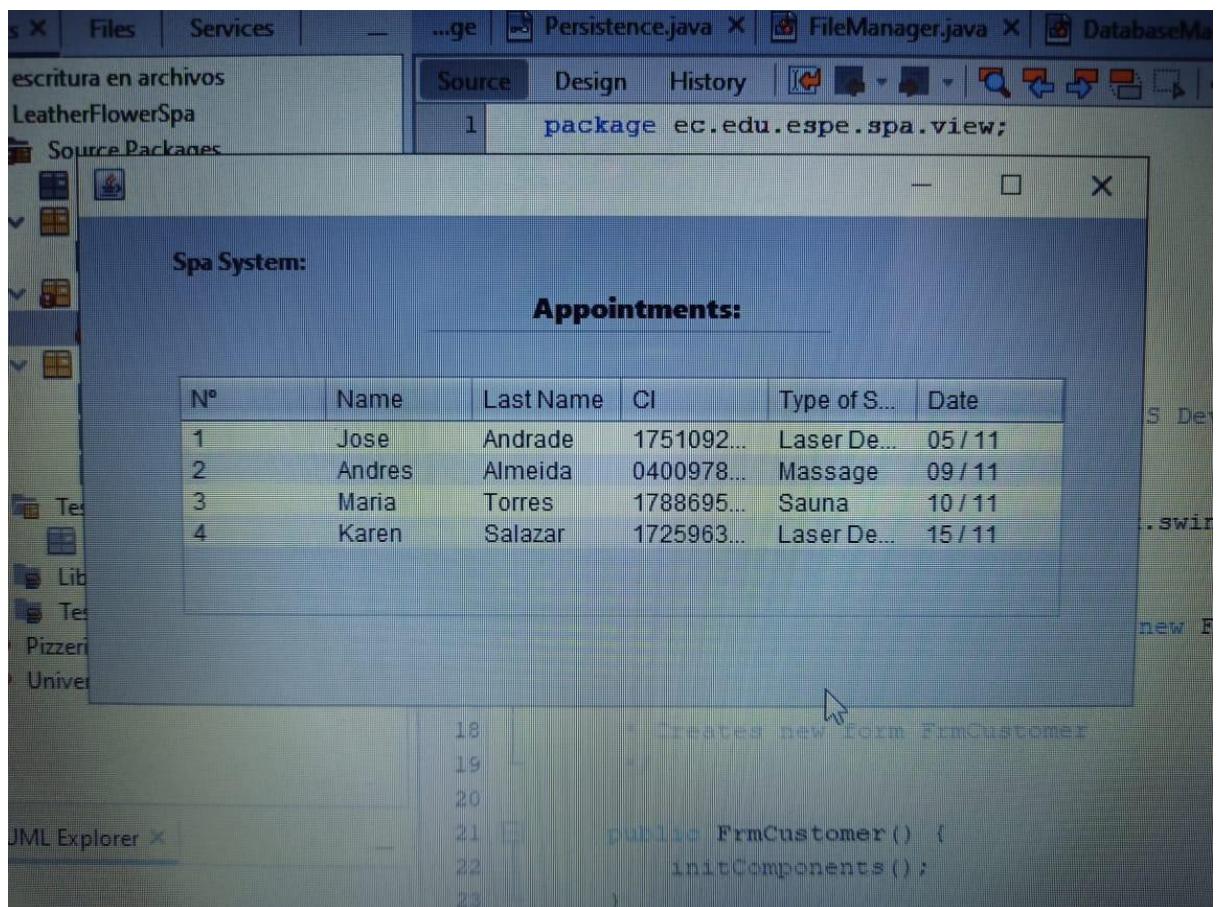
16. Moreira Vinueza Erick Patricio 9

<u>SERVICE</u>					
ID	First Name	Last Name	Date	Type	Price
1315110534	Erick	Moraira	7/11/22	Laser Depilation	10.00\$
1753368899	Steven	Poto	7/12/22	Massage	15.00\$
1303252298	Sonia	Veliz	7/13/22	Acne Treatment	25.00\$

17. Pabon Gonzalez Elkin Andres

5

Appointments					
Nº	Name	Lastname	CI	Type of Service	Date
1	Jose	Andrade	17512055	Laser Depilation	05/11
2	Andres	Almeida	17510930	Massage	09/11
3	Maria	Torres	17902001	Sauna	09/11



LeatherFlowerSpa - Apache NetBeans IDE 13

FrmCustomer.java | FrmTable.java | Customer.java

```

public void populateCustomersTable() {
    ArrayList<Customer> customers;
    customers = new ArrayList<>();
    Customer customer;

    customer = new Customer(1, "Elkin", "Pabon", 1719492054);
    customers.add(customer);
    customer = new Customer(2, "Jose", "Andrade", 1781592254);
    customers.add(customer);
    customer = new Customer(3, "Erick", "Pupiales", 1741092004);
    customers.add(customer);
    customer = new Customer(4, "Andres", "Gonzalez", 1751392054);
    customers.add(customer);

    Object[] objects = new Object[customers.size()];
    for (int i = 0; i < customers.size(); i++) {
        objects[i][0] = customers.get(i).getID();
        objects[i][1] = customers.get(i).getName();
        objects[i][2] = customers.get(i).getLastName();
        objects[i][3] = customers.get(i).getDate();
        objects[i][4] = customers.get(i).getTypeOfService();
        objects[i][5] = customers.get(i).getPrice();
    }

    tblCustomers.setModel(new javax.swing.table.DefaultTableModel(
        objects,
        new String[]{
            "N", "Name", "Lastname", "ID", "Type of service", "Date"
        }
    ));
}

```

Spa System:  
Appointments:

N	Name	Lastname	ID	Type of service	Date
1	Elkin	Pabon	1719492054	Laser Depilat...	10 July
2	Jose	Andrade	1781592254	Massage	5 August
3	Erick	Pupiales	1741092004	Sauna	6 August
4	Andres	Gonzalez	1751392054	Depilation	17 August

LOAD

### 18. Ponce Arguello Diego Armando 9 Customers = clientes, clients is wrong

Ticket	Full Name	Number	Date	Service	Total
36	Diego Ponce	0996131005	14/07/2022	Massage	\$ 10,00
37	Elkin Pabon	0988251582	15/07/2022	Massage	\$ 10,00
38	Steven Polo	0996481314	16/07/2022	Massage	\$ 10,00
39	Erick Moreiru	0915944259	17/07/2022	Massage	\$ 10,00
40	Edison Lowluno	0919142022	18/07/2022	Massage	\$ 10,00

CUSTOMERS

Ticket	Name	Number	Date	Service	Total
36	Diego P...	0996131...	14/07/20...	Massage	\$10
37	Elkin Pa...	0988251...	15/07/20...	Massage	\$10
38	Steven ...	0968945...	16/07/20...	Massage	\$10

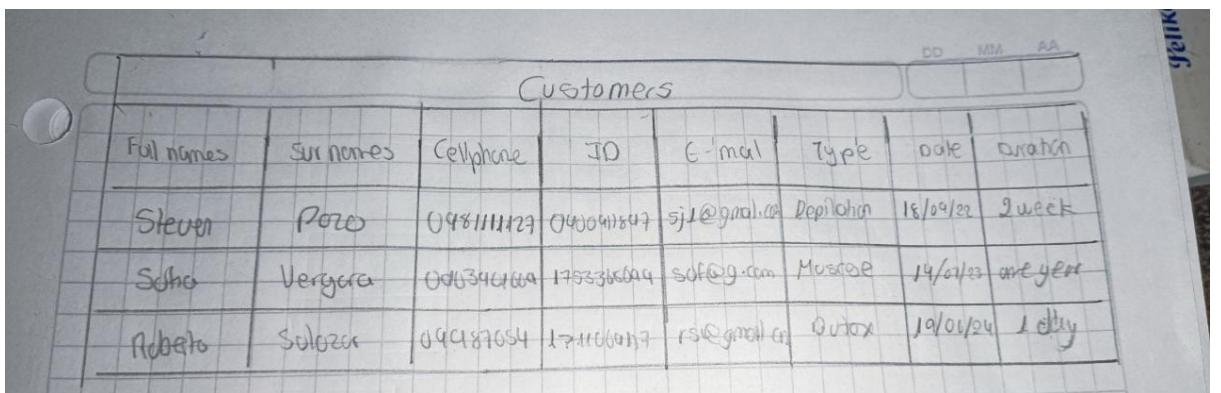
The screenshot shows the NetBeans IDE interface. The code editor displays Java code for populating a table with customer data. A preview window titled 'CUSTOMERS' shows a table with columns: Tickect, Name, Number, Date, Service, and Total. The data includes entries for Diego Ponce, Elkin Pabon, Steven Pozo, Erick Moreira, and others.

```

19
20 }
21
22 public void populateCustomersTable() {
23     ArrayList<Customer> customers;
24     customers = new ArrayList<>();
25     Customer customer;
26
27     //read from database
28     customer = new Customer(36, "Diego Ponce", "0996131005", "14/07/2022",
29     customers.add(customer);
30     customer = new Customer(37, "Elkin Pabon", "0988251582", "15/07/2022",
31     customers.add(customer);
32     customer = new Customer(38, "Steven Pozo", "0985462148", "16/07/2022",
33     customers.add(customer);
34     customer = new Customer(39, "Erick Moreira", "0965478865", "17/07/2022"
35     customers.add(customer);
36
37     Object[][] objects = new Object[customers.size()][6];
38
39     for (int i = 0; i < customers.size(); i++) {
40         objects[i][0] = customers.get(i).getTickect();
41         objects[i][1] = customers.get(i).getName();
42         objects[i][2] = customers.get(i).getNumber();
43         objects[i][3] = customers.get(i).getDate();
44     }

```

### 19. Pozo Analuisa Steven Jefferson 10



The screenshot shows the NetBeans IDE interface with a running application window titled 'CUSROMERS'. The window displays a table with columns: Full name, Surname, Cellphone, ID, E-mail, Type of Service, Cost, and Date. The data matches the handwritten note above. The Java code in the editor shows the implementation of the application logic.

```

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
W26 MenuAndTables - Apache NetBeans IDE 13
Projects Services
FrmCustomer.java FrmCustomersTable.java
Source Design History
CUSROMERS
Full name Surname Cellphone ID E-mail Type of Service Cost Date
Juanito Alcachofa 0981515127 175336899 juanito@espe.edu.deplation Laser 23.4$ 2 week
Elsa Patto 0963901669 0400911897 elsa@gmail.com Botox 100$ 1 month
Chilindrina Patata 0998765433 1753378987 papas@gmail.com Massage 20$ 1 day

```

**UPLOAD INFORMATION:**

The screenshot shows the Apache NetBeans IDE interface with the following details:

- File Menu:** File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help.
- Toolbar:** Includes icons for file operations like Open, Save, Print, and a search bar labeled "Search (Ctrl+F)".
- Project Explorer:** Shows various projects including "FileManager", "FruitApp", "HW12DatabasePersistence2", "HW15TemporaryFields", "JavaApplication1", "LeatherFlowerSystem", "LeatherFlowerSystemProject", "modifiable", "UMLDiagrams", "Class Diagrams", "UnitTest\_Areas", "UnitTestAreas", "W22Equation", "W24 GUI", and "W26 MenuAndTables".
- Code Editor:** Displays the content of Customer.java, which contains code to initialize a list of customers with their details (name, surname, cellphone, ID, email).
- Output Window:** Shows the output for the "W26 MenuAndTables (run)" configuration, indicating the application is "running...".
- Modal Dialog:** A window titled "CUSROMERS" is displayed, showing a table with the following data:

full name	Surname	Cellphone	ID	E-mail
Steven Pozo	Pozo	+1753368099	0981515126	stevenpozo@gmail.com
Esteban Ponce	Ponce	+17110691717	0987171723	estevan@gamil.com
Emily Pabón	Pabón	+1711079888	0987654432	emily@gmail.com
Fernanda Moreira	Moreira	+1711079888	098736252	fernanda@gmail.com

**Buttons:** "Load Information" button at the bottom of the dialog.



The screenshot shows the Apache NetBeans IDE interface with the following details:

- File Menu:** File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help.
- Toolbar:** Includes icons for file operations like Open, Save, Print, and a search bar labeled "Search (Ctrl+F)".
- Project Explorer:** Shows the project structure with packages like FileManager, FruitApp, HW12 DataPersistence2, HW15 TemporaryFields, JavaApplication1, leatherFlowerSystem, LeatherFlowerSystemProject, modifiable, UMLDiagrams, Class Diagrams, UnitTest\_Areas, UnitTestAreas, W22Equation, W24 GUI, and W26 MenuAndTables. Under W26 MenuAndTables, there are Source Packages, Images, and e.edu.espe.Spa.controller.
- Code Editor:** The active file is FrmCustomer.java, which contains Java code to manage a list of Customer objects. It includes a section for reading data from a database and adding new Customer instances to an ArrayList.
- Output Window:** Shows the output for the "W26 MenuAndTables (run)" configuration, indicating the application is running.
- Table View:** A JTable component titled "CUSTOMERS" is displayed, showing the following data:

Full Name	Surname	cellphone	ID	Email
Steven	Pozo	0981515126	1753368099	stevenpozo@gmail.com
Esteban	Ponce	0987171723	17110691717	estevan@gmail.com
Emily	Pabón	0987654432	1711079888	emily@gmail.com
Fernanda	Moreira	0928725222	098736252	ffernanda@gmail.com

A "Load Information" button is located at the bottom of the table view.



20 Quilumbaquin Lanchimba Jairo Smith 10

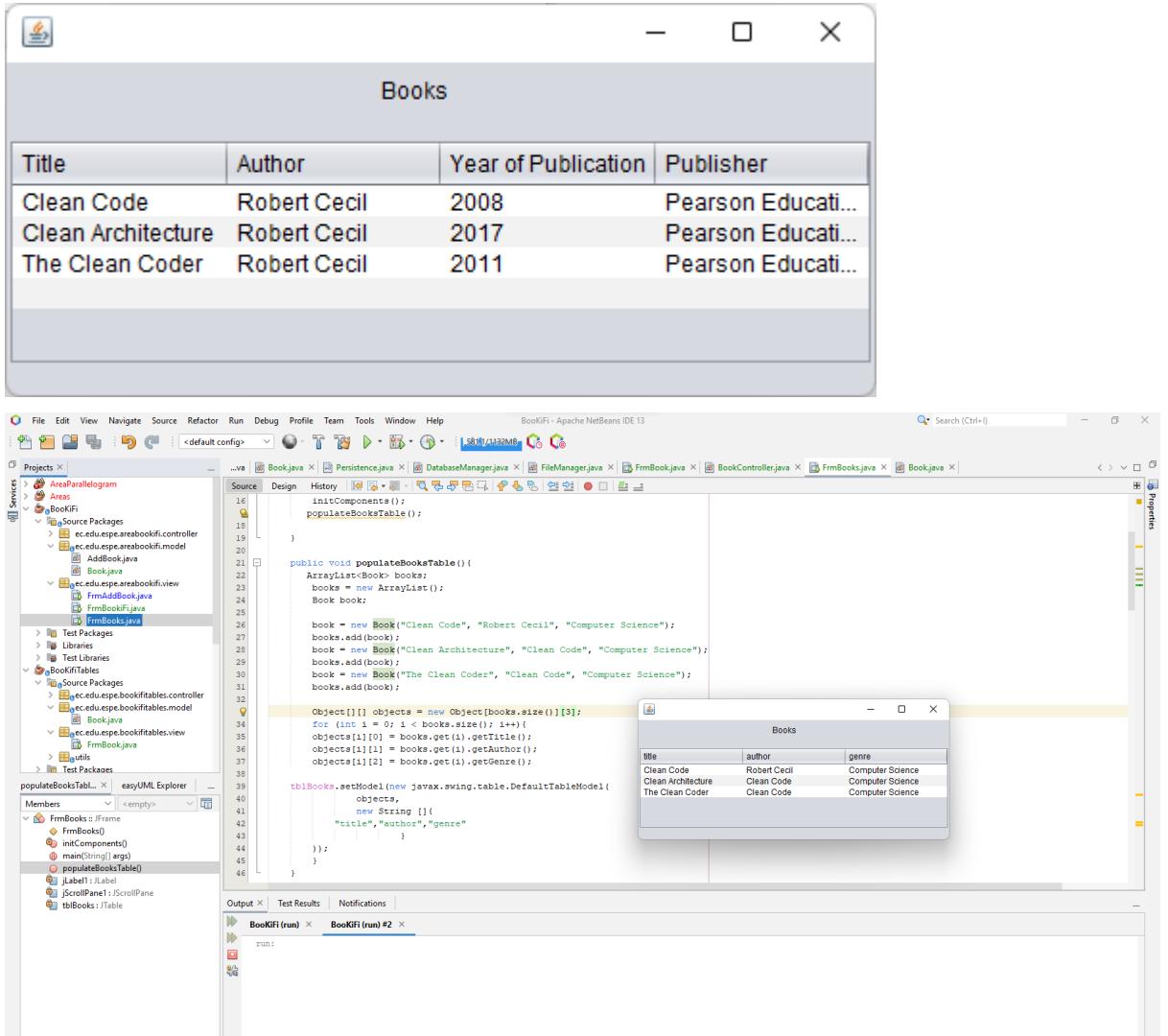
Users					
ID	Name	Age	Phone	Address	Email
1	Jairo	21	0993809393	Calderon	jairo@gmail.com
2	Luis	21	0997304545	El Inca	Luis@gmail.com
3	Josue	21	0993809187	El Valle	Josue@gmail.com

Users

ID	Name	Age	Phone	Address	Email
1	Jairo	21	0993809393	Calderon	Jairo@gmail.com
2	Luis	21	0995608439	El Inca	Luis@gmail.com
3	Josue	21	0995604391	El Valle	Josue@gmail.com
4	Diego	18	0985628512	Calderon	diego@gmail.com

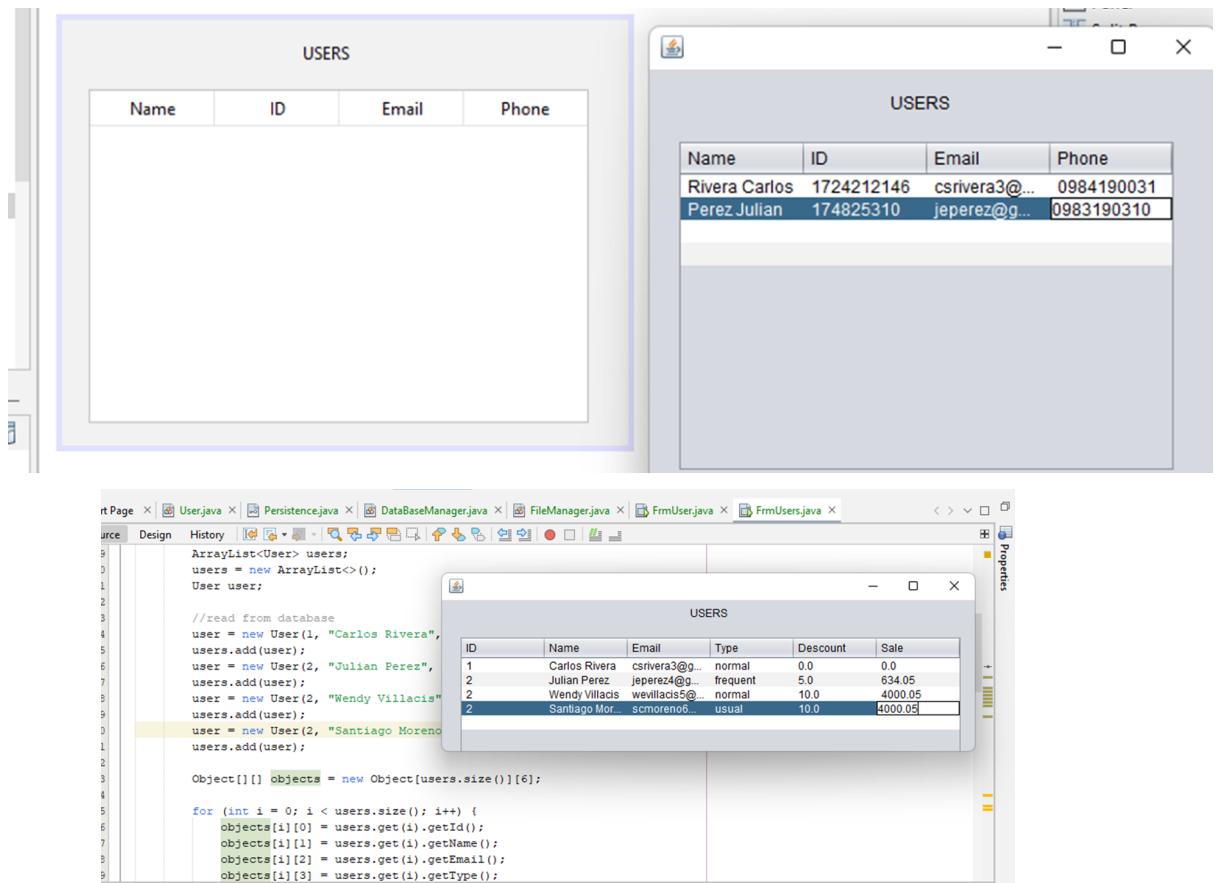
21. Quimbiulco Juan Diego 10

Books				
Title	Author	Year of Publication	Publisher	
Clean Code	Robert Cecil	2008	Pearson Education	
Clean Architecture	Robert Cecil	2017	Pearson Education	
The Clean Coder	Robert Cecil	2011	Pearson Education	



22. Rivera Espin Carlos Sebastian 10

Ord	Name	ID	Email	Phone	Age
1	Rivera Carlos	1724212146	csrivera@gmail	0984190031	20
2	Perez Julian	174825310	jepered@gmail	0983190310	17



23. Silva Velasquez Raul Andres 10

User			
Id	Name	Phone number	age
1234	Raul	0970967634	20
1235	Juan	699990999	20

The screenshot shows two instances of the Apache NetBeans IDE. The top instance is running a Java Swing application titled "USERS". The application has a simple GUI with a table containing four rows of data:

Id	Name	Phone Number	Age
1233	pepito	099999999	20

The bottom instance of NetBeans shows the Java code for the "FrmUsers.java" file, which populates the table. The code uses an ArrayList to store User objects and then sets the table model with columns for Id, Name, and Phone Number.

```

22 users = new ArrayList<>();
23 User user;
24
25 //read from database
26 user = new User(123, "Juanito", 89887879);
27 users.add(user);
28 user = new User(125, "Steven", 1231231);
29 users.add(user);
30 user = new User(124, "Raul", 2312341);
31 users.add(user);
32 user = new User(126, "Daniela", 341243);
33 users.add(user);
34 Object[][] objects = new Object[users.size()][4];
35
36 for (int i = 0; i < users.size(); i++) {
37     objects[i][0] = users.get(i).getId();
38     objects[i][1] = users.get(i).getName();
39     objects[i][2] = users.get(i).getPhoneNumber();
40 }
41
42 tblUsers.setModel(new javax.swing.table.DefaultTable
43     objects,
44     new String[]{
45         "Id", "Name", "Phone Number"
46     });

```

24. Sosa Romero Diana Carolina 9

A handwritten table on lined paper with five rows of data. The columns are labeled "id", "Name", "Address", "CellPhone", and "Email". The data is as follows:

1729519801	Pedro Suarez	San Juan	093354120	pedro.suarez@gmail.com
1917552802	Andres Alvarez	Guamán	0903759326	andres.alvarez@gmail.com
1820511205	Jaime Sanchez	San Bartolo	0912483340	jaime.sanchez@gmail.com
1512119535	Carmen Salazar	San Pedro	0940796213	carmen.salazar@gmail.com

**CUSTOMERS**

ID	NAME	ADRESS	CELLPHONE	EMAIL
1929519801	Pedro Gutierrez	San Juan	0986578935	pedrogutierrez...
1917552805	Andres Alvarez	Guamani	0903759326	andresalvarez...
1820511205	Jaime Sanchez	San Bartolo	0912438840	jaimesanchez...
1512119535	Carmen Salazar	San pedro	094079513	carmensalaza...

**CustomerController.java**

```

public void populateCustomersTable() {
    DefaultTableModel model = (DefaultTableModel) tblCustomers.getModel();

    List<Customer> customers = customerService.getCustomers();
    Object[] row;

    for (Customer c : customers) {
        row = new Object[5];
        row[0] = c.getId();
        row[1] = c.getName();
        row[2] = c.getAddress();
        row[3] = c.getCellPhone();
        row[4] = c.getEmail();

        model.addRow(row);
    }
}

```

## 25. Tituaña Moreno Daniela Lissette 9

**CASHIERS**

ID	Name	Address	Cellphone	E-Mail
1752530999	Daniela Tituaña	Abdon Calderon, E1-34	0989028782	dltiuana@hotmail.com
1725354857	Christofer Carvajal	Chillogallo - Calle 15	0994580403	chiscar28@gmail.com
1717035506	Carlos Villamarín	Solanda, Calle Prin...	0983540700	VillaCar1@yahoo.c...

Screenshot of Apache NetBeans IDE 13 showing code in Persistence.java and a JTable preview window.

```

33     cashiers.add(cashier);
34
35     Object[][] objects = new Object[cashiers.size()][5];
36
37     for (int i=0; i<cashiers.size(); i++){
38         objects[i][0]=cashiers.get(i).getId();
39         objects[i][1]=cashiers.get(i).getName();
40         objects[i][2]=cashiers.get(i).getAddress();
41         objects[i][3]=cashiers.get(i).getCellPhone();
42         objects[i][4]=cashiers.get(i).getEmail();
43
44         tblCashierTable.setModel(new javax.swing.table.DefaultT
45             objects,
46             new String[]{
47                 "Id", "Name", "Address", "Cellphone", "E-ma
48             });
49
50     }
51
52 }
53
54 /**
55 * This method is called from within the constructor to initialize the form.
56 * WARNING: Do NOT modify this code. The content of this method is always
57 * regenerated by the Form Editor.
58 */
59
60 @SuppressWarnings("unchecked")
61 // <editor-fold defaultstate="collapsed" desc="Generated Code">
62 private void initComponents() {

```

Find:DefaultTableModel 2 matches X

Notifications Test Results Output 02:51 p.m. ESP 14/07/2022

## 26. Toapanta Velasquez Martin Honorio 4

ID	Full Name	E-mail	Cellphone
1715042198	José Luis Roa	jlp.22@gmail.com	0984227667
1709035412	Anahí López	a1.230@gmail.com	0987264546
1726070228	Martín Toapanta	mhtoapanta1@espe.edu.ec	0979211689

Customer			
ID	Full Name	E-mail	Cellphones
1709035414	Martín Toapanta	mhtoapanta1@espe.edu.ec	0984227667
17150496335	Mateo Salazar	mz.21@gmail.com	0998456321

## 27. Villavicencio Campoverde Bolívar Josue 10

WS 201				
Ord.	ID	Name / last Name	Cellphone	Email
1	1715439695	Emurgo Sanchez	0935648971	Emurgo @hotmail.com
2	1929568301	Luis Villanueva	0948959617	Villan@v@hotmail.com
3	1726009798	Josue Villa	0961311098	Villa@hotalicam

ID	Full Name	Cellphone	E-mail	Company	Address	Ord.
1726449794	Enrique Sanchez	593+9557658...	Enrique@hot...	Nestle	Quito	1
1945448534	Luis Villamar	593+893838008	Villamar@hot...	La favorita	Guayaquil	2
1534990001	Josue Villa	593+516198013	Villa@hotmail....	Red bull	EEUU	3

The screenshot shows the Apache NetBeans IDE interface with the following details:

- File Menu:** File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help.
- Title Bar:** ViveresPedroSystem - Apache NetBeans IDE 13, Search (Ctrl+F).
- Toolbar:** Includes icons for file operations like Open, Save, Print, and a magnifying glass for search.
- Project Explorer:** Shows the project structure with files like Cashier.java, Customer.java, Inventory.java, Money.java, Product.java, Provider.java, Providers.java, and ViveresPedro.java.
- Code Editor:** Displays the `FrameProvider.java` file with Java code for managing customers. A tooltip "Provider" is shown over the `Provider` class name.
- Output:** Shows the message "ESPE2020S-OOP-SW-4600 - C:\Users\ESPE2020S-OOP-SW-4600\NetBeansProjects\ViveresPedro\src\ViveresPedro\FrameProvider.java:1: error: package ViveresPedro does not exist" and "Run: build successful".
- Preview:** A JTable preview window titled "Provider" displays customer data with columns: Id, Full Name, CellPhone, Address, and E-mail. The data is as follows:

ID	Full Name	CellPhone	Address	E-mail
1726449794	Josue Villavicencio	0981311998	Quito	bjvillavicencio1@espe.edu.ec
1234567890	Alfredo Salazar	0987654321	Quito	Alfredo@hotmail.com
1725449893	Elkin Pabón	0935439945	Guayaquil	elkin@gmail.com
1153646689	Erick Pavon	0947812457	Cuenca	Erick@gmail.com

