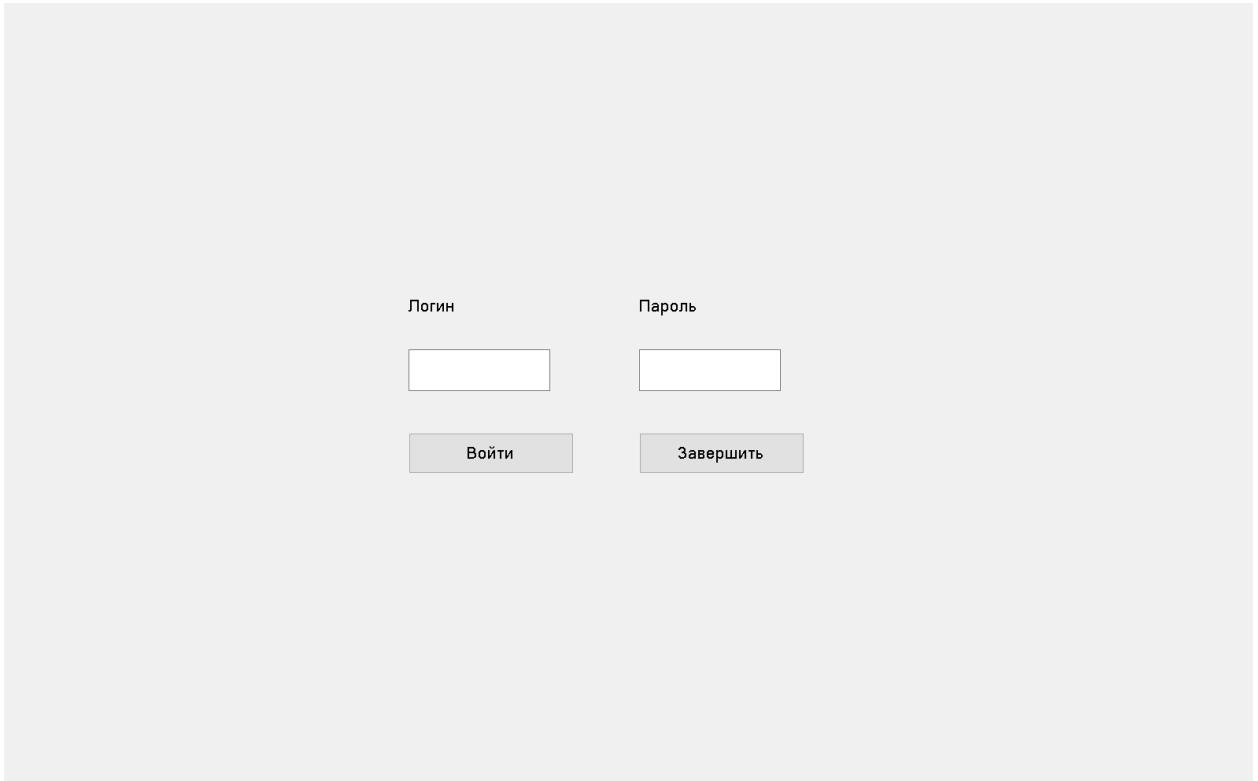


Form 1



Код:

```
namespace WindowsFormsApp16
{
    public partial class Form1 : Form
    {
        private Dictionary<string, (string password, UserRole role)> users = new
Dictionary<string, (string, UserRole)>
        {
            { "admin", ("admin123", UserRole.Admin) },
            { "client", ("client123", UserRole.Client) }
        };

        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            string username = textBox1.Text.Trim();
            string password = textBox2.Text;
            if (string.IsNullOrEmpty(username) || string.IsNullOrEmpty(password))
            {
                MessageBox.Show("Введите логин и пароль", "Ошибка",
                MessageBoxButtons.OK);
            }
        }
    }
}
```

```
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    return;
}

if (users.ContainsKey(username))
{
    var userData = users[username];

    if (userData.password == password)
    {
        UserSession.Username = username;
        UserSession.Role = userData.role;

        Form2 form2 = new Form2();
        form2.ShowDialog();
    }
    else
    {
        MessageBox.Show("Неверный пароль", "Ошибка",
            MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
else
{
    MessageBox.Show("Пользователь не найден", "Ошибка",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
}

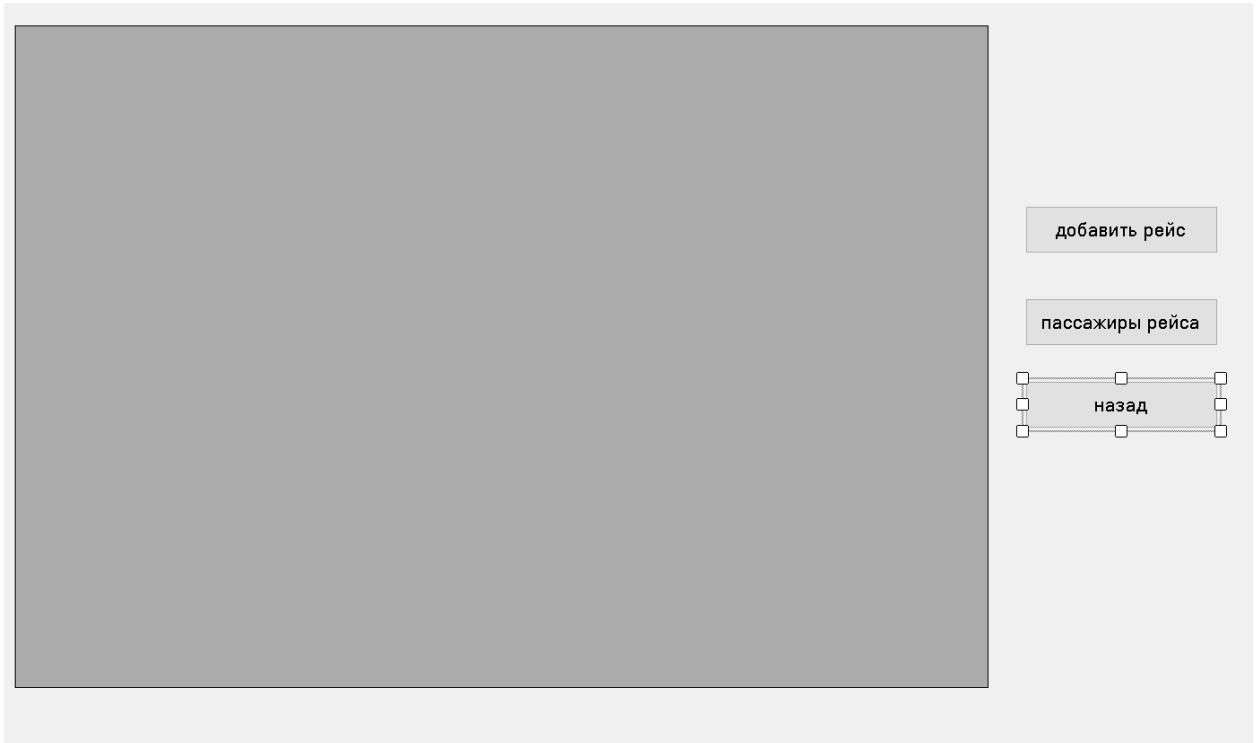
private void button2_Click(object sender, EventArgs e)
{
    Application.Exit();
}

public enum UserRole
{
    Admin,
    Client
}

public static class UserSession
{
    public static UserRole Role { get; set; }
    public static string Username { get; set; }
```

```
        }  
    }
```

Form 2



Код:

```
using Npgsql;  
  
namespace WindowsFormsApp16  
{  
    public partial class Form2 : Form  
    {  
        private string connectionString =  
"Host=localhost;Port=5432;Database=kval;Username=postgres;Password=1234";  
  
        public Form2()  
        {  
            InitializeComponent();  
        }  
  
        private void Form2_Load(object sender, EventArgs e)  
        {  
            bool isAdmin = UserSession.Role == UserRole.Admin;  
            button1.Visible = isAdmin;  
            button2.Visible = isAdmin;  
            LoadTripsData();  
        }  
    }
```

```
private void LoadTripsData()
{
    try
    {
        using (NpgsqlConnection connection = new
NpgsqlConnection(connectionString))
        {
            connection.Open();

            string query = "SELECT * FROM public.trips ORDER BY id";

            using (NpgsqlDataAdapter adapter = new NpgsqlDataAdapter(query,
connection))
            {
                DataTable dataTable = new DataTable();
                adapter.Fill(dataTable);

                dataGridView1.DataSource = dataTable;
                dataGridView1.AutoSizeColumnsMode =
DataGridViewAutoSizeColumnsMode.Fill;

                if (dataGridView1.Columns.Count > 0)
                {
                    dataGridView1.Columns["id"].HeaderText = "ID";
                    dataGridView1.Columns["trips_id"].HeaderText = "ID
Поездки";
                    dataGridView1.Columns["company_id"].HeaderText = "ID
Компании";
                    dataGridView1.Columns["company"].HeaderText =
"Компания";
                    dataGridView1.Columns["city_to"].HeaderText = "Город
прибытия";
                    dataGridView1.Columns["city_from"].HeaderText = "Город
отправления";
                    dataGridView1.Columns["data_to"].HeaderText = "Дата
прибытия";
                    dataGridView1.Columns["data_from"].HeaderText = "Дата
отправления";
                    dataGridView1.Columns["capacity"].HeaderText =
"Вместимость";
                }
            }
        }
    }
}
```

```

        catch (Exception ex)
    {
        MessageBox.Show($"Ошибка при загрузке данных: {ex.Message}",
        "Ошибка",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

private void dataGridView1_CellContentClick(object sender,
DataGridViewCellEventArgs e)
{
    if (e.RowIndex >= 0 && e.ColumnIndex >= 0)
    {
        DataGridViewRow row = dataGridView1.Rows[e.RowIndex];

        string tripId = row.Cells["trips_id"].Value?.ToString();
        string company = row.Cells["company"].Value?.ToString();
        string cityFrom = row.Cells["city_from"].Value?.ToString();
        string cityTo = row.Cells["city_to"].Value?.ToString();
        string dateFrom = row.Cells["data_from"].Value?.ToString();
        string dateTo = row.Cells["data_to"].Value?.ToString();
        string capacity = row.Cells["capacity"].Value?.ToString();

        string message = $"Информация о поездке:\n" +
            $"ID: {tripId}\n" +
            $"Компания: {company}\n" +
            $"Откуда: {cityFrom}\n" +
            $"Куда: {cityTo}\n" +
            $"Дата отправления: {dateFrom}\n" +
            $"Дата прибытия: {dateTo}\n" +
            $"Вместимость: {capacity} человек";

        MessageBox.Show(message, "Информация о поездке",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}

private void button1_Click(object sender, EventArgs e)
{
    Form4 form = new Form4();
    form.ShowDialog();
}

private void button2_Click(object sender, EventArgs e)
{
}

```

```
        Form3 form = new Form3();
        form.ShowDialog();
    }

    private void button3_Click(object sender, EventArgs e)
    {
        Form1 form = new Form1();
        form.ShowDialog();
    }
}
```

Form 3



Код:

```
using Npgsql;

namespace WindowsFormsApp16
{
    public partial class Form3 : Form
    {
        private string connectionString =
            "Host=localhost;Port=5432;Database=kval;Username=postgres;Password=1234";
```

```

public Form3()
{
    InitializeComponent();
}

private void Form3_Load(object sender, EventArgs e)
{
    LoadAllTickets();
}

private void LoadAllTickets()
{
    try
    {
        using (NpgsqlConnection connection = new
NpgsqlConnection(connectionString))
        {
            connection.Open();
            string query = "SELECT * FROM public.tickets ORDER BY id";

            using (NpgsqlDataAdapter adapter = new NpgsqlDataAdapter(query,
connection))
            {
                DataTable dataTable = new DataTable();
                adapter.Fill(dataTable);
                dataGridView1.DataSource = dataTable;
            }
        }
    catch (Exception ex)
    {
        MessageBox.Show($"Ошибка подключения: {ex.Message}\n\n" +
                    "Возможные причины:\n" +
                    "1. PostgreSQL не запущен\n" +
                    "2. Не установлен Npgsql пакет\n" +
                    "3. Неправильные данные подключения\n\n" +
                    "Для установки Npgsql:\n" +
                    "1. Откройте Package Manager Console\n" +
                    "2. Введите: Install-Package Npgsql -Version 6.0.10",
                    "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

private void button1_Click(object sender, EventArgs e)

```

```

{
    string firstName = textBox2.Text.Trim();
    string midName = textBox3.Text.Trim();
    string lastName = textBox1.Text.Trim();

    try
    {
        using (NpgsqlConnection connection = new
NpgsqlConnection(connectionString))
        {
            connection.Open();

            string query = "SELECT * FROM public.tickets WHERE 1=1";

            if (!string.IsNullOrWhiteSpace(firstName))
                query += " AND first_name LIKE @firstName";
            if (!string.IsNullOrWhiteSpace(midName))
                query += " AND mid_name LIKE @midName";
            if (!string.IsNullOrWhiteSpace(lastName))
                query += " AND last_name LIKE @lastName";

            query += " ORDER BY id";

            using (NpgsqlCommand command = new NpgsqlCommand(query,
connection))
            {
                if (!string.IsNullOrWhiteSpace(firstName))
                    command.Parameters.AddWithValue("@firstName",
$"%" {firstName}%" );
                if (!string.IsNullOrWhiteSpace(midName))
                    command.Parameters.AddWithValue("@midName",
$"%" {midName}%" );
                if (!string.IsNullOrWhiteSpace(lastName))
                    command.Parameters.AddWithValue("@lastName",
$"%" {lastName}%" );

                using (NpgsqlDataAdapter adapter = new
NpgsqlDataAdapter(command))
                {
                    DataTable dataTable = new DataTable();
                    adapter.Fill(dataTable);
                    dataGridView1.DataSource = dataTable;
                }
            }
        }
    }
}

```

```

        }
    catch (Exception ex)
    {
        MessageBox.Show($"Ошибка при поиске: {ex.Message}", "Ошибка",
            MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

private void dataGridView1_CellContentClick(object sender,
DataGridViewCellEventArgs e)
{
    if (e.RowIndex >= 0)
    {
        DataGridViewRow row = dataGridView1.Rows[e.RowIndex];

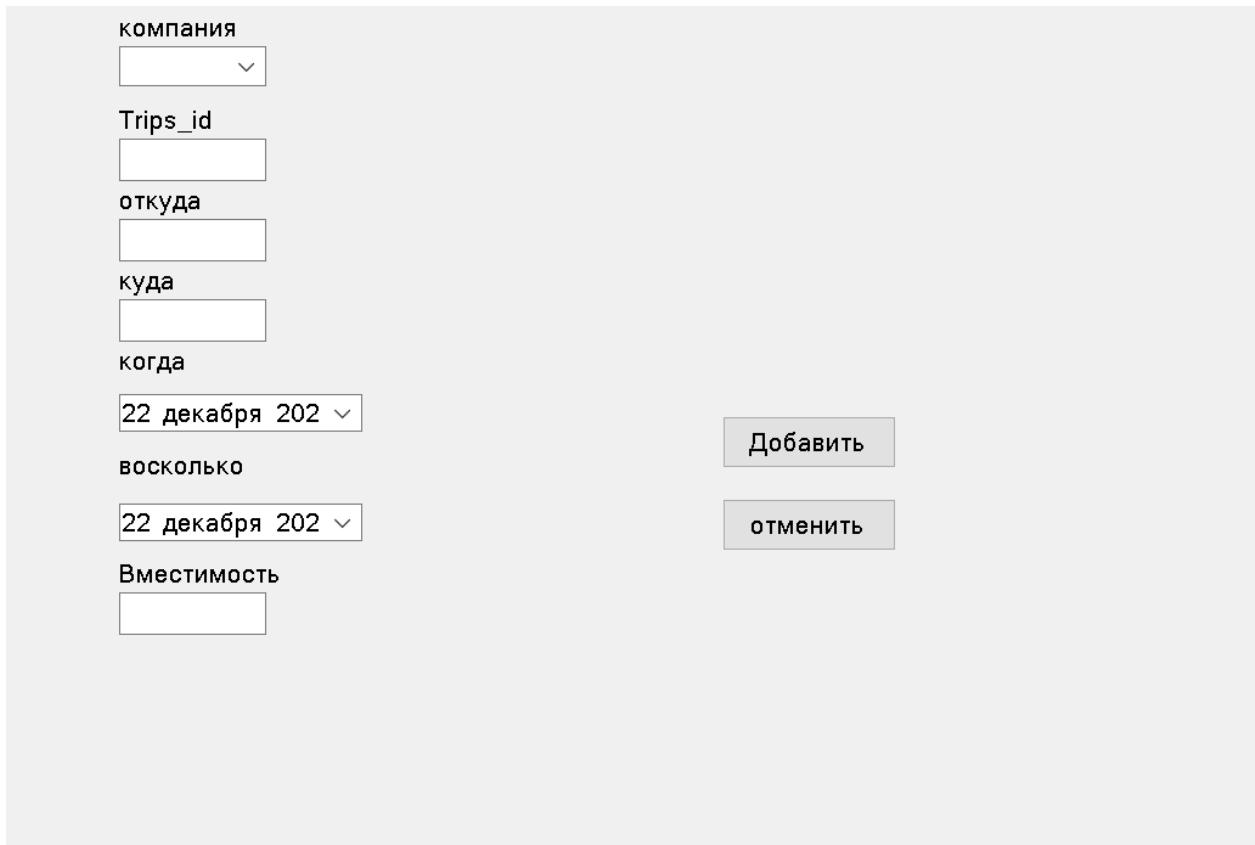
        string id = row.Cells["id"].Value?.ToString() ?? "";
        string firstName = row.Cells["first_name"].Value?.ToString() ?? "";
        string midName = row.Cells["mid_name"].Value?.ToString() ?? "";
        string lastName = row.Cells["last_name"].Value?.ToString() ?? "";

        string message = $"ID: {id}\nФИО: {lastName} {firstName} {midName}";
        MessageBox.Show(message, "Информация",
            MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}

private void button2_Click(object sender, EventArgs e)
{
    Form2 form = new Form2();
    form.ShowDialog();
}
}

```

Form 4



Код:

```
using Npgsql;  
  
namespace WindowsFormsApp16  
{  
    public partial class Form4 : Form  
    {  
        private string connectionString =  
"Host=localhost;Port=5432;Database=kval;Username=postgres;Password=1234";  
  
        public Form4()  
        {  
            InitializeComponent();  
        }  
  
        private void Form4_Load(object sender, EventArgs e)  
        {  
            comboBox1.Items.AddRange(new string[] { "РЖД", "Аэрофлот",  
"Автобусы России", "Речной флот" });  
            if (comboBox1.Items.Count > 0)  
                comboBox1.SelectedIndex = 0;  
        }  
    }  
}
```

```

private void button1_Click(object sender, EventArgs e)
{
    if (string.IsNullOrWhiteSpace(comboBox1.Text) ||
        string.IsNullOrWhiteSpace(textBox1.Text) ||
        string.IsNullOrWhiteSpace(textBox2.Text))
    {
        MessageBox.Show("Заполните все поля", "Ошибка",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        return;
    }

    if (!int.TryParse(textBox3.Text, out int capacity) || capacity <= 0)
    {
        MessageBox.Show("Введите корректную вместимость", "Ошибка",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        return;
    }

    try
    {
        InsertTrip(comboBox1.Text, textBox1.Text, textBox2.Text,
            dateTimePicker1.Value, dateTimePicker2.Value, capacity);

        MessageBox.Show("Поездка добавлена!", "Успех",
            MessageBoxButtons.OK, MessageBoxIcon.Information);

        ClearFields();
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Ошибка: {ex.Message}", "Ошибка",
            MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

private void InsertTrip(string company, string cityFrom, string cityTo,
    DateTime dateFrom, DateTime dateTo, int capacity)
{
    using (NpgsqlConnection connection = new
        NpgsqlConnection(connectionString))
    {
        connection.Open();

        int newId = GetNextId(connection);
    }
}

```

```

        int newTripId = GetNextTripId(connection);

        string query = @"INSERT INTO public.trips
                        (id, trips_id, company_id, company, city_from, city_to,
                        data_from, data_to, capacity)
                        VALUES (@id, @trips_id, 1, @company, @city_from,
                        @city_to, @data_from, @data_to, @capacity)";

        using (NpgsqlCommand command = new NpgsqlCommand(query,
connection))
        {
            command.Parameters.AddWithValue("@id", newId);
            command.Parameters.AddWithValue("@trips_id", newTripId);
            command.Parameters.AddWithValue("@company", company);
            command.Parameters.AddWithValue("@city_from", cityFrom);
            command.Parameters.AddWithValue("@city_to", cityTo);
            command.Parameters.AddWithValue("@data_from",
dateFrom.ToString("yyyy-MM-dd HH:mm"));
            command.Parameters.AddWithValue("@data_to",
dateTo.ToString("yyyy-MM-dd HH:mm"));
            command.Parameters.AddWithValue("@capacity", capacity);

            command.ExecuteNonQuery();
        }
    }

private int GetNextId(NpgsqlConnection connection)
{
    string query = "SELECT COALESCE(MAX(id), 0) + 1 FROM
public.trips";
    using (NpgsqlCommand command = new NpgsqlCommand(query,
connection))
    {
        return Convert.ToInt32(command.ExecuteScalar());
    }
}

private int GetNextTripId(NpgsqlConnection connection)
{
    string query = "SELECT COALESCE(MAX(trips_id), 0) + 1 FROM
public.trips";
    using (NpgsqlCommand command = new NpgsqlCommand(query,
connection))
    {

```

```
        return Convert.ToInt32(command.ExecuteScalar());
    }
}

private void ClearFields()
{
    if (comboBox1.Items.Count > 0)
        comboBox1.SelectedIndex = 0;
    textBox1.Text = "";
    textBox2.Text = "";
    textBox3.Text = "";
    dateTimePicker1.Value = DateTime.Now;
    dateTimePicker2.Value = DateTime.Now.AddHours(2);
}

private void button2_Click(object sender, EventArgs e)
{
    Form2 form = new Form2();
    form.ShowDialog();
}
}
```