Звіт

Автор: Водолазський Микола Анатолійович

КІТ-118а

Лабораторна робота №16

Розробка графічного інтерфейсу користувача

Мета:

- Розширення функціональності параметризованих класів.

Вимоги

Розробити графічний інтерфейс користувача для програми
рішення попередньої лабораторної роботи з використанням засобів JavaFX.

-

ПРИКЛАДНА ЗАДАЧА:

Кадрове агентство. Сортування за назвою фірми, за назвою запропонованої спеціальності, за вказаною освітою.

ОПИС ПРОГРАМИ

2.1 Опис змінних:

LinkedContainer<SecondCreate> stringLinked = new LinkedContainer<>();// об’єкт параметризованого контейнера

Scanner scan = new Scanner(System.in); // змінна для активування зчитування з консолі

2.2 Ієрархія та структура класів.

Main class – головний клас. Містить метод main(точку входу у програму) та методи по роботі з програмою для реалізації індивідуального завдання.

interface iLinked - інтерфейс контенеру

class SecondCreate - клас прикладної задачі кадрового агенства

class linkedContainer - параметризований клас-контейнер, котрий зберігає інформацію агенства

ТЕКСТ ПРОГРАМИ

Main.java

```
package sample;

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

public class Main extends Application {

    @Override
    public void start(Stage primaryStage) throws Exception{
        Parent root = FXMLLoader.load(getClass().getResource("sample.fxml"));
        primaryStage.setTitle("Recruitment agency");
        primaryStage.setScene(new Scene(root, 1280, 720));
        primaryStage.show();
    }


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

FindVcController.java

```
package sample;

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

public class Main extends Application {

    @Override
    public void start(Stage primaryStage) throws Exception{
        Parent root = FXMLLoader.load(getClass().getResource("sample.fxml"));
        primaryStage.setTitle("Recruitment agency");
        primaryStage.setScene(new Scene(root, 1280, 720));
        primaryStage.show();
    }


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

ListController.java

```
package sample;
```

```java
import java.net.URL;
import java.util.ResourceBundle;

import javafx.beans.property.SimpleStringProperty;
import javafx.beans.value.ObservableValue;
import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
import javafx.collections.transformation.FilteredList;
import javafx.fxml.FXML;
import javafx.fxml.Initializable;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.control.TableColumn;
import javafx.scene.control.TableView;
import javafx.scene.control.TextField;
import javafx.scene.control.cell.TextFieldTableCell;

public class listController {

    @FXML
    private ResourceBundle resources;

    @FXML
    private URL location;

    @FXML
    private TableView<SecondCreate> TableView;

    @FXML
    private TableColumn<SecondCreate, String> company;

    @FXML
    private TableColumn<SecondCreate, String> specialisation;

    @FXML
    private TableColumn<SecondCreate, String> workingConditions;

    @FXML
    private TableColumn<SecondCreate, String> payment;

    @FXML
    private TableColumn<SecondCreate, String> workingExperience;

    @FXML
    private TableColumn<SecondCreate, String> education;

    @FXML
    private TableColumn<SecondCreate, String> English;

    @FXML
    private TableColumn<SecondCreate, String> License;

    @FXML
    private Button DeleteButton;

    @FXML
    private Button EditButton;

    @FXML
    private Button AddButton;

    @FXML
```

```
    private TextField SearchField;

    @FXML
    private Button FindButton;

    @FXML
    private TextField companyButton;

    @FXML
    private TextField workingExpButton;

    @FXML
    private TextField paymentButton;

    @FXML
    private TextField conditionsButton;

    @FXML
    private TextField specialisationButton;

    @FXML
    private TextField LicenseButton;

    @FXML
    private TextField educationButton;

    @FXML
    private TextField EnglishButton;

    @FXML
    private Label joobleButton;

    @FXML
    private Button FindVacancyButton;

    @FXML
    private Button PlaceVacancyButton;

    @FXML
    private void add() {
        items.add(new SecondCreate(companyButton.getText(),
specialisationButton.getText(), conditionsButton.getText(), paymentButton.getText(),
                workingExpButton.getText(), educationButton.getText(),
EnglishButton.getText(), LicenseButton.getText()));
        companyButton.clear();
        specialisationButton.clear();
        conditionsButton.clear();
        paymentButton.clear();
        workingExpButton.clear();
        educationButton.clear();
        EnglishButton.clear();
        LicenseButton.clear();
    }

   /* @FXML
    private void SearchBtn()
    {
        *//*PlaceVacancyButton.setOnAction(event -> {
            searchField();
        });*//*
        searchField();
```

```
}*/


private FilteredList<SecondCreate> filterdata;

@FXML
public void SearchBtn()
{

    SearchField.textProperty().addListener((o, ov, nv) -> {
        filterdata.setPredicate((SecondCreate s) -> {
            String newVal=nv.toLowerCase();

            return s.getCompany().toLowerCase().contains(newVal);

            //return false;
        });
    });

    TableView.setItems(filterdata);
}




@FXML
private void delete()
{
    ObservableList<SecondCreate> selectedRows, allPeople;

    allPeople = TableView.getItems();

    //this gives us the rows that were selected
    selectedRows = TableView.getSelectionModel().getSelectedItems();

    //loop over the selected rows and remove the Person objects from the table
    for (SecondCreate person: selectedRows)
    {
        allPeople.remove(person);
    }

}

@FXML
private void edit()
{
    TableView.refresh();
}

public void editCompany(TableColumn.CellEditEvent edittedCell)
{
SecondCreate person = TableView.getSelectionModel().getSelectedItem();
person.setCompany(edittedCell.getNewValue().toString());
}



ObservableList<SecondCreate> items = FXCollections.observableArrayList();

@FXML
void initialize() {
```

```
        filterdata = new FilteredList<SecondCreate>(items ,e->true);
    TableView.itemsProperty().setValue(items);

    company.setCellValueFactory(param-> new
SimpleStringProperty(param.getValue().getCompany()));
        specialisation.setCellValueFactory(param-> new
SimpleStringProperty(param.getValue().getSpecialisation()));
        workingConditions.setCellValueFactory(param-> new
SimpleStringProperty(param.getValue().getWorkingConditions()));
        payment.setCellValueFactory(param-> new
SimpleStringProperty(param.getValue().getPayment()));
        workingExperience.setCellValueFactory(param-> new
SimpleStringProperty(param.getValue().getWorkingExperience()));
        education.setCellValueFactory(param-> new
SimpleStringProperty(param.getValue().getEducation()));
        English.setCellValueFactory(param-> new
SimpleStringProperty(param.getValue().getEnglish()));
        License.setCellValueFactory(param-> new
SimpleStringProperty(param.getValue().getLicence()));

        TableView.setEditable(true);
        company.setCellFactory(TextFieldTableCell.forTableColumn());

    }


}
```

SecondCreate.java

```
package sample;

import javafx.beans.property.SimpleStringProperty;

import java.io.Serializable;


public class SecondCreate {//implements Comparable<SecondCreate>, Serializable {


    public SecondCreate()
    {

    }

    private String company;
    private String specialisation;
    private String workingConditions;
    private String payment;
    private String workingExperience;
    private String education;
    private String Licence;
    private String English;

    public SecondCreate(String company, String specialisation, String
workingConditions, String payment, String workingExperience, String education, String
Licence, String English) {
        this.company = company;
        this.specialisation = specialisation;
        this.workingConditions = workingConditions;
        this.payment = payment;
```

```
        this.workingExperience = workingExperience;
        this.education = education;
        this.Licence = Licence;
        this.English = English;
    }


    public String getPayment() {
        return payment;
    }

    public void setPayment(String payment)
    {
        this.payment=payment;
    }

    public String getSpecialisation() {
        return specialisation;
    }

    public void setSpecialisation(String specialisation) {
        this.specialisation = specialisation;
    }

    public String getEducation() {
        return education;
    }

    public void setEducation(String education) {
        this.education = education;
    }

    public void setCompany(String company) {
        this.company = company;
    }

    public String getCompany() {
        return company;
    }

    public String getWorkingConditions() {
        return workingConditions;
    }

    public void setWorkingConditions(String workingConditions) {
        this.workingConditions = workingConditions;
    }

    public String getWorkingExperience() {
        return workingExperience;
    }

    public void setWorkingExperience(String workingExperience) {
        this.workingExperience = workingExperience;
    }

    public String getLicence() {
        return Licence;
    }

    public void setLicence(String licence) {
        Licence = licence;
```

```
    }

    public String getEnglish() {
        return English;
    }

    public void setEnglish(String english) {
        English = english;
    }

   /* @Override
    public String toString() {
        return "SecondCreate{" +
                "company = " + company + "\n" +
                "specialisation =" + specialisation + "\n" +
                "workingConditions='" + workingConditions + "\n" +
                "payment=" + payment + "\n" +
                "workingExperience=" + workingExperience + "\n" +
                "education='" + education + "\n" +
                "Licence='" + Licence + "\n" +
                "English='" + English + "\n" +
                '}';
    }*/

   /* @Override
    public int compareTo(SecondCreate o) {
        SecondCreate entry = (SecondCreate) o;

        int tmp = company.compareTo(entry.company);
        // this.payment - ((SecondCreate)o).payment;
        return tmp;
    }*/
}
```

List.fxml

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

<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TableColumn?>
<?import javafx.scene.control.TableView?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.text.Font?>
<?import javafx.scene.text.Text?>

<AnchorPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity"
minWidth="-Infinity" prefHeight="714.0" prefWidth="1275.0"
xmlns="http://javafx.com/javafx/11.0.1" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="sample.listController">
   <children>
      <AnchorPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity"
minWidth="-Infinity" opacity="0.84" prefHeight="720.0" prefWidth="1280.0" style="-fx-
background-color: #0b0b0b; -fx-border-color: #ffa600; -fx-border-width: 5px;">
         <children>
            <AnchorPane layoutX="5.0" layoutY="57.0" opacity="0.79"
prefHeight="659.0" prefWidth="1270.0" style="-fx-background-color: #ffffff;">
               <children>
                  <TableView fx:id="TableView" prefHeight="328.0" prefWidth="722.0">
                    <columns>
```

```
                    <TableColumn fx:id="company" onEditCommit="#editCompany"
prefWidth="63.99999713897705" text="Company" />
                    <TableColumn fx:id="specialisation"
prefWidth="90.39999389648438" text="Specialisation" />
                      <TableColumn fx:id="workingConditions" prefWidth="88.0"
text="Work.. Cond.." />
                      <TableColumn fx:id="payment" prefWidth="71.19996643066406"
text="Payment" />
                      <TableColumn fx:id="workingExperience"
prefWidth="100.00006103515625" text="Work... Exp..." />
                      <TableColumn fx:id="education" prefWidth="120.0"
text="Education" />
                      <TableColumn fx:id="English" prefWidth="108.800048828125"
text="English" />
                      <TableColumn fx:id="License" prefWidth="82.39996337890625"
text="License" />
                  </columns>
                </TableView>
                <Button fx:id="DeleteButton" layoutX="764.0" layoutY="192.0"
mnemonicParsing="false" onAction="#delete" prefHeight="51.0" prefWidth="217.0"
text="Delete Vacancy">
                  <font>
                    <Font size="24.0" />
                  </font>
                </Button>
                <Button fx:id="EditButton" layoutX="1027.0" layoutY="192.0"
mnemonicParsing="false" onAction="#edit" prefHeight="51.0" prefWidth="200.0"
text="Edit">
                  <font>
                    <Font size="24.0" />
                  </font>
                </Button>
                <Button fx:id="AddButton" layoutX="1026.0" layoutY="422.0"
mnemonicParsing="false" onAction="#add" prefHeight="51.0" prefWidth="207.0" text="Add
Vacancy">
                  <font>
                    <Font size="24.0" />
                  </font>
                </Button>
                <Button fx:id="FindButton" layoutX="1029.0" layoutY="98.0"
mnemonicParsing="false" text="Find">
                  <font>
                    <Font size="24.0" />
                  </font>
                </Button>
                <Text layoutX="805.0" layoutY="61.0" strokeType="OUTSIDE"
strokeWidth="0.0" text="Enter company name which u want to find">
                  <font>
                    <Font size="19.0" />
                  </font>
                </Text>
                <TextField fx:id="companyButton" layoutX="764.0" layoutY="299.0"
prefHeight="26.0" prefWidth="218.0" promptText="company" />
                <TextField fx:id="workingExpButton" layoutX="285.0" layoutY="368.0"
prefHeight="26.0" prefWidth="191.0" promptText="workingExp" />
                <TextField fx:id="paymentButton" layoutX="504.0" layoutY="368.0"
prefHeight="26.0" prefWidth="218.0" promptText="payment" />
                <TextField fx:id="conditionsButton" layoutX="768.0" layoutY="368.0"
prefHeight="26.0" prefWidth="218.0" promptText="workingCond" />
                <TextField fx:id="specialisationButton" layoutX="1028.0"
layoutY="299.0" prefHeight="26.0" prefWidth="207.0" promptText="specialisation" />
                <TextField fx:id="LicenseButton" layoutX="154.0" layoutY="368.0"
```

```
prefHeight="26.0" prefWidth="94.0" promptText="License" />
                  <TextField fx:id="educationButton" layoutX="1027.0" layoutY="368.0"
prefHeight="26.0" prefWidth="207.0" promptText="education" />
                  <TextField fx:id="EnglishButton" layoutX="29.0" layoutY="368.0"
prefHeight="26.0" prefWidth="88.0" promptText="English" />
                  <TextField fx:id="SearchField" layoutX="764.0" layoutY="111.0"
onKeyReleased="#SearchBtn" prefHeight="26.0" prefWidth="211.0" promptText="Company"
/>
               </children>
            </AnchorPane>
            <Label fx:id="joobleButton" layoutX="23.0" layoutY="10.0" opacity="0.84"
text="JOOBLE" textFill="#a89e9e">
               <font>
                  <Font name="Calibri" size="35.0" />
               </font>
            </Label>
            <Button fx:id="FindVacancyButton" layoutX="922.0" layoutY="9.0"
mnemonicParsing="false" prefHeight="42.0" prefWidth="144.0" style="-fx-background-
color: orange; -fx-border-radius: 10px;" text="Find Vacancy" textFill="WHITE">
               <font>
                  <Font size="19.0" />
               </font>
            </Button>
            <Button fx:id="PlaceVacancyButton" layoutX="1113.0" layoutY="9.0"
mnemonicParsing="false" prefHeight="42.0" prefWidth="144.0" style="-fx-background-
color: orange; -fx-border-radius: 10px;" text="Place Vacancy" textFill="#ffffff">
               <font>
                  <Font size="19.0" />
               </font>
            </Button>
         </children>
      </AnchorPane>
   </children>
</AnchorPane>
```

Controller.java

```
package sample;

import java.io.IOException;
import java.net.URL;
import java.util.ResourceBundle;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.image.ImageView;
import javafx.stage.Stage;

public class Controller {

    @FXML
    private ResourceBundle resources;

    @FXML
    private URL location;

    @FXML
    private ImageView EpamImageButton;
```

```
    @FXML
    private ImageView NixImageButton;

    @FXML
    private ImageView GLimageButton;

    @FXML
    private ImageView BeetrotAcademyImageButton;

    @FXML
    private Label joobleButton;

    @FXML
    private Button FindVacancyButton;

    @FXML
    private Button PlaceVacancyButton;


    @FXML
    void initialize() {

        PlaceVacancyButton.setOnAction(event -> {
            PlaceVacancyButton.getScene().getWindow().hide();
            FXMLLoader loader = new FXMLLoader();
            loader.setLocation(getClass().getResource("/sample/list.fxml"));
            try {
                loader.load();
            } catch (IOException e) {
                e.printStackTrace();
            }
            Parent root = loader.getRoot();
            Stage stage = new Stage();
            stage.setScene(new Scene(root));
            stage.showAndWait();
        });


    }
}
```

PlaceVacancy.fxml

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

<!--<?import com.gluonhq.charm.glisten.control.TextField?>-->

<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.text.Font?>

<AnchorPane fx:id="Anchor" maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-
Infinity" minWidth="-Infinity" opacity="0.84" prefHeight="600.0" prefWidth="700.0"
style="-fx-background-color: #0b0b0b; -fx-border-color: #ffa600; -fx-border-width:
5px;" xmlns="http://javafx.com/javafx/11.0.1" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="sample.PlaceVcController">
    <children>
        <AnchorPane layoutX="5.0" layoutY="65.0" opacity="0.79" prefHeight="529.0"
```

```
prefWidth="690.0" style="-fx-background-color: #ffffff;">
         <children>
            <TextField fx:id="companyField" layoutX="494.0" layoutY="112.0" />
            <TextField fx:id="specialisationField" layoutX="494.0" layoutY="157.0" />
            <TextField fx:id="workingConditions" layoutX="494.0" layoutY="204.0" />
            <TextField fx:id="paymentField" layoutX="494.0" layoutY="252.0" />
            <TextField fx:id="epxField" layoutX="494.0" layoutY="298.0" />
            <TextField fx:id="educationField" layoutX="494.0" layoutY="339.0" />
            <TextField fx:id="EnglishField" layoutX="494.0" layoutY="380.0" />
            <TextField fx:id="licenseField" layoutX="494.0" layoutY="423.0" />
            <Button fx:id="PvSubmitButton" layoutX="293.0" layoutY="457.0"
mnemonicParsing="false" onAction="#submit" style="-fx-background-color: orange; -fx-
border-radius: 5px;" text="Submit">
               <font>
                  <Font size="24.0" />
               </font>
            </Button>
         </children>
        </AnchorPane>
        <Label layoutX="23.0" layoutY="10.0" opacity="0.84" text="JOOBLE"
textFill="#a89e9e">
            <font>
                <Font name="Calibri" size="35.0" />
            </font>
        </Label>
        <Button fx:id="FindVacancyButton" layoutX="347.0" layoutY="5.0"
mnemonicParsing="false" prefHeight="52.0" prefWidth="144.0" style="-fx-background-
color: #0b0b0b;" text="Find Vacancy" textFill="WHITE">
            <font>
                <Font size="19.0" />
            </font>
        </Button>
        <Button fx:id="PlaceVacancyButton" layoutX="510.0" layoutY="5.0"
mnemonicParsing="false" prefHeight="52.0" prefWidth="144.0" style="-fx-background-
color: #0b0b0b;" text="Place Vacancy" textFill="#ffffff">
            <font>
                <Font size="19.0" />
            </font>
        </Button>
    </children>
</AnchorPane>
```

PlaceVcController.java

```
package sample;

import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.TextField;
import javafx.stage.Stage;

import java.io.IOException;

public class PlaceVcController {

    @FXML
    private TextField companyField;
```

```
    @FXML
    private TextField specialisationField;

    @FXML
    private TextField workingConditions;

    @FXML
    private TextField paymentField;

    @FXML
    private TextField epxField;

    @FXML
    private TextField educationField;

    @FXML
    private TextField EnglishField;

    @FXML
    private TextField licenseField;

    @FXML
    private Button PvSubmitButton;

    @FXML
    private Button FindVacancyButton;

    @FXML
    private Button PlaceVacancyButton;

    @FXML
    void submit(ActionEvent event) throws IOException {
       /* FXMLLoader loader = new
FXMLLoader(getClass().getResource("sample/list.fxml"));
        Parent root = (Parent)loader.load();*/


        /*FXMLLoader loader = new FXMLLoader();
        loader.setLocation(getClass().getResource("/sample/list.fxml"));
        try {
            loader.load();
            listController listController = loader.getController();

listController.myFunction(companyField.getText(),specialisationField.getText(),workin
gConditions.getText(),paymentField.getText(),

epxField.getText(),educationField.getText(),EnglishField.getText(),licenseField.getTe
xt());
            Parent root = loader.getRoot();
            Stage stage = new Stage();
            stage.setScene(new Scene(root));
            stage.showAndWait();
        } catch (IOException e) {
            e.printStackTrace();
        }*/

        Parent root;
        try {
            FXMLLoader loader = new
FXMLLoader(getClass().getResource("/sample/list.fxml"));
            root = loader.load();
```

```
            Stage stage = new Stage();
            stage.setTitle("Second window");

            Scene scene = new Scene(root);
            stage.setScene(scene);
            stage.initOwner(PvSubmitButton.getScene().getWindow());

            //SecondWindowController controller = loader.getController();
            //controller.someMethodCall();

            stage.showAndWait();
        } catch (Exception exc) {
            exc.printStackTrace();
        }



        /*Stage stage = new Stage();
        stage.setScene(new Scene(root));
        stage.show();*/

    }

    @FXML
    void initialize()
    {

    }

}
```

ВАРІАНТИ ВИКОРИСТАННЯ

Рисунок 1 – Головне меню програми

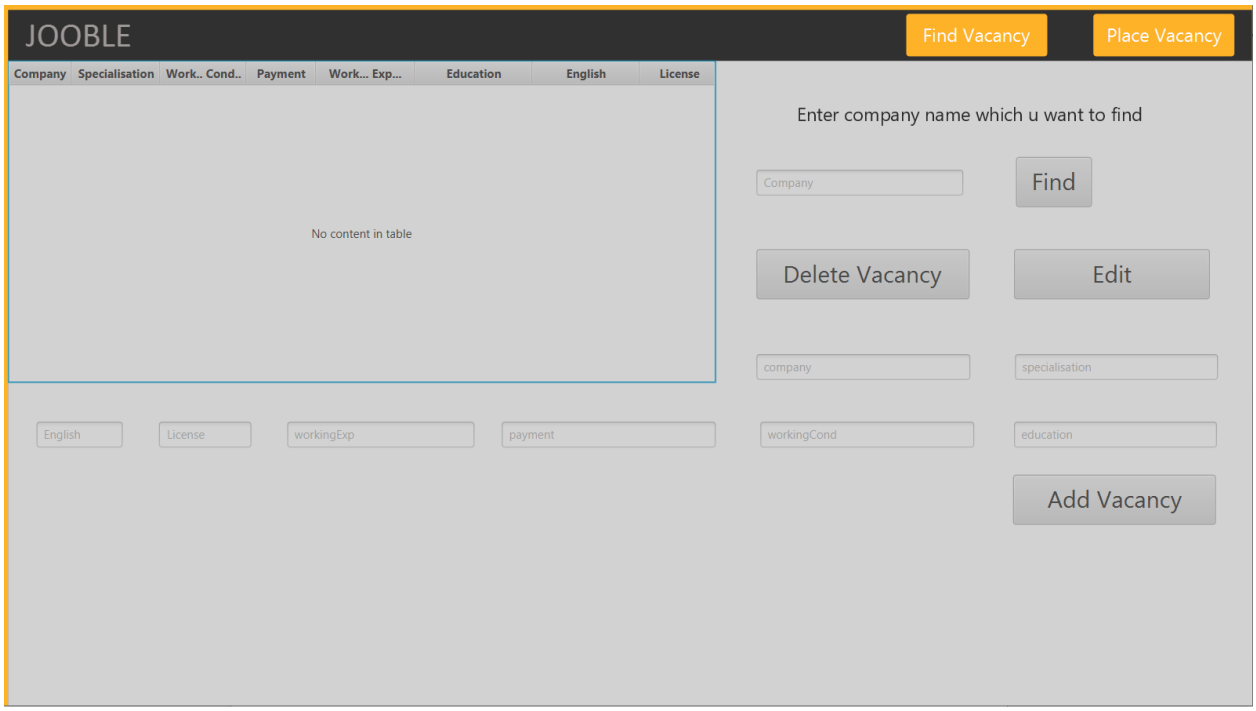

Рисунок 2 – Вікно в якому відбувається додавання, видалення, редагування та пошук елементів у контейнері.

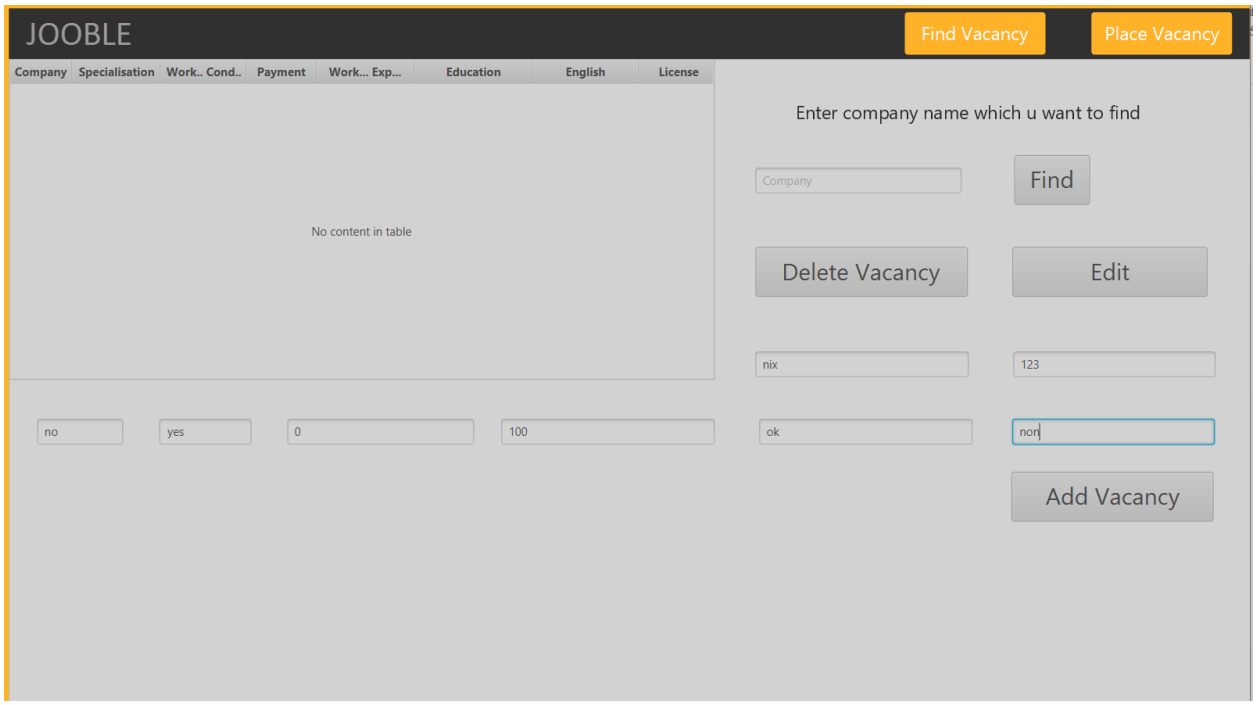

Рисунок 3 – додавання елементу в контейнер

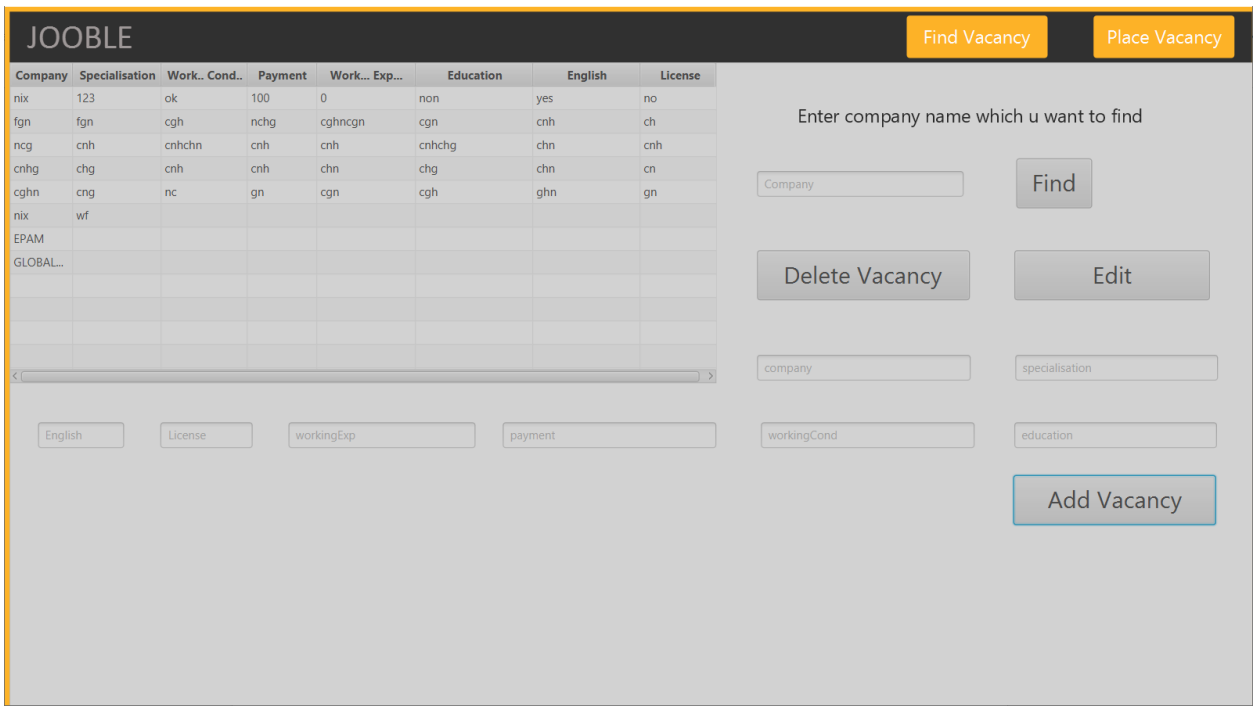

Рисунок 4 – результат додавання

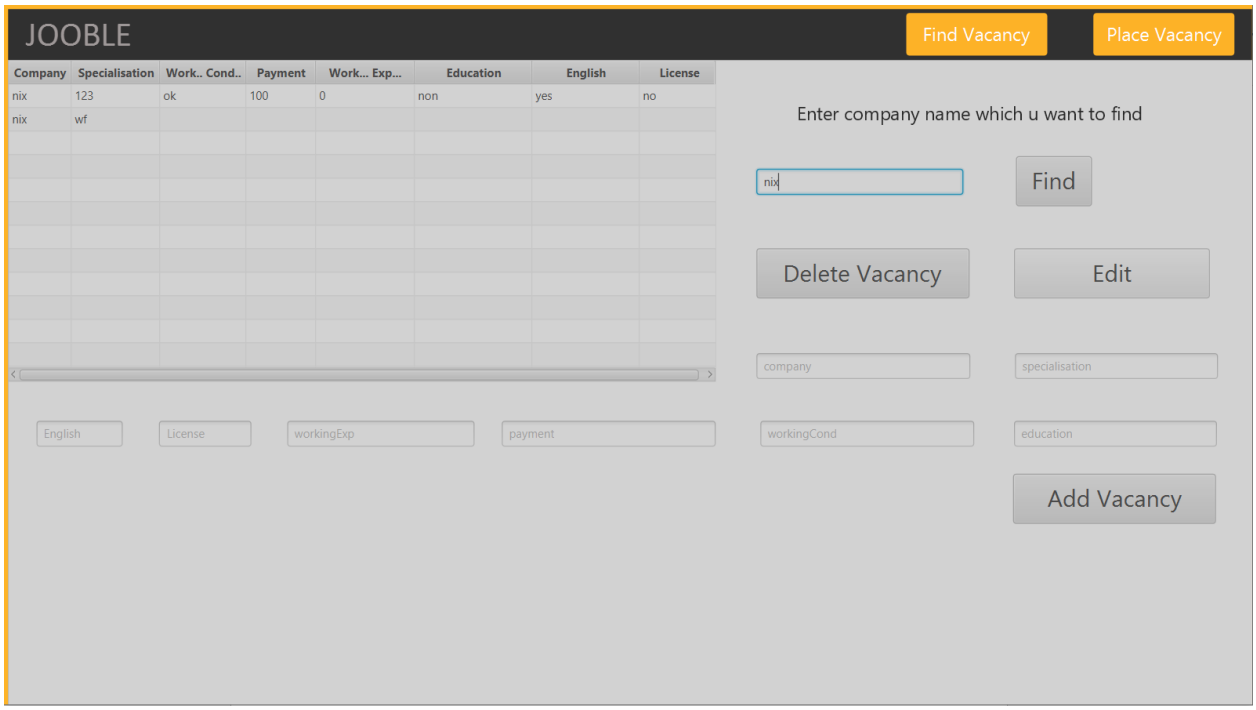

Рисунок 5 – пошук елементів за заданою назвою фірми

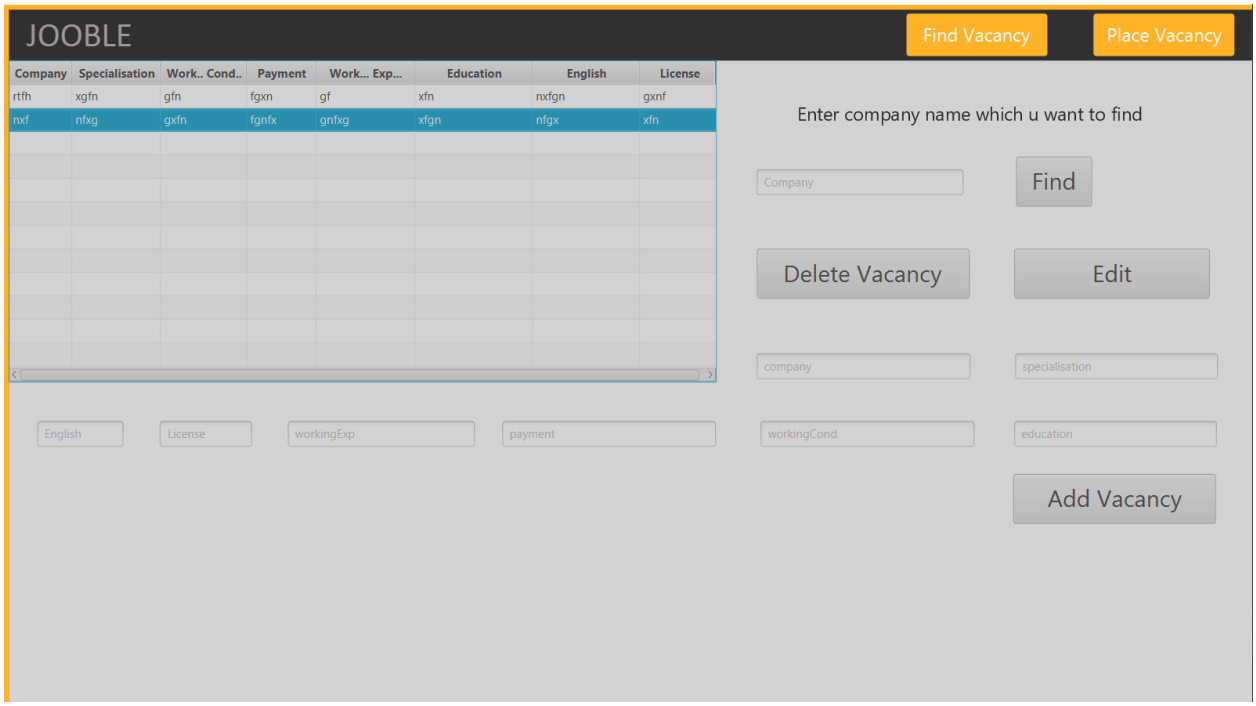

Рисунок 6 – видалення елементу

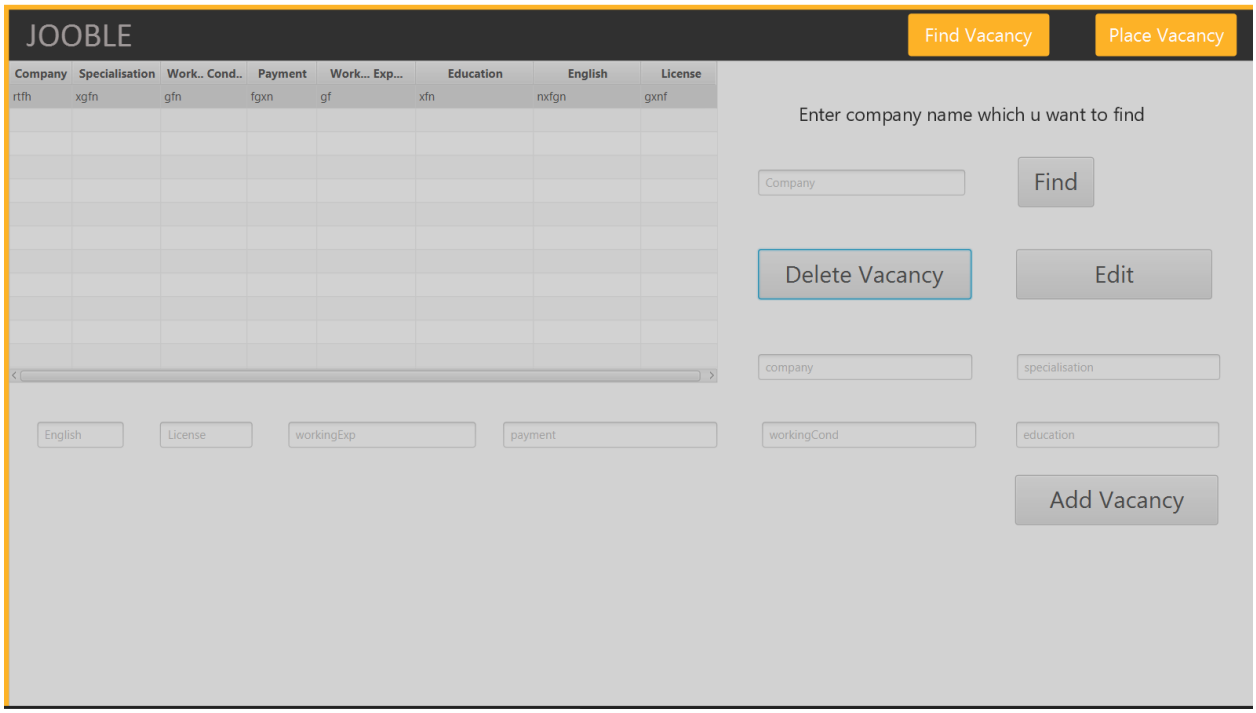

Рисунок 7 – результат видалення елементу

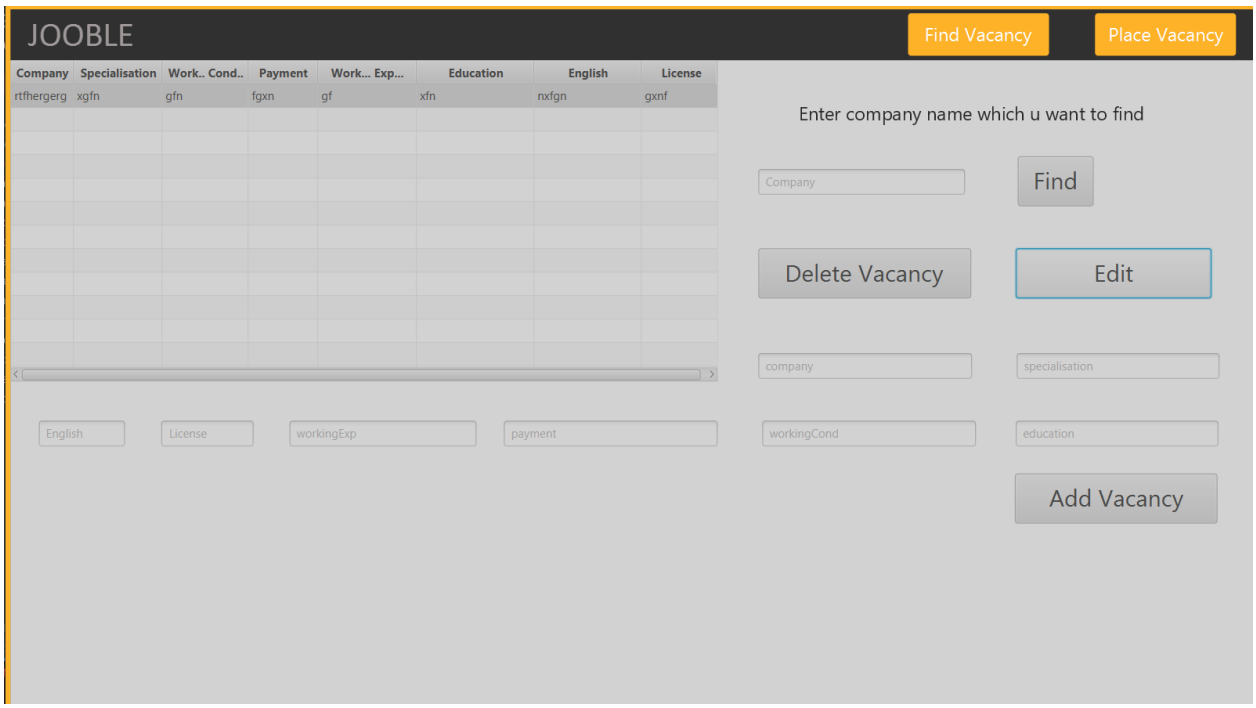

Рисунок 8 – назва фірми 1 елементу контейнеру була відредагована.

Програму можна використовувати задля створення бази даних. Завдяки параметризації зв'язного списка, базу даних можна використати для будь-яких типів даних. Переважно у нашому варіанті - кадрове агенство, в якому представляються різноманітні вакансії. Також для вибору доступно багато інших можливостей. Реалізовано меню для поліпшення користування програмою.

ВИСНОВКИ

При виконанні лабораторної роботи набуто практичних навичок щодо розробки параметризованих класів. Завдяки цієї можливості в JAVA, можливо створювати колекції та інші класи на основі будь-яких типів. Також навчився обробляти параметризовані контейнери. Завдання виконане! Програма працює успішно!