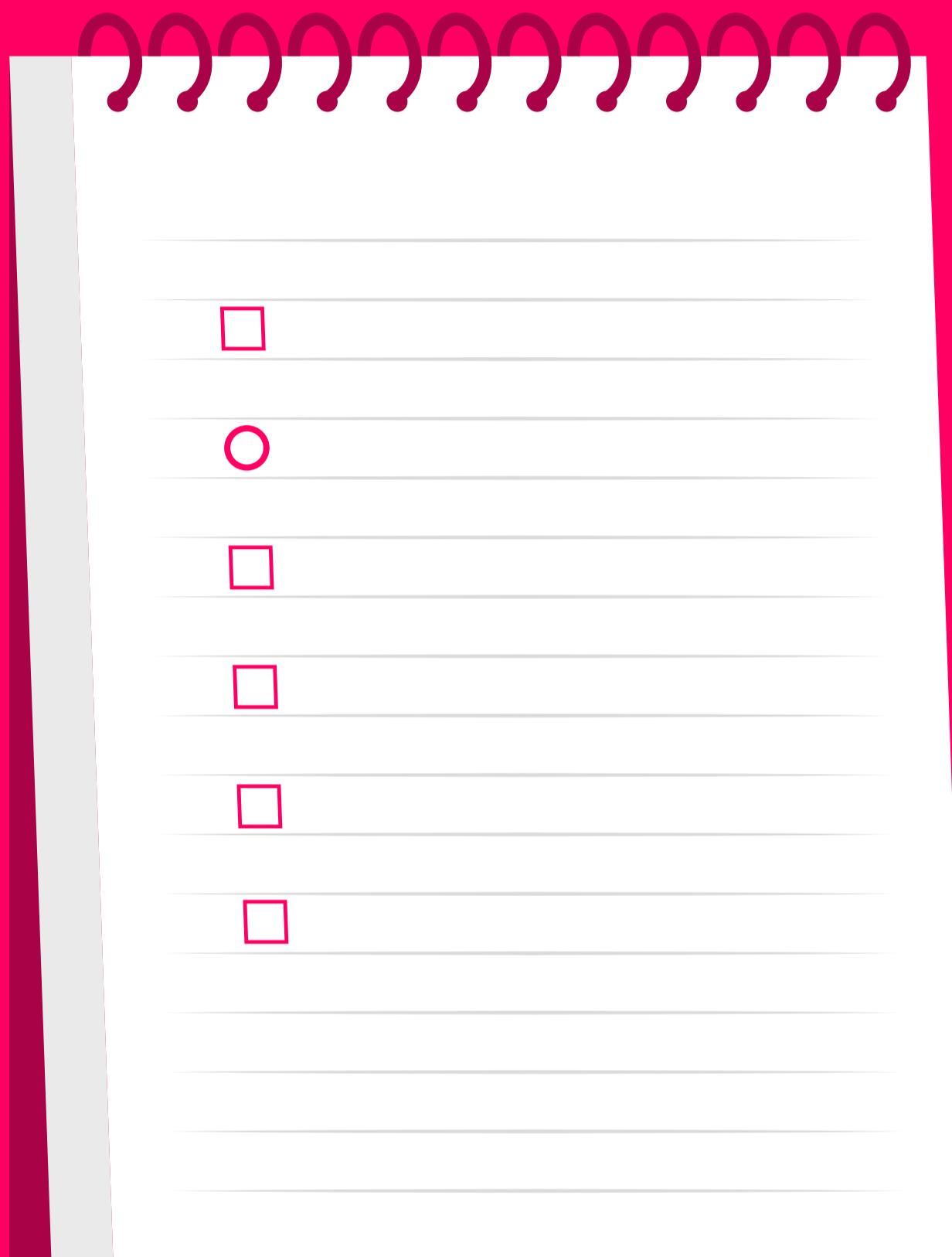




El futuro digital
es de todos

MinTIC



Función Modificar y Listar con JDBC

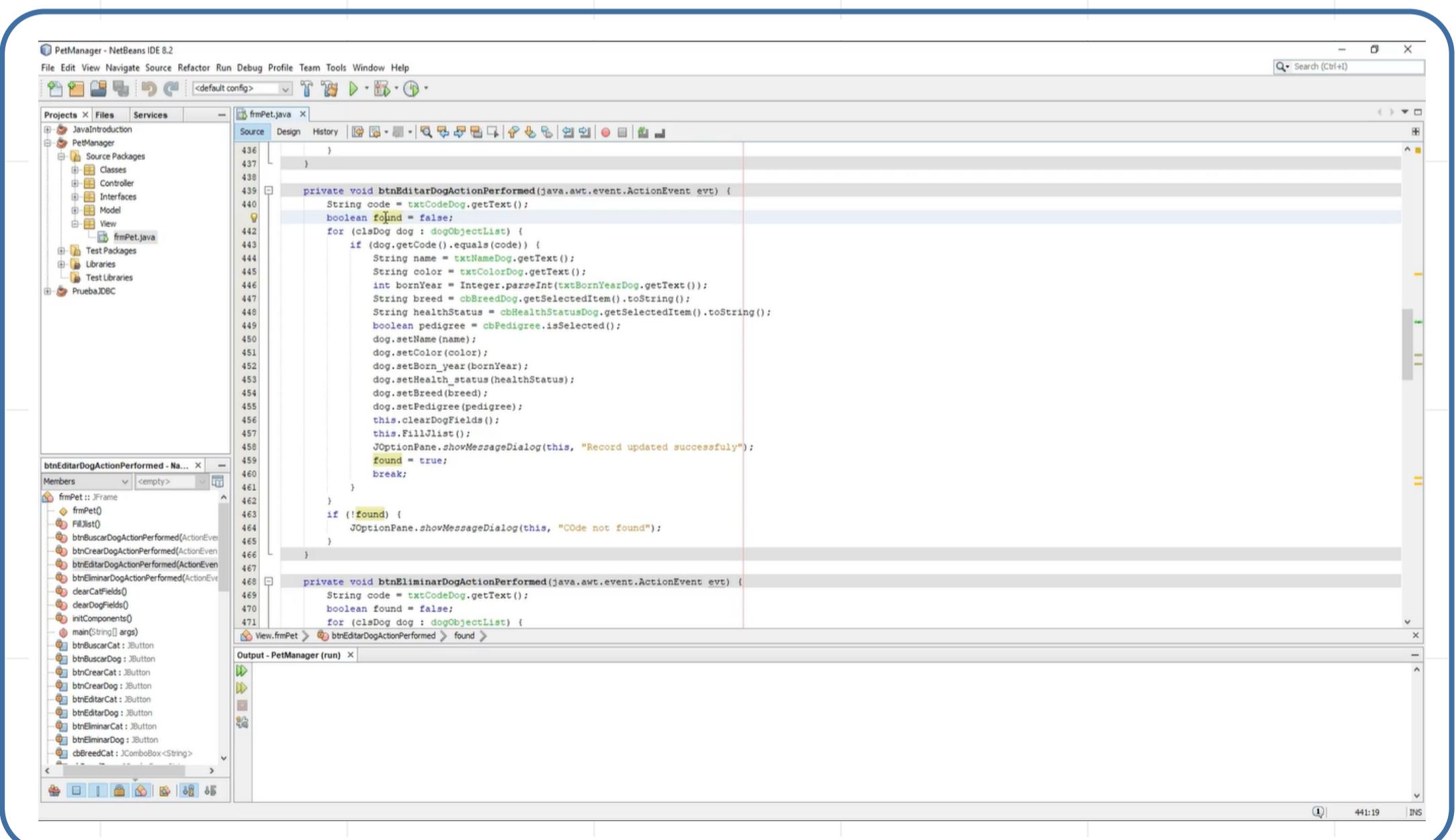


Universidad de Caldas

Hola:

Hemos visto cómo configurar el **JDBC** para conectarnos y gestionar la base de datos en Mysql con la funcionalidad de creación y búsqueda de registros. Ahora integraremos las demás operaciones del **CRUD** en la capa de modelo con el apoyo del **JDBC** para así completar la gestión de mascotas en nuestra aplicación, editar la información y cómo listar los datos que se encuentran almacenados.

Video de implementación de las funcionalidades de edición y listado de información.



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

- Project Explorer:** Shows the **PetManager** project structure, including JavaIntroduction, PetManager (Source Packages: Classes, Controller, Interfaces, Model, View), Test Packages, Libraries, and PruebaJDBC.
- Code Editor:** The **frmPet.java** file is open, displaying Java code for button event handlers. The code implements functionality for updating dog records based on a search code and deleting dog records based on a search code.
- Output:** The Output window shows the message "PetManager (run)".
- Status Bar:** The status bar at the bottom right indicates "441:19 INS".

PetManager - NetBeans IDE 8.2

```

private void btnEditarDogActionPerformed(java.awt.event.ActionEvent evt) {
    String code = txtCodeDog.getText();
    String name = txtNameDog.getText();
    String color = txtColorDog.getText();
    int bornYear = Integer.parseInt(txtBornYearDog.getText());
    String breed = cbBreedDog.getSelectedItem().toString();
    String healthStatus = cbHealthStatusDog.getSelectedItem().toString();
    boolean pedigree = cbPedigree.isSelected();
    try {
        int petId = Integer.parseInt(txtPetId.getText());
        int dogId = Integer.parseInt(txtDogId.getText());
        clsDog dog = new clsDog(dogId, breed, pedigree, petId, code, name, bornYear, color, healthStatus);
        boolean response = ctlPet.EditPet(dog);
        if(response)
            JOptionPane.showMessageDialog(this, "Record updated");
        else
            JOptionPane.showMessageDialog(this, "Error updating the record");
    } catch (Exception e) {
        JOptionPane.showMessageDialog(this, "You have to search the record to update");
    }
}

private void btnEliminarDogActionPerformed(java.awt.event.ActionEvent evt) {
    String code = txtCodeDog.getText();
    boolean found = false;
    for (clsDog dog : dogObjectList) {
        if (dog.getCode().equals(code)) {
            this.dogObjectList.remove(dog);
            this.clearDogFields();
            this.FillList();
            JOptionPane.showMessageDialog(this, "Record removed successfully");
            found = true;
        }
    }
}

```

PetManager - NetBeans IDE 8.2

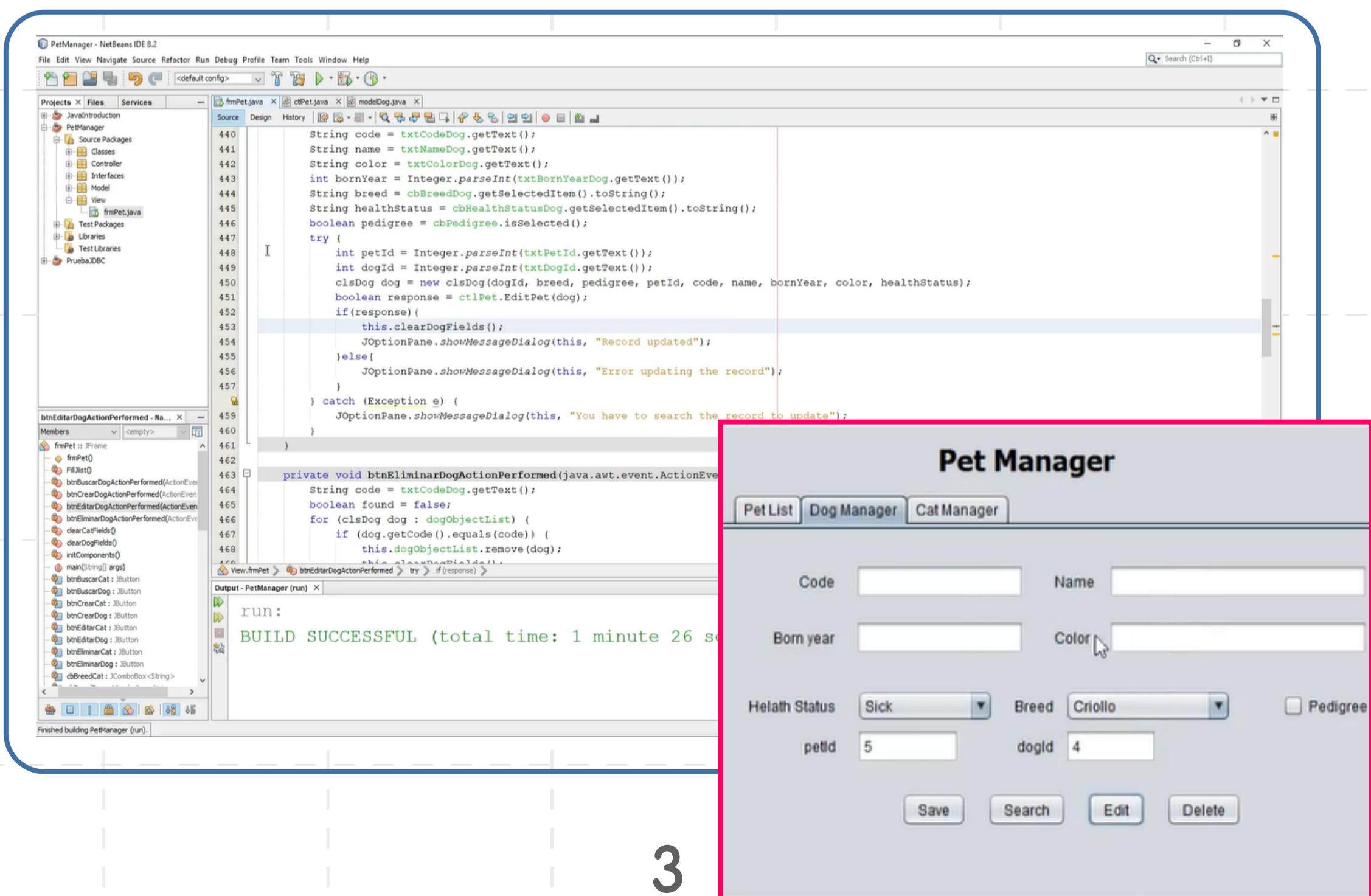
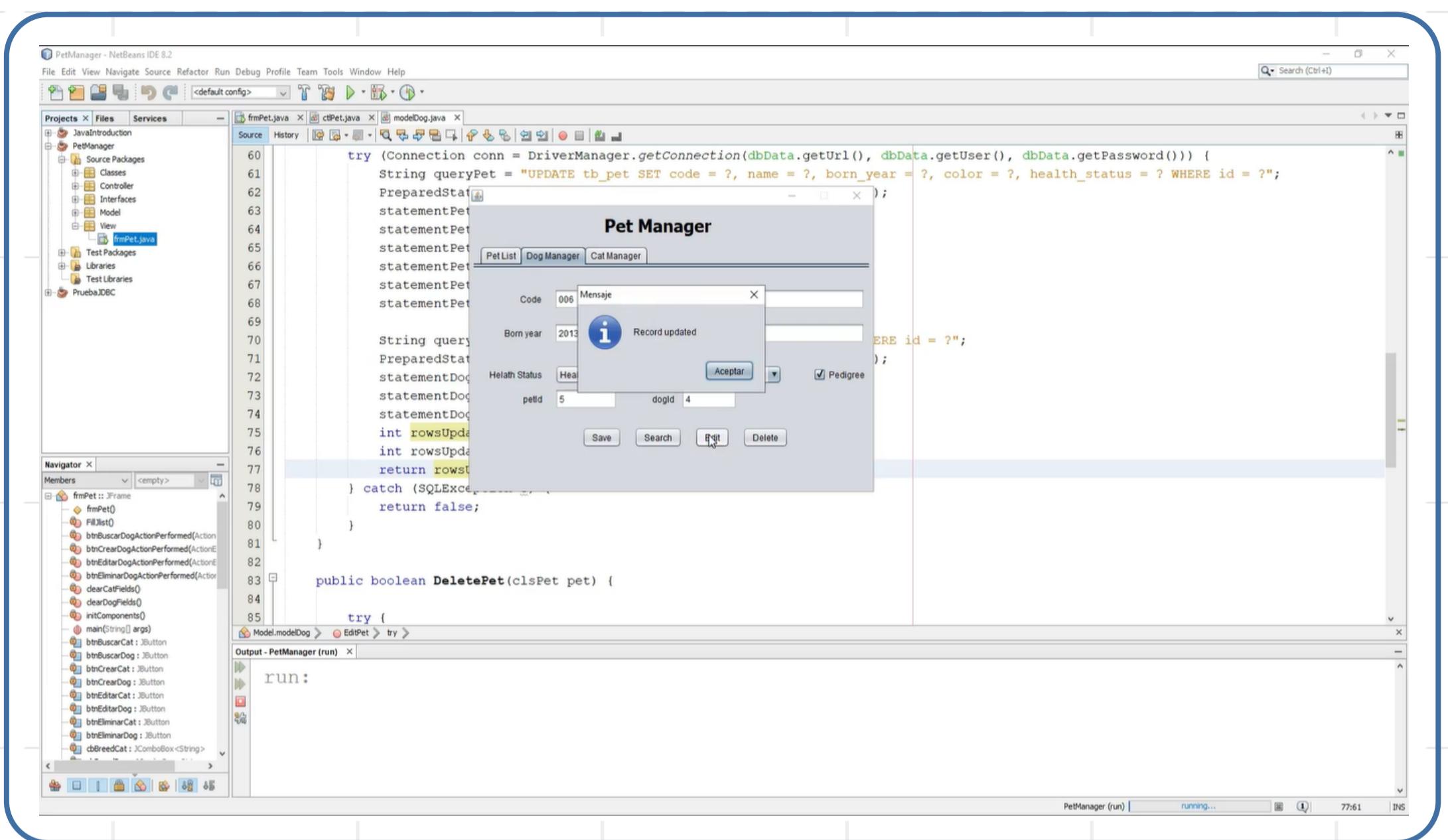
```

try (Connection conn = DriverManager.getConnection(dbData.getUrl(), dbData.getUser(), dbData.getPassword())) {
    String queryPet = "UPDATE tb_pet SET code = ?, name = ?, born_year = ?, color = ?, health_status = ? WHERE id = ?";
    PreparedStatement statementPet = conn.prepareStatement(queryPet);
    statementPet.setString(1, dog.getCode());
    statementPet.setString(2, dog.getName());
    statementPet.setInt(3, dog.getBorn_year());
    statementPet.setString(4, dog.getColor());
    statementPet.setString(5, dog.getHealth_status());
    statementPet.setInt(6, dog.getPetId());

    String queryDog = "UPDATE tb_dog SET breed = ?, pedigree = ? WHERE id = ?";
    PreparedStatement statementDog = conn.prepareStatement(queryDog);
    statementDog.setString(1, dog.getBreed());
    statementDog.setBoolean(2, dog.isPedigree());
    statementDog.setInt(3, dog.getDogId());
    int rowsUpdatedPet = statementPet.executeUpdate();
    int rowsUpdatedDog = statementDog.executeUpdate();
    return rowsUpdatedPet > 0 && rowsUpdatedDog > 0;
} catch (SQLException e) {
    return false;
}

public boolean DeletePet(clsPet pet) {
    try {

```



3

PetManager - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Projects Files Services

frmPet.java modelDog.java

Source History

117
118
119 public LinkedList<clsPet> SearchPet() {
120 LinkedList<clsPet> dogList = new LinkedList<>();
121 try (Connection conn = DriverManager.getConnection(dbData.getUrl(), dbData.getUser(), dbData.getPassword())) {
122 String query = "SELECT * FROM tb_pet AS p INNER JOIN tb_dog AS d ON p.id = d.id_pet";
123 PreparedStatement statementPet = conn.prepareStatement(query);
124 ResultSet result = statementPet.executeQuery();
125 while (result.next()) {
126 int petId = result.getInt(1);
127 String petCode = result.getString(2);
128 String petName = result.getString(3);
129 int petBornYear = result.getInt(4);
130 String petColor = result.getString(5);
131 String petHealthStatus = result.getString(6);
132 int dogId = result.getInt(7);
133 String petBreed = result.getString(8);
134 Boolean petPedigree = result.getBoolean(9);
135 clsDog dog = new clsDog(dogId, petBreed, petPedigree, petId, petCode, petName, petBornYear, petColor, petHealthStatus);
136 dogList.add(dog);
137 }
138 } catch (Exception e) {
139 return dogList;
140 }
141 }
142 }
143 }

SearchPet - Navigator

Members <empty>

modelDog

- modelDog0
- CreatePet(dsDog dog) : boolean
- DeletePet(dsPet pet) : boolean
- EditPet(dsPet dog) : boolean
- SearchPet(String code) : dsPet
- SearchPet() : LinkedList<dsPet>

dbData : dbData

Output - PetManager (run) x

run:
BUILD SUCCESSFUL (total time: 1 minute 6 seconds)

modelDog.java saved.

PetManager - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Projects Files Services

frmPet.java modelDog.java ctPet.java

Source History

84
85
86
87
88
89
90
91
92 public LinkedList<clsPet> ListPet(String code, String type) {
93 LinkedList<clsPet> petList = null;
94 try {
95 switch (type) {
96 case "Cat":
97 break;
98 case "Dog":
99 petList = this.modelDog.ListPet();
100 break;
101 }
102 } catch (Exception e) {
103 return petList;
104 } catch (Exception e) {
105 return null;
106 }
107 }
108 }

ctPet - Navigator

Members <empty>

ctPet

- ctPet0
- CreatePet(dsPet pet) : boolean
- DeletePet(dsPet pet) : boolean
- EditPet(dsPet pet) : boolean
- ListPet(String code, String type) : LinkedList<clsPet>
- SearchPet(String code, String type) : dsPet

modelCat : modelCat

modelDog : modelDog

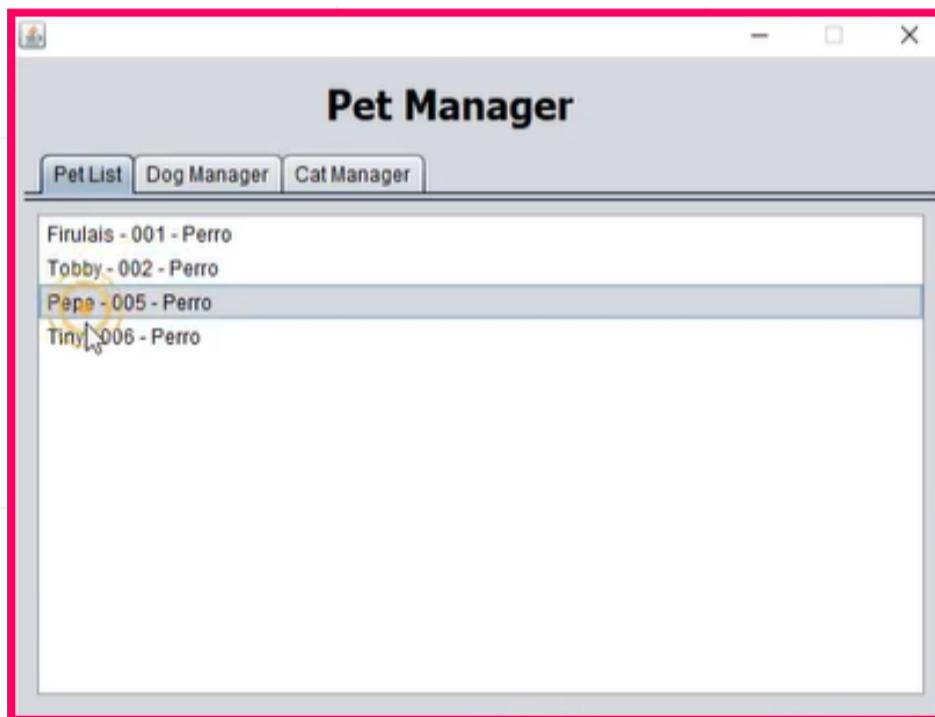
Output - PetManager (run) x

run:
BUILD SUCCESSFUL (total time: 1 minute 6 seconds)

ctPet.java saved.

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

- Project Explorer:** Shows the PetManager project structure with packages like JavaIntroduction, PetManager, and PruebaJDBC.
- Code Editor:** Displays the `frmPet.java` file containing Java code for a pet manager application. A specific section of the code is highlighted, showing a loop that fills a list with dog objects. The highlighted line is: `model.add(index, data);`
- Output Window:** Shows the build log with the message "BUILD SUCCESSFUL (total time: 1 minute 6 seconds)".



Una vez aprendido cómo realizar las funciones de edición y listado en una base de datos Mysql con el apoyo del **JDBC** para implementarlo desde Java, los invitamos a practicar con la solución del reto de la semana y a profundizar en esta temática con el material adicional.



**Mision
TIC2022**



Universidad de Caldas