ASSIGNMENT

SUBMITTED BY,

ANGEL SEBASTIAN – 11

DEVANANDANA AK – 22

DEVIKA SAJI - 24

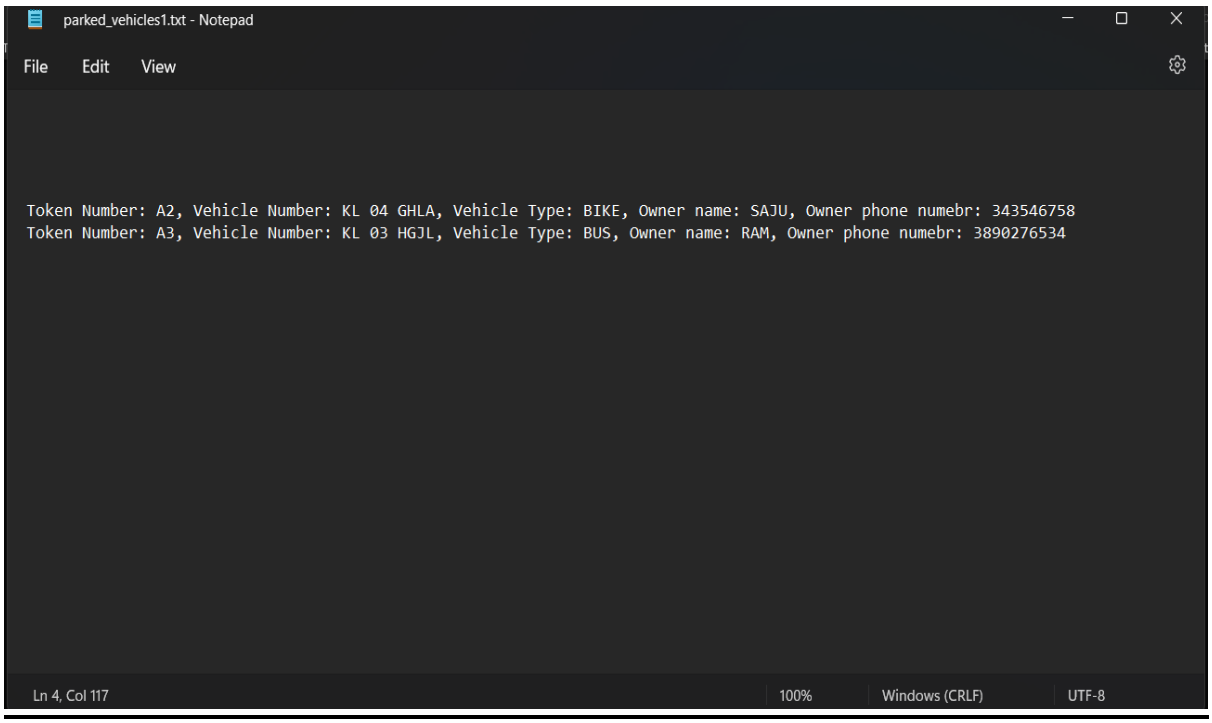
PHASE 1

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

public class ParkingManagementSystem1 {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        while (true) {
            System.out.println("Parking Management System Menu:");
            System.out.println("1. Add a vehicle");
            System.out.println("2. Delete a vehicle");
            System.out.println("3. Search for a vehicle");
            System.out.println("4. Display all vehicle details");
            System.out.println("5. Exit");
            System.out.print("Enter your choice (1, 2, 3, 4 or 5): ");

            int choice = scanner.nextInt();
            scanner.nextLine();
            switch (choice) {
                case 1:
                    addVehicle(scanner);
                    break;
                case 2:
                    deleteVehicle(scanner);
                    break;
                case 3:
                    searchVehicle(scanner);
                    break;
                case 4:
                    displayAllVehicles();
                    break;
                case 5:
                    System.out.println("Exiting program.");
                    System.exit(0);
                    break;
                default:
                    System.out.println("Invalid choice. Please enter 1, 2, 3, 4,5.");
```

```java
                    System.out.println("Invalid choice. Please enter 1, 2, 3, 4,5.");
            }
        }
    }


    private static void addVehicle(Scanner scanner) {
        try (BufferedWriter writer = new BufferedWriter(new FileWriter("parked_vehicles1.txt", true))) {
            System.out.print("Token Number: ");
            String tokenNumber = scanner.nextLine();


            System.out.print("Vehicle Number: ");
            String vehicleNumber = scanner.nextLine();

            System.out.print("Vehicle Type: ");
            String vehicleType = scanner.nextLine();


            System.out.print("Owner name: ");
            String ownerName = scanner.nextLine();

            System.out.print("Owner Phone Number: ");
            String phNumber = scanner.nextLine();


            writer.write("Token Number: " + tokenNumber + ", Vehicle Number: "+vehicleNumber+ ", Vehicle Type: " +vehicleType+  ", Owner name: "+ownerName+  ", Owner phone numebr: "+phNumb
            writer.newLine();

            System.out.println("Vehicle added successfully.");
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

     private static void deleteVehicle(Scanner scanner) {
    try {
        System.out.print("Enter the Token Number of the vehicle to delete: ");
        String vehicleToDelete = scanner.nextLine();
```

```
        String vehicleToDelete = scanner.nextLine();

        File file = new File("parked_vehicles1.txt");
        RandomAccessFile raf = new RandomAccessFile(file, "rw");

        String currentLine;
        long currentPosition = 0;
        boolean vehicleFound = false;

        while ((currentLine = raf.readLine()) != null) {
            if (currentLine.contains("Token Number: " + vehicleToDelete)) {
                System.out.println("Vehicle removed: " + currentLine);
                vehicleFound = true;
                break;
            }
            currentPosition = raf.getFilePointer();
        }

        if (vehicleFound) {
            raf.seek(currentPosition - currentLine.length() - 2);
            for (int i = 0; i < currentLine.length(); i++) {
                raf.writeByte(' ');
            }

            System.out.println("Vehicle details deleted successfully.");
        } else {
            System.out.println("Vehicle not parked here!!");
        }

        raf.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
}


    private static void searchVehicle(Scanner scanner) {
        try (BufferedReader reader = new BufferedReader(new FileReader("parked_vehicles1.txt"))) {
            System.out.print("Enter the Token Number of vehicle to search: ");
```

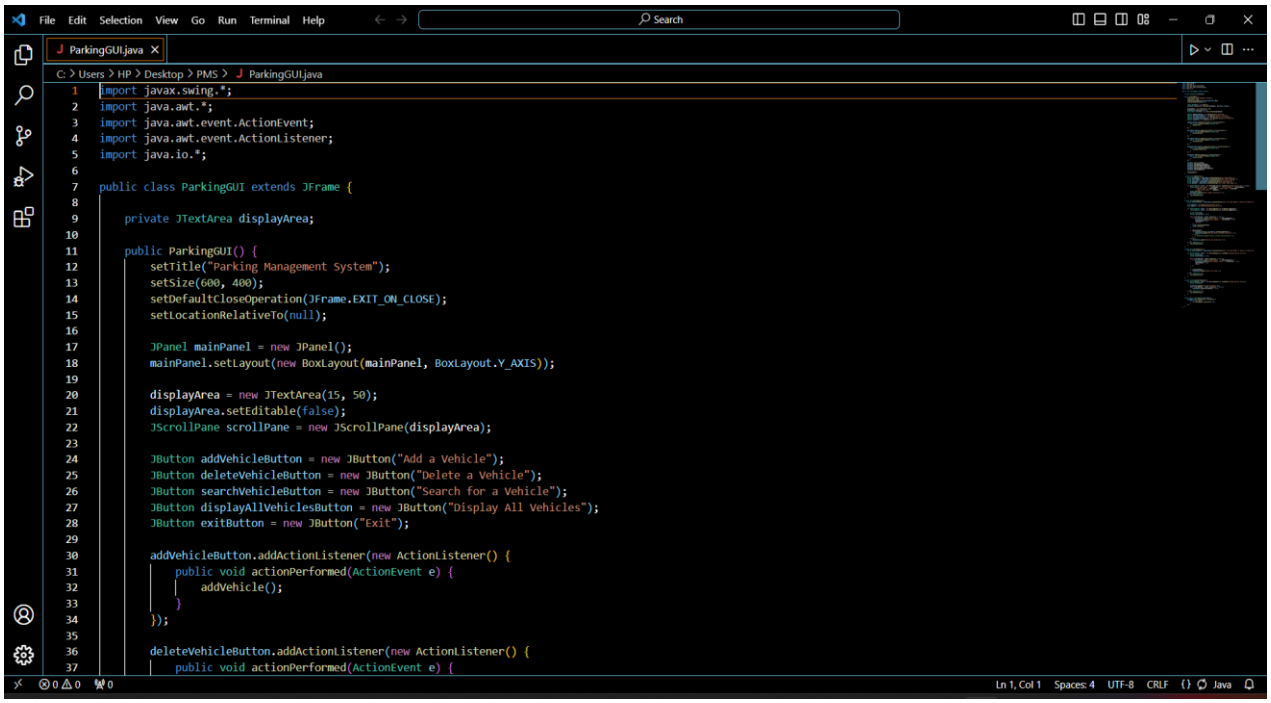
OUTPUT



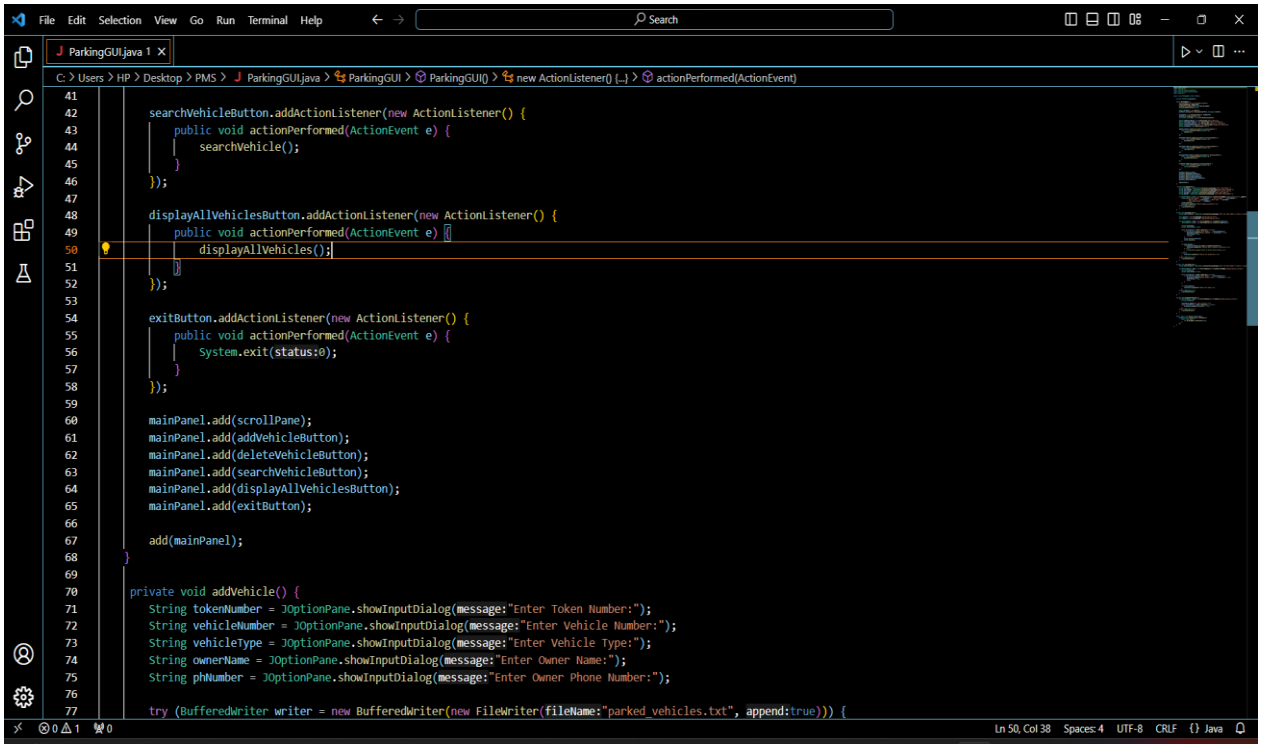
```
Parking Management System Menu:
PS C:\lab\ParkingManagement.java> javac ParkingManagementSystem1.java
PS C:\lab\ParkingManagement.java> java ParkingManagementSystem1
Parking Management System Menu:
1. Add a vehicle
2. Delete a vehicle
3. Search for a vehicle
4. Display all vehicle details
5. Exit
Enter your choice (1, 2, 3, 4 or 5): 1
Token Number: A1
Vehicle Number: KL 09 HGDF
Vehicle Type: CAR
Owner name: SHIFA
Owner Phone Number: 84653457487
Vehicle added successfully.
Parking Management System Menu:
1. Add a vehicle
2. Delete a vehicle
3. Search for a vehicle
4. Display all vehicle details
5. Exit
Enter your choice (1, 2, 3, 4 or 5): 1
Token Number: A2
Vehicle Number: KL 04 GHLA
Vehicle Type: BIKE
Owner name: SAJU
Owner Phone Number: 343546758
Vehicle added successfully.
Parking Management System Menu:
1. Add a vehicle
2. Delete a vehicle
3. Search for a vehicle
4. Display all vehicle details
5. Exit
Enter your choice (1, 2, 3, 4 or 5): 1
Token Number: A3
Vehicle Number: KL 03 HGJL
Vehicle Type: BUS
Owner name: RAM
```

```
Owner name: RAM
Owner Phone Number: 3890276534
Vehicle added successfully.
Parking Management System Menu:
1. Add a vehicle
2. Delete a vehicle
3. Search for a vehicle
4. Display all vehicle details
5. Exit
Enter your choice (1, 2, 3, 4 or 5): 4
All Vehicle Details:



Token Number: A1, Vehicle Number: KL 09 HGDF, Vehicle Type: CAR, Owner name: SHIFA, Owner phone numebr: 84653457487
Token Number: A2, Vehicle Number: KL 04 GHLA, Vehicle Type: BIKE, Owner name: SAJU, Owner phone numebr: 343546758
Token Number: A3, Vehicle Number: KL 03 HGJL, Vehicle Type: BUS, Owner name: RAM, Owner phone numebr: 3890276534
Parking Management System Menu:
1. Add a vehicle
2. Delete a vehicle
3. Search for a vehicle
4. Display all vehicle details
5. Exit
Enter your choice (1, 2, 3, 4 or 5): 3
Enter the Token Number of vehicle to search: A3
Vehicle details found: Token Number: A3, Vehicle Number: KL 03 HGJL, Vehicle Type: BUS, Owner name: RAM, Owner phone numebr: 3890276534
Parking Management System Menu:
1. Add a vehicle
2. Delete a vehicle
3. Search for a vehicle
4. Display all vehicle details
5. Exit
Enter your choice (1, 2, 3, 4 or 5): 2
Enter the Token Number of the vehicle to delete: A1
Vehicle removed: Token Number: A1, Vehicle Number: KL 09 HGDF, Vehicle Type: CAR, Owner name: SHIFA, Owner phone numebr: 84653457487
Vehicle details deleted successfully.
Parking Management System Menu:
1. Add a vehicle
```

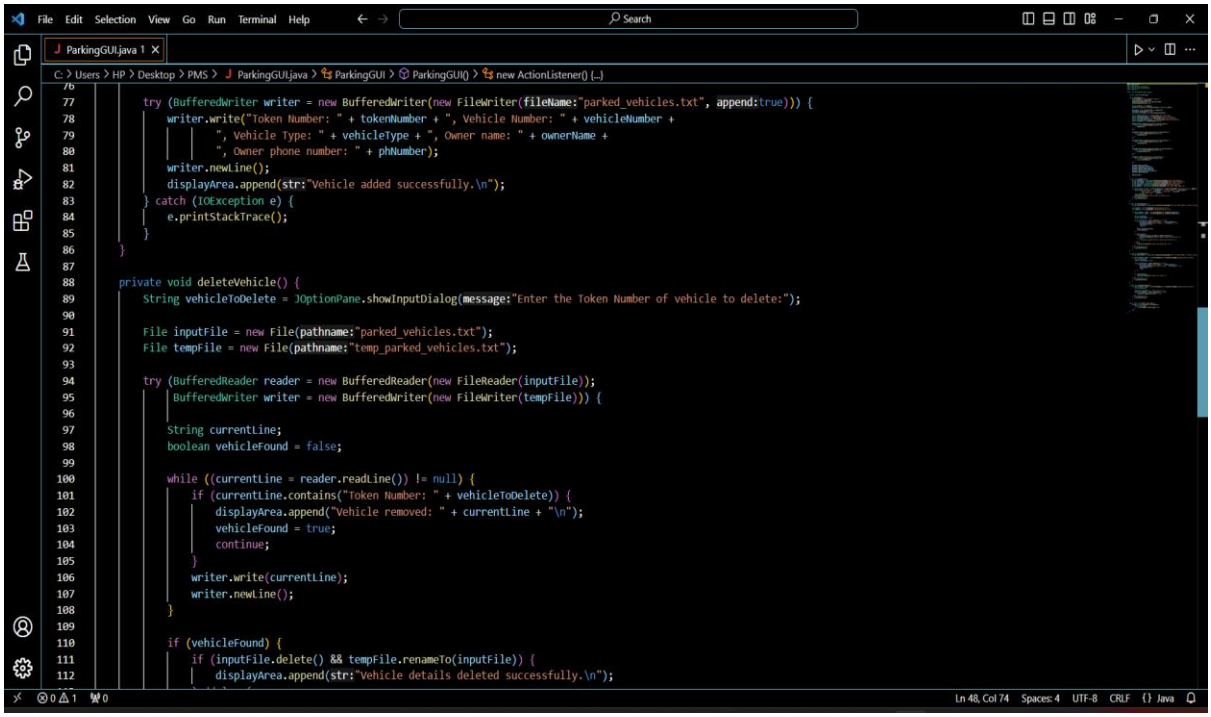
PHASE 2



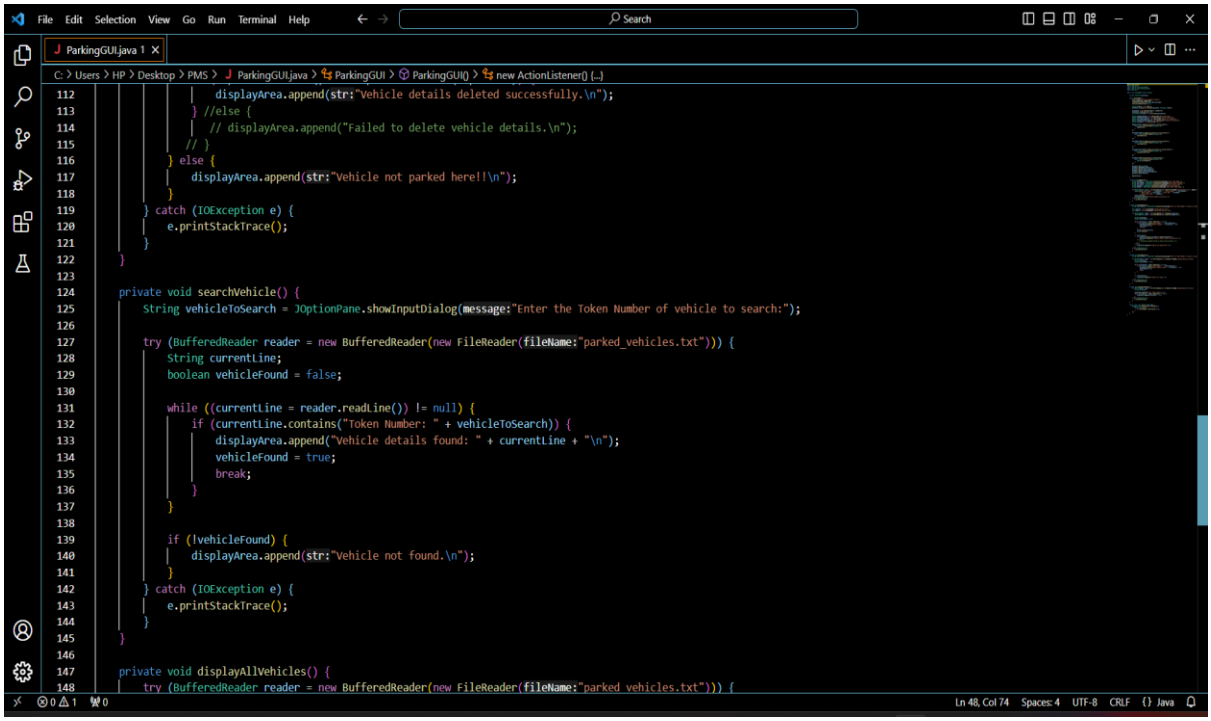
```java
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.io.*;

public class ParkingGUI extends JFrame {

    private JTextArea displayArea;

    public ParkingGUI() {
        setTitle("Parking Management System");
        setSize(600, 400);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLocationRelativeTo(null);

        JPanel mainPanel = new JPanel();
        mainPanel.setLayout(new BoxLayout(mainPanel, BoxLayout.Y_AXIS));

        displayArea = new JTextArea(15, 50);
        displayArea.setEditable(false);
        JScrollPane scrollPane = new JScrollPane(displayArea);

        JButton addVehicleButton = new JButton("Add a Vehicle");
        JButton deleteVehicleButton = new JButton("Delete a Vehicle");
        JButton searchVehicleButton = new JButton("Search for a Vehicle");
        JButton displayAllVehiclesButton = new JButton("Display All Vehicles");
        JButton exitButton = new JButton("Exit");

        addVehicleButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                addVehicle();
            }
        });

        deleteVehicleButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
```



```java
        searchVehicleButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                searchVehicle();
            }
        });

        displayAllVehiclesButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                displayAllVehicles();
            }
        });

        exitButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                System.exit(status:0);
            }
        });

        mainPanel.add(scrollPane);
        mainPanel.add(addVehicleButton);
        mainPanel.add(deleteVehicleButton);
        mainPanel.add(searchVehicleButton);
        mainPanel.add(displayAllVehiclesButton);
        mainPanel.add(exitButton);

        add(mainPanel);
    }

    private void addVehicle() {
        String tokenNumber = JOptionPane.showInputDialog(message:"Enter Token Number:");
        String vehicleNumber = JOptionPane.showInputDialog(message:"Enter Vehicle Number:");
        String vehicleType = JOptionPane.showInputDialog(message:"Enter Vehicle Type:");
        String ownerName = JOptionPane.showInputDialog(message:"Enter Owner Name:");
        String phNumber = JOptionPane.showInputDialog(message:"Enter Owner Phone Number:");

        try (BufferedWriter writer = new BufferedWriter(new FileWriter(fileName:"parked_vehicles.txt", append:true))) {
```



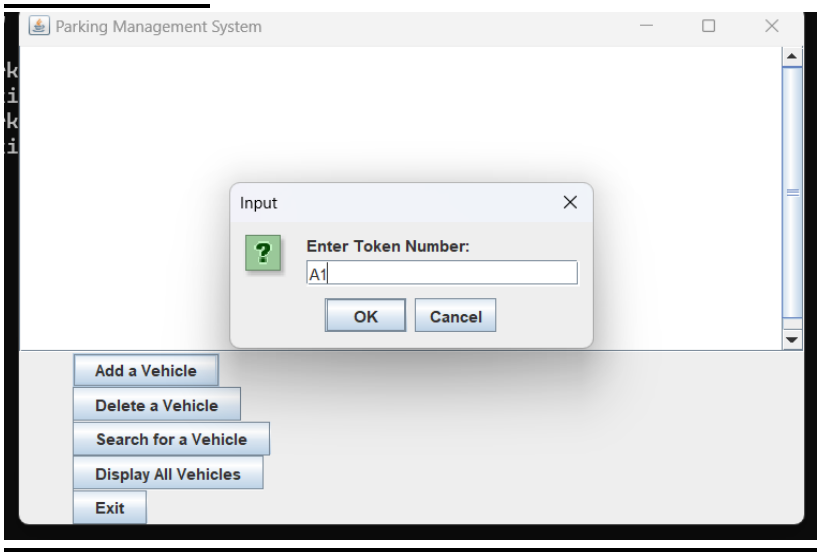
```
        try (BufferedWriter writer = new BufferedWriter(new FileWriter(fileName:"parked_vehicles.txt", append:true))) {
            writer.write("Token Number: " + tokenNumber + ", Vehicle Number: " + vehicleNumber +
                         ", Vehicle Type: " + vehicleType + ", Owner name: " + ownerName +
                         ", Owner phone number: " + phNumber);
            writer.newLine();
            displayArea.append(str:"Vehicle added successfully.\n");
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    private void deleteVehicle() {
        String vehicleToDelete = JOptionPane.showInputDialog(message:"Enter the Token Number of vehicle to delete:");

        File inputFile = new File(pathname:"parked_vehicles.txt");
        File tempFile = new File(pathname:"temp_parked_vehicles.txt");

        try (BufferedReader reader = new BufferedReader(new FileReader(inputFile));
             BufferedWriter writer = new BufferedWriter(new FileWriter(tempFile))) {

            String currentLine;
            boolean vehicleFound = false;

            while ((currentLine = reader.readLine()) != null) {
                if (currentLine.contains("Token Number: " + vehicleToDelete)) {
                    displayArea.append("Vehicle removed: " + currentLine + "\n");
                    vehicleFound = true;
                    continue;
                }
                writer.write(currentLine);
                writer.newLine();
            }

            if (vehicleFound) {
                if (inputFile.delete() && tempFile.renameTo(inputFile)) {
                    displayArea.append(str:"Vehicle details deleted successfully.\n");
```

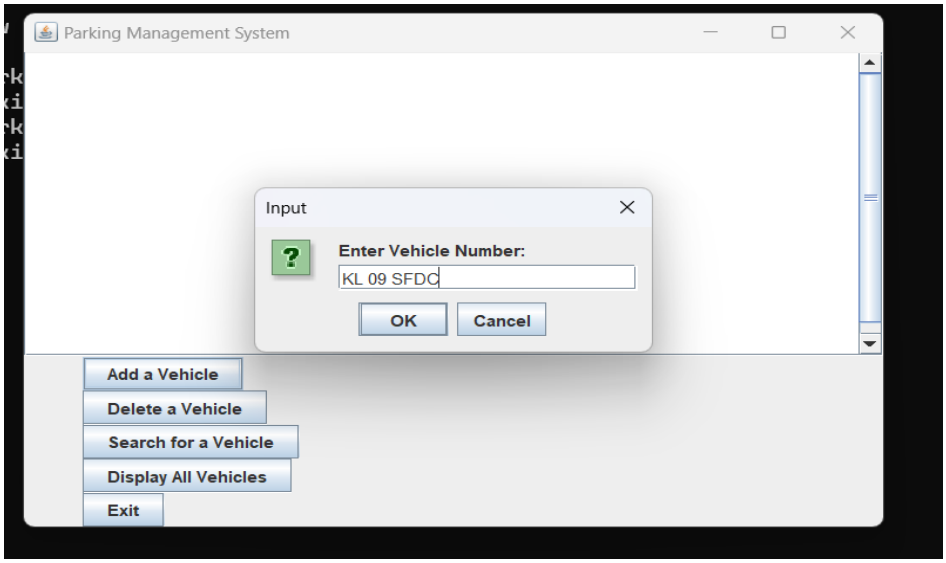


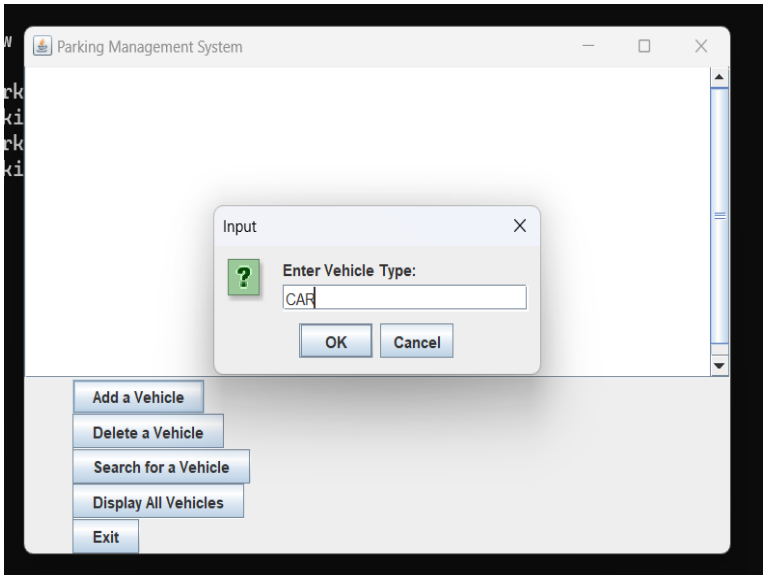
```
                    displayArea.append(str:"Vehicle details deleted successfully.\n");
                } //else {
                    // displayArea.append("Failed to delete vehicle details.\n");
                // }
            } else {
                displayArea.append(str:"Vehicle not parked here!!\n");
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    private void searchVehicle() {
        String vehicleToSearch = JOptionPane.showInputDialog(message:"Enter the Token Number of vehicle to search:");

        try (BufferedReader reader = new BufferedReader(new FileReader(fileName:"parked_vehicles.txt"))) {
            String currentLine;
            boolean vehicleFound = false;

            while ((currentLine = reader.readLine()) != null) {
                if (currentLine.contains("Token Number: " + vehicleToSearch)) {
                    displayArea.append("Vehicle details found: " + currentLine + "\n");
                    vehicleFound = true;
                    break;
                }
            }

            if (!vehicleFound) {
                displayArea.append(str:"Vehicle not found.\n");
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    private void displayAllVehicles() {
        try (BufferedReader reader = new BufferedReader(new FileReader(fileName:"parked_vehicles.txt"))) {
```

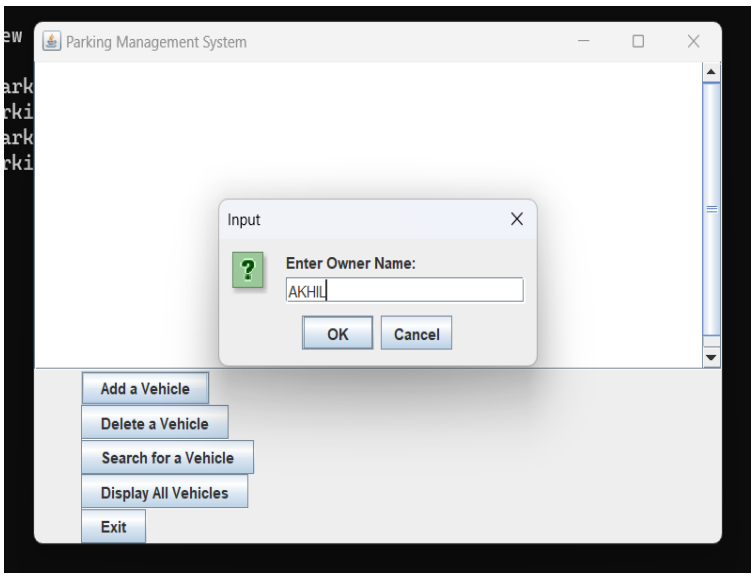
```
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
}

private void displayAllVehicles() {
    try (BufferedReader reader = new BufferedReader(new FileReader(fileName:"parked_vehicles.txt"))) {
        String currentLine;

        displayArea.append(str:"All Vehicle Details:\n");
        while ((currentLine = reader.readLine()) != null) {
            displayArea.append(currentLine + "\n");
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
}

Run | Debug
public static void main(String[] args) {
    SwingUtilities.invokeLater(new Runnable() {
        public void run() {
            new ParkingGUI().setVisible(b:true);
        }
    });
}
}
```

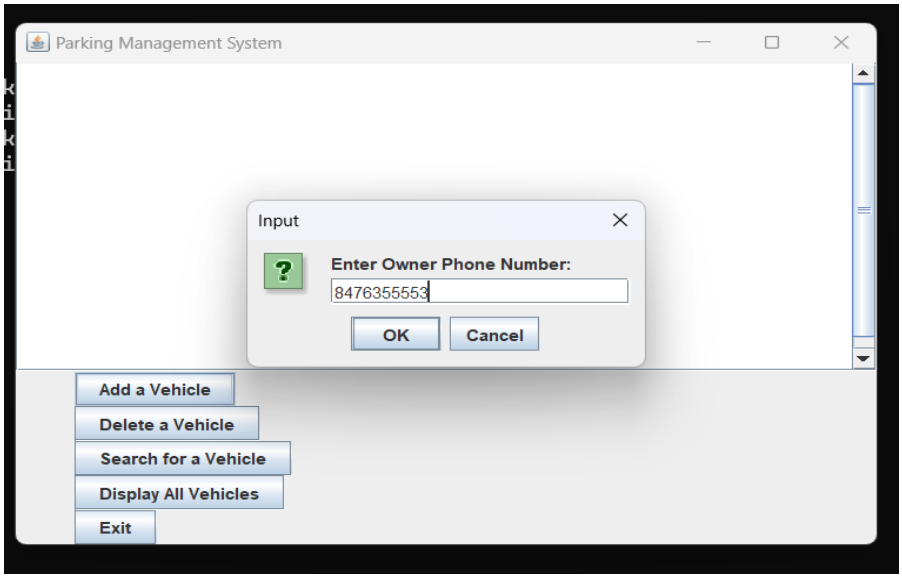
OUTPUT

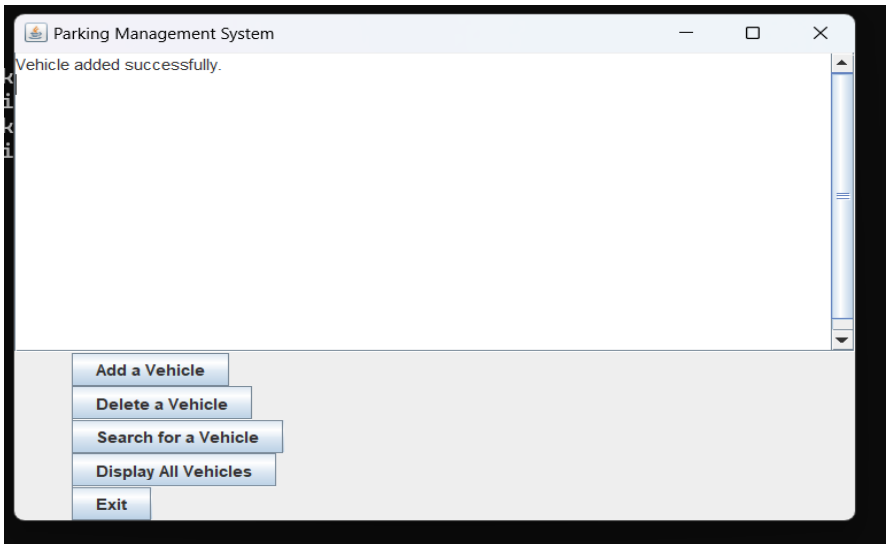


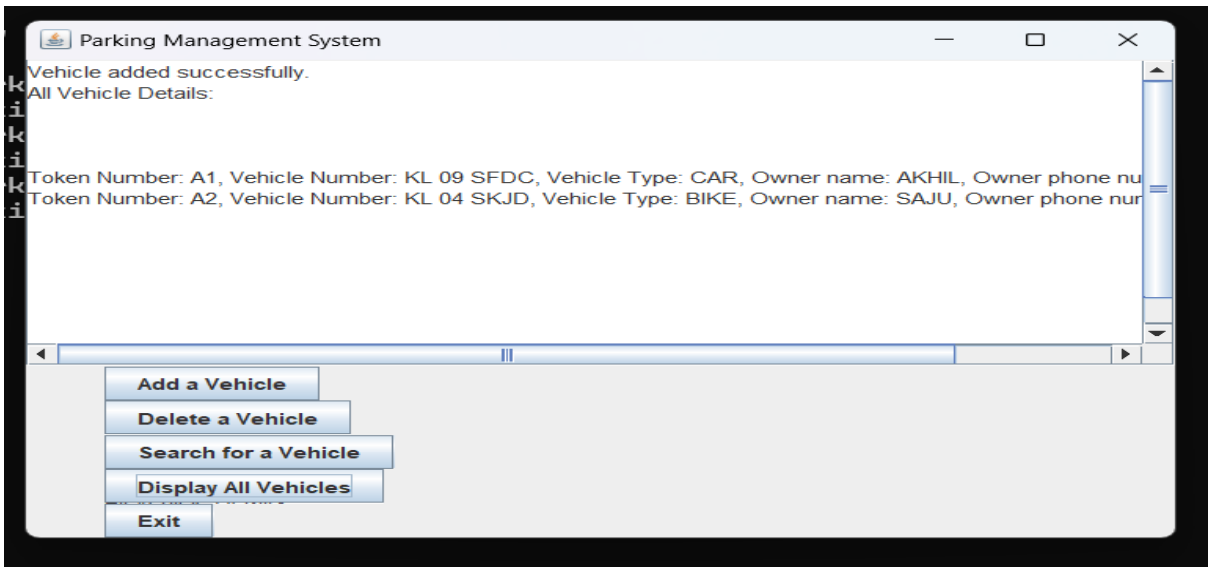

Parking Management System
Input
Enter Vehicle Number:
KL 09 SFDC
OK
Cancel
Add a Vehicle
Delete a Vehicle
Search for a Vehicle
Display All Vehicles
Exit

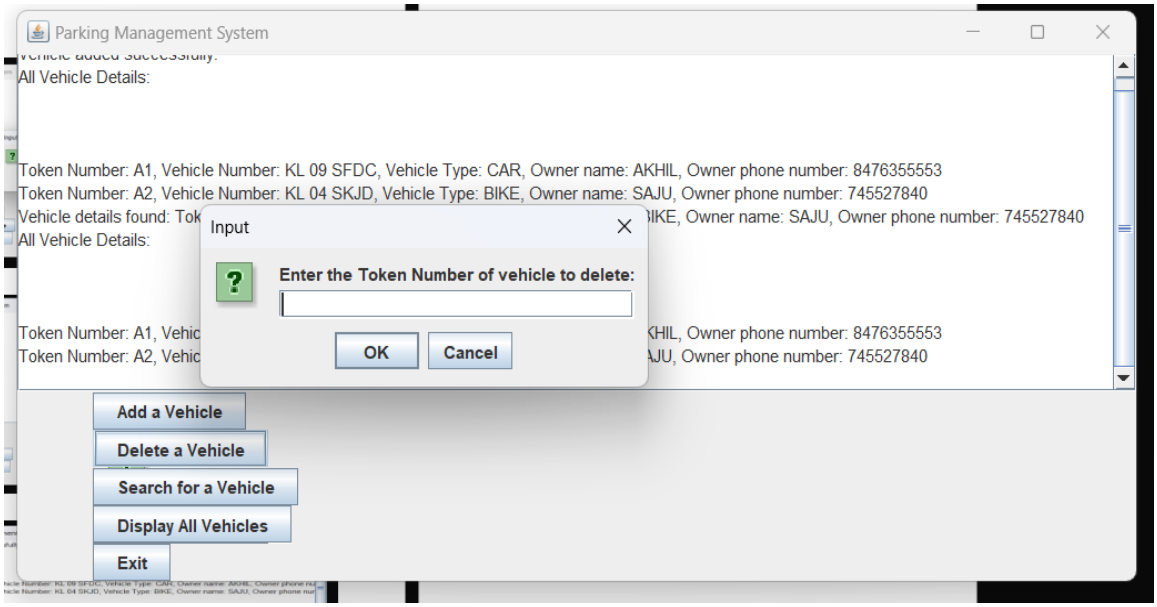

Parking Management System
Input
Enter Vehicle Type:
CAR
OK
Cancel
Add a Vehicle
Delete a Vehicle
Search for a Vehicle
Display All Vehicles
Exit

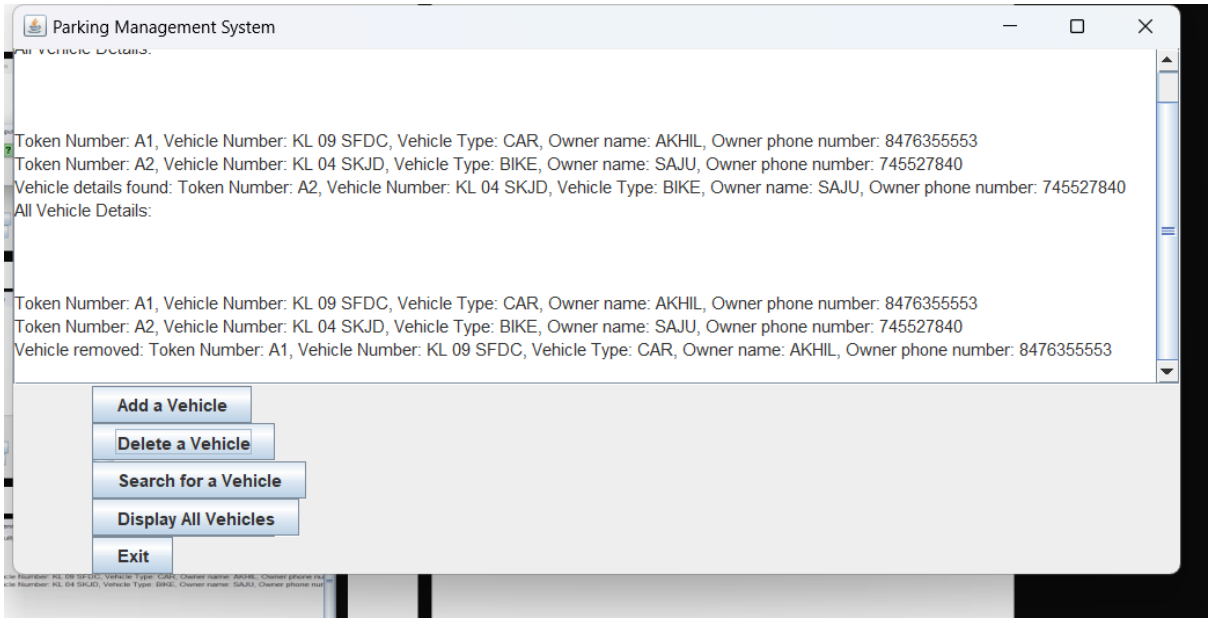

Parking Management System
Input
Enter Owner Name:
AKHIL
OK
Cancel
Add a Vehicle
Delete a Vehicle
Search for a Vehicle
Display All Vehicles
Exit


Parking Management System
Input
Enter Owner Phone Number:
8476355553
OK
Cancel
Add a Vehicle
Delete a Vehicle
Search for a Vehicle
Display All Vehicles
Exit


Parking Management System
Vehicle added successfully.
Add a Vehicle
Delete a Vehicle
Search for a Vehicle
Display All Vehicles
Exit


Parking Management System
Vehicle added successfully.
All Vehicle Details:
Token Number: A1, Vehicle Number: KL 09 SFDC, Vehicle Type: CAR, Owner name: AKHIL, Owner phone nu
Token Number: A2, Vehicle Number: KL 04 SKJD, Vehicle Type: BIKE, Owner name: SAJU, Owner phone nur
Add a Vehicle
Delete a Vehicle
Search for a Vehicle
Display All Vehicles
Exit


Parking Management System
All Vehicle Details:
Token Number: A1, Vehicle Number: KL 09 SFDC, Vehicle Type: CAR, Owner name: AKHIL, Owner phone number: 8476355553
Token Number: A2, Vehicle Number: KL 04 SKJD, Vehicle Type: BIKE, Owner name: SAJU, Owner phone number: 745527840
Vehicle details found: Tok
IKE, Owner name: SAJU, Owner phone number: 745527840
All Vehicle Details:
Token Number: A1, Vehic
KHIL, Owner phone number: 8476355553
Token Number: A2, Vehic
JU, Owner phone number: 745527840
Input
Enter the Token Number of vehicle to delete:
OK
Cancel
Add a Vehicle
Delete a Vehicle
Search for a Vehicle
Display All Vehicles
Exit


Parking Management System
Token Number: A1, Vehicle Number: KL 09 SFDC, Vehicle Type: CAR, Owner name: AKHIL, Owner phone number: 8476355553
Token Number: A2, Vehicle Number: KL 04 SKJD, Vehicle Type: BIKE, Owner name: SAJU, Owner phone number: 745527840
Vehicle details found: Token Number: A2, Vehicle Number: KL 04 SKJD, Vehicle Type: BIKE, Owner name: SAJU, Owner phone number: 745527840
All Vehicle Details:
Token Number: A1, Vehicle Number: KL 09 SFDC, Vehicle Type: CAR, Owner name: AKHIL, Owner phone number: 8476355553
Token Number: A2, Vehicle Number: KL 04 SKJD, Vehicle Type: BIKE, Owner name: SAJU, Owner phone number: 745527840
Vehicle removed: Token Number: A1, Vehicle Number: KL 09 SFDC, Vehicle Type: CAR, Owner name: AKHIL, Owner phone number: 8476355553
Add a Vehicle
Delete a Vehicle
Search for a Vehicle
Display All Vehicles
Exit

PHASE 3

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.*;

class Park {
   private static final String JDBC_URL =
"jdbc:mysql://localhost:3306/db";
   private static final String USERNAME = "root";
   private static final String PASSWORD = "adng@4490";

   void insertPark(String token, String vehtype, String vehno,
String owner, String owno) {
       try (Connection connection = getConnection();
            PreparedStatement preparedStatement =
connection.prepareStatement(
                   "INSERT INTO parkn(token, vehtype, vehno,
owner, owno) VALUES (?, ?, ?, ?, ?)")) {

           preparedStatement.setString(1, token);
           preparedStatement.setString(2, vehtype);
           preparedStatement.setString(3, vehno);
           preparedStatement.setString(4, owner);
           preparedStatement.setString(5, owno);

           int rowsAffected =
preparedStatement.executeUpdate();
           if (rowsAffected > 0) {
```

```
            System.out.println("Park details inserted
successfully.");
          } else {
            System.out.println("Failed to insert token
details.");
          }

       } catch (SQLException e) {
          e.printStackTrace();
       }
    }

    void show() {
       try (Connection connection = getConnection();
            Statement statement =
connection.createStatement();
            ResultSet resultSet =
statement.executeQuery("SELECT * FROM parkn")) {

          while (resultSet.next()) {
             String n = resultSet.getString("token");
             String i = resultSet.getString("vehno");
             String p = resultSet.getString("vehtype");
             String s = resultSet.getString("owner");
             String a = resultSet.getString("owno");

             System.out.println();
             System.out.println("Name:  " + n);
             System.out.println("Vehcle no: " + i);
             System.out.println("vehtype: " + p);
             System.out.println("owner: " + s);
```

```
            System.out.println("Owner phone no: " + a);
          }

      } catch (SQLException e) {
          e.printStackTrace();
      }
   }

   void showowner(String owner) {
      try (Connection connection = getConnection();
           PreparedStatement preparedStatement =
connection.prepareStatement(
                 "SELECT * FROM parkn WHERE owner = ?")) {

          preparedStatement.setString(1, owner);
          ResultSet resultSet =
preparedStatement.executeQuery();

          while (resultSet.next()) {
             String n = resultSet.getString("token");
             String i = resultSet.getString("vehno");
             String p = resultSet.getString("vehtype");
             String s = resultSet.getString("owner");
             String a = resultSet.getString("owno");

             System.out.println();
             System.out.println("Name:  " + n);
             System.out.println("Vehicle no: " + i);
             System.out.println("Vehicle type: " + p);
             System.out.println("owner: " + s);
             System.out.println("Owner phone no: " + a);
```

```
            }

        } catch (SQLException e) {
            e.printStackTrace();
        }
    }

    void deleteByvehno(String vehno) {
        try (Connection connection = getConnection();
             PreparedStatement preparedStatement =
connection.prepareStatement(
                     "DELETE FROM parkn WHERE vehno = ?")) {

            preparedStatement.setString(1, vehno);
            int rowsAffected =
preparedStatement.executeUpdate();

            if (rowsAffected > 0) {
                System.out.println("Park with ID " + vehno + "
deleted successfully.");
            } else {
                System.out.println("No token found with ID " +
vehno + ".");
            }

        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
```

```
    private static Connection getConnection() throws
SQLException {
        return DriverManager.getConnection(JDBC_URL,
USERNAME, PASSWORD);
    }
}

class ParkGUI extends JFrame {
    private Park token = new Park();

    private JTextField tokenField;
    private JTextField vehtypeField;
    private JTextField vehnoField;
    private JTextField ownerField;
    private JTextField ownoField;

    public ParkGUI() {
        setTitle("Parking Area");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(500, 400);
        setLocationRelativeTo(null);

        createGUIComponents();

        setVisible(true);
    }

    private void createGUIComponents() {
        JPanel panel = new JPanel();
        panel.setLayout(new GridLayout(11, 2));
```

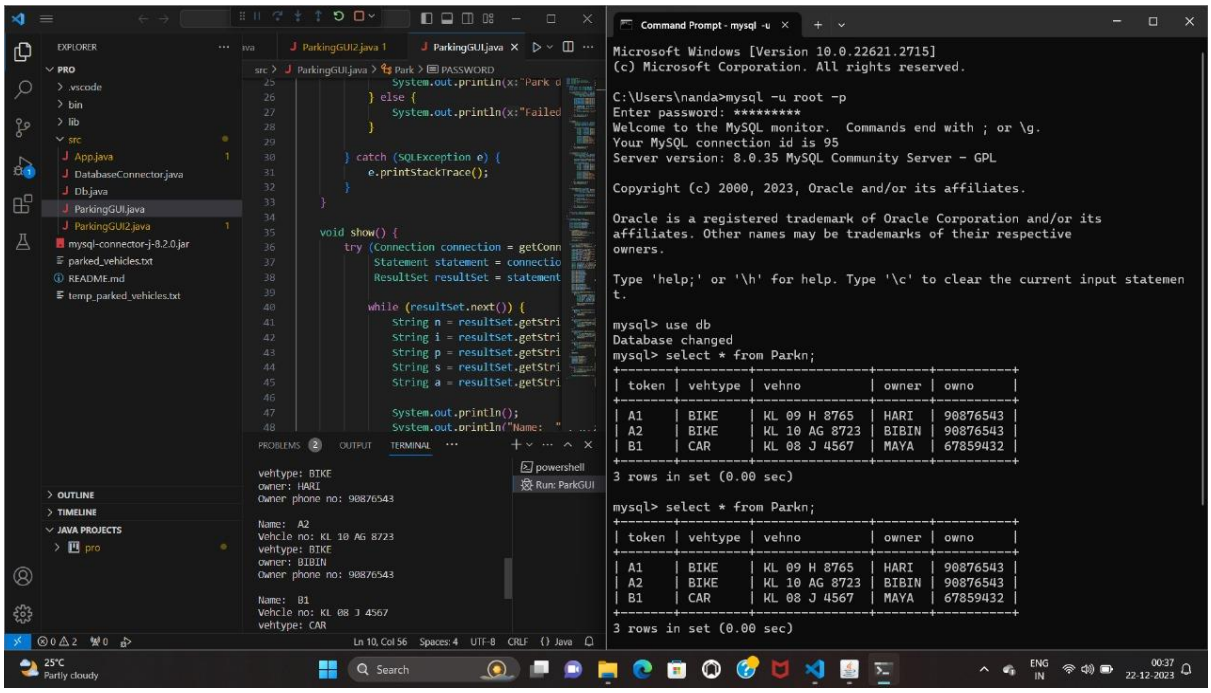
```
        JLabel tokenLabel = new JLabel("token:");
        tokenField = new JTextField();
        JLabel vehtypeLabel = new JLabel("vehtype:");
        vehtypeField = new JTextField();
        JLabel vehnoLabel = new JLabel("vehno:");
        vehnoField = new JTextField();
        JLabel ownerLabel = new JLabel("owner:");
        ownerField = new JTextField();
        JLabel ownoLabel = new JLabel("owno:");
        ownoField = new JTextField();

        JButton insertButton = new JButton("Insert Park
Details");
        JButton viewAllButton = new JButton("View All");
        JButton ownerButton = new JButton("Search by
owner");
        JButton deleteByvehnoButton = new JButton("Delete by
vehicle no");
        JButton exitButton = new JButton("Exit");

        panel.add(tokenLabel);
        panel.add(tokenField);
        panel.add(vehtypeLabel);
        panel.add(vehtypeField);
        panel.add(vehnoLabel);
        panel.add(vehnoField);
        panel.add(ownerLabel);
        panel.add(ownerField);
        panel.add(ownoLabel);
        panel.add(ownoField);
```

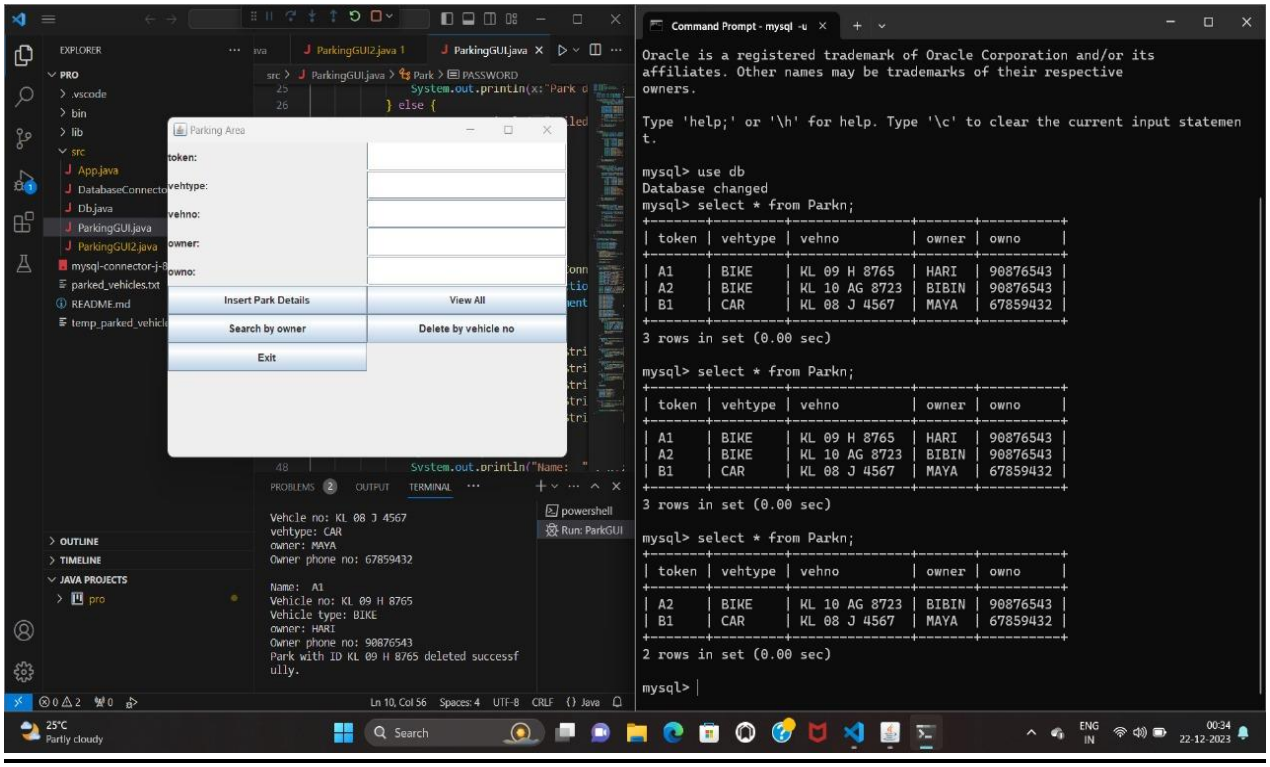
```
        panel.add(insertButton);
        panel.add(viewAllButton);
        panel.add(ownerButton);
        panel.add(deleteByvehnoButton);
        panel.add(exitButton);

        insertButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                String inputToken = tokenField.getText();
                String vehtype = vehtypeField.getText();
                String vehno = vehnoField.getText();
                String owner = ownerField.getText();
                String owno = ownoField.getText();

                token.insertPark(inputToken, vehtype, vehno,
owner, owno);
                clearFields();
            }
        });

        viewAllButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                token.show();
                clearFields();
            }
        });

        ownerButton.addActionListener(new ActionListener() {
            @Override
```

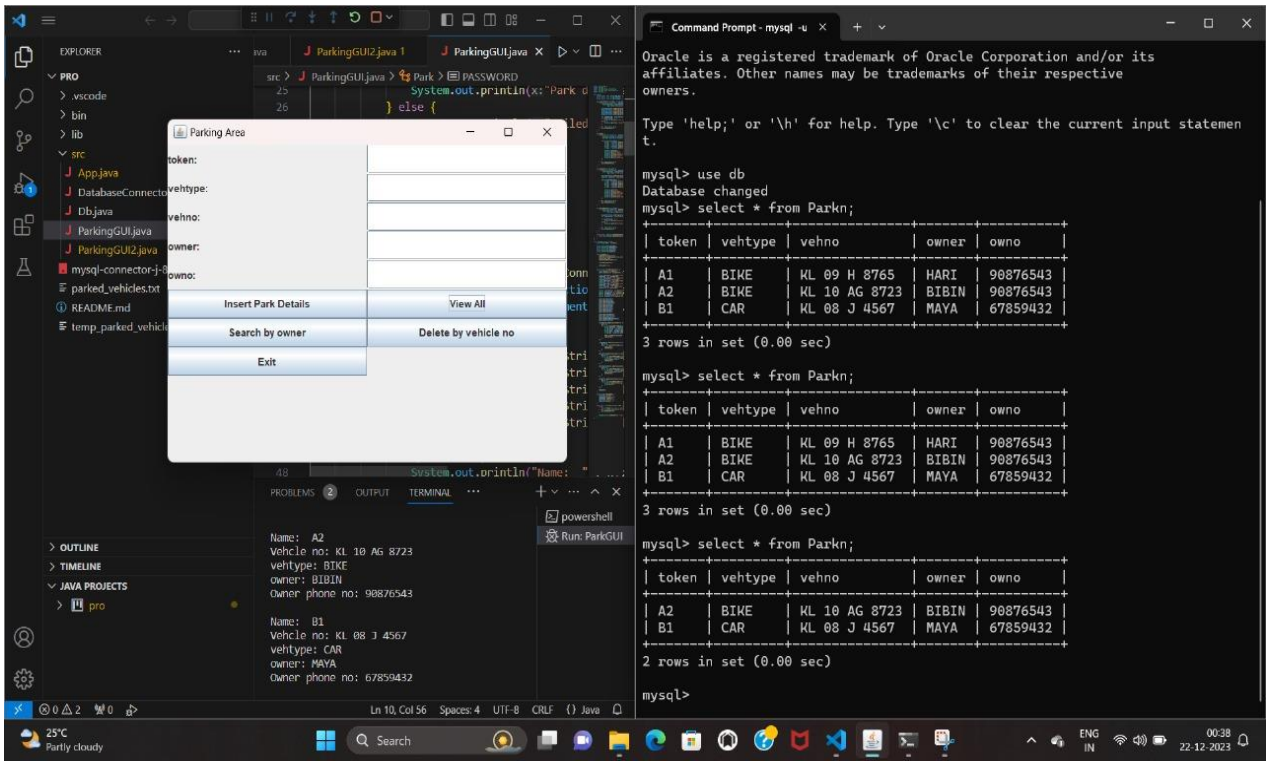
```
            public void actionPerformed(ActionEvent e) {
                String owner = ownerField.getText();
                token.showowner(owner);
                clearFields();
            }
        });

        deleteByvehnoButton.addActionListener(new
ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                String vehno = vehnoField.getText();
                token.deleteByvehno(vehno);
                clearFields();
            }
        });

        exitButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                System.out.println("Exiting.....");
                System.exit(0);
            }
        });

        add(panel);
    }

    private void clearFields() {
        tokenField.setText("");
        vehtypeField.setText("");
```

```
        vehnoField.setText("");
        ownerField.setText("");
        ownoField.setText("");
    }

    public static void main(String[] args) {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
        } catch (ClassNotFoundException e) {
            e.printStackTrace();
        }

        SwingUtilities.invokeLater(() -> new ParkGUI());
    }
}
```

OUTPUT




Parking Area
token:
vehtype:
vehno:
owner:
owno:
Insert Park Details
View All
Search by owner
Delete by vehicle no
Exit
Vehcle no: KL 08 J 4567
vehtype: CAR
owner: MAYA
Owner phone no: 67859432
Name: A1
Vehcle no: KL 09 H 8765
Vehicle type: BIKE
owner: HARI
Owner phone no: 90876543
Park with ID KL 09 H 8765 deleted successf
ully.
Command Prompt - mysql -u
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statemen
t.
mysql> use db
Database changed
mysql> select * from Parkn;
token	vehtype	vehno	owner	owno
A1	BIKE	KL 09 H 8765	HARI	90876543
A2	BIKE	KL 10 AG 8723	BIBIN	90876543
B1	CAR	KL 08 J 4567	MAYA	67859432
3 rows in set (0.00 sec)
mysql> select * from Parkn;
token	vehtype	vehno	owner	owno
A1	BIKE	KL 09 H 8765	HARI	90876543
A2	BIKE	KL 10 AG 8723	BIBIN	90876543
B1	CAR	KL 08 J 4567	MAYA	67859432
3 rows in set (0.00 sec)
mysql> select * from Parkn;
token	vehtype	vehno	owner	owno
A2	BIKE	KL 10 AG 8723	BIBIN	90876543
B1	CAR	KL 08 J 4567	MAYA	67859432
2 rows in set (0.00 sec)
mysql>


Parking Area
token:
vehtype:
vehno:
owner:
owno:
Insert Park Details
View All
Search by owner
Delete by vehicle no
Exit
Name: A2
Vehcle no: KL 10 AG 8723
vehtype: BIKE
owner: BIBIN
Owner phone no: 90876543
Name: B1
Vehcle no: KL 08 J 4567
vehtype: CAR
owner: MAYA
Owner phone no: 67859432
Command Prompt - mysql -u
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statemen
t.
mysql> use db
Database changed
mysql> select * from Parkn;
token	vehtype	vehno	owner	owno
A1	BIKE	KL 09 H 8765	HARI	90876543
A2	BIKE	KL 10 AG 8723	BIBIN	90876543
B1	CAR	KL 08 J 4567	MAYA	67859432
3 rows in set (0.00 sec)
mysql> select * from Parkn;
token	vehtype	vehno	owner	owno
A1	BIKE	KL 09 H 8765	HARI	90876543
A2	BIKE	KL 10 AG 8723	BIBIN	90876543
B1	CAR	KL 08 J 4567	MAYA	67859432
3 rows in set (0.00 sec)
mysql> select * from Parkn;
token	vehtype	vehno	owner	owno
A2	BIKE	KL 10 AG 8723	BIBIN	90876543
B1	CAR	KL 08 J 4567	MAYA	67859432
2 rows in set (0.00 sec)
mysql>