Министерство образования и науки Российской Федерации Федеральное агентство по образованию

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Вятский государственный университет»

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Связывание приложения на C# с базой данных под управлением PostgreSQL

Лабораторная работа №4

по курсу «Базы данных»

Выполнил студент группы ИВТб-31 /Категов А.Д/ Проверил преподаватель /Клюкин В.Л./

Киров 2024

**Цели лабораторной работы:**

1. Познакомиться c библиотекой в C++ для связывания приложения с БД;
2. Изучить некоторые шаблоны проектирования, связанные с работой с БД;
3. Освоить на практике основы взаимодействия с БД под управлением PostgreSQL в приложении на C++.

**Задание на лабораторную работу:**

* Выводить строки выбранной таблицы;
* Предоставлять возможность добавить новую строку в таблицу;
* Предоставлять возможность удалить строку из таблицы.

**Ход работы:**

1. Реализация подключения к базе данных.

bool ConnectOpen()

{

String^ sql = String::Format("Host={0};Username={1};Password={2};Database={3}",

textBox1->Text, textBox2->Text, textBox3->Text, textBox4->Text);

connect = gcnew NpgsqlConnection(sql);

try

{

connect->Open();

return true;

}

catch (Exception^ ex)

{

return false;

}

}

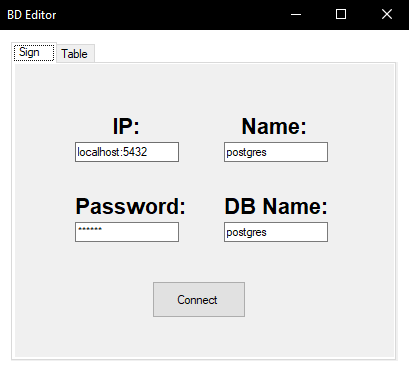


Рисунок 1 – Экранная форма

2. Вывод таблицы

bool LoadTable()

{

if (!ConnectOpen()) return false;

NpgsqlCommand^ cmd = gcnew NpgsqlCommand(

"select e.first\_name as Имя, e.second\_name as Фамилия, e.middle\_name as Отчество, v.model as Модель, v.kind as Gear, v.id as Номер\n" +

"from examinee e, vehicle v\n" +

"where v.id = e.id\_vehicle;", connect);

DataTable^ dt = gcnew DataTable();

dt->Load(cmd->ExecuteReader());

dataGridView->DataSource = dt;

connect->Close();

return true;

}

3. Функция добавления элемента

void buttonAdd\_MouseClick(System::Object^ sender, System::Windows::Forms::MouseEventArgs^ e)

{

if (textBoxName->Text->Length > 0)

{

ConnectOpen();

int id;

Int32::TryParse(textBoxId->Text, id);

array<String^>^ NameArr = textBoxName->Text->Split(' ');

try

{

NpgsqlCommand^ cmd = gcnew NpgsqlCommand(

String::Format("insert into examinee(first\_name, second\_name, middle\_name, id\_vehicle) values('{0}', '{1}', '{2}', {3});",

NameArr[0], NameArr[1], NameArr[2], id), connect);

System::Data::Common::DbDataReader^ reader = cmd->ExecuteReader();

}

catch (Exception^ ex)

{

MessageBox::Show("Ошибка.\nВведены неверные данные");

}

connect->Close();

LoadTable();

}

}

4. Функция удаления элемента

void buttonDelete\_MouseClick(System::Object^ sender, System::Windows::Forms::MouseEventArgs^ e)

{

ConnectOpen();

int id;

Int32::TryParse(textBoxId->Text, id);

array<String^>^ NameArr = textBoxName->Text->Split(' ');

try

{

NpgsqlCommand^ cmd = gcnew NpgsqlCommand(

String::Format("DELETE FROM examinee WHERE first\_name = '{0}' and second\_name = '{1}' and middle\_name = '{2}' and id\_vehicle = {3};",

NameArr[0], NameArr[1], NameArr[2], id), connect);

System::Data::Common::DbDataReader^ reader = cmd->ExecuteReader();

}

catch (Exception^ ex)

{

MessageBox::Show("Ошибка.\nВведены неверные данные");

}

connect->Close();

LoadTable();

}

**Вывод:** в ходе выполнения данной лабораторной работы были ознакомлены с библиотекой в C++ для связывания приложения с БД. Изучены некоторые шаблоны проектирования, связанные с работой с БД. Освоены на практике основы взаимодействия с БД под управлением PostgreSQL в приложении на C++.