На главной форме добавил PictureBox для того чтобы туда генерировался QR

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Код для генерации –

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Результат - Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Добавил нового пользователя – Менеджер в БД

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Написал код формы для менеджера с конкретными условиями

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Appliances

{

public partial class Manager : Form

{

private static SqlConnection connection;

private SqlCommand command;

private int userId; // Идентификатор пользователя

public Manager(int userId)

{

InitializeComponent();

dataGridViewRequests.DefaultCellStyle.Font = new Font("Arial", 12);

dataGridViewRequests.ColumnHeadersDefaultCellStyle.Font = new Font("Arial", 12);

StartPosition = FormStartPosition.CenterScreen;

this.userId = userId; // Сохраняем переданный userId

LoadUserInfo();

LoadRequests();

}

static private void Connect()

{

try

{

connection = new SqlConnection("Data Source=DESKTOP-P7R9Q2V;Initial Catalog=Yarema;Integrated Security=True;");

connection.Open();

}

catch (SqlException ex)

{

Console.WriteLine($"Ошибка доступа к базе данных. Исключение: {ex.Message}");

}

}

private void LoadUserInfo()

{

Connect();

string userInfoQuery = @"

SELECT FullName, Phone, UserTypeName

FROM Users

JOIN UserTypes ON Users.UserTypeID = UserTypes.UserTypeID

WHERE UserID = @UserId";

try

{

using (command = new SqlCommand(userInfoQuery, connection))

{

command.Parameters.AddWithValue("@UserId", this.userId);

using (SqlDataReader reader = command.ExecuteReader())

{

if (reader.Read())

{

label1.Text = reader["FullName"].ToString(); // ФИО

label2.Text = reader["Phone"].ToString(); // Номер телефона

label3.Text = reader["UserTypeName"].ToString(); // Роль

}

}

}

}

catch (SqlException ex)

{

Console.WriteLine($"Ошибка получения данных о пользователе: {ex.Message}");

}

finally

{

if (connection.State == ConnectionState.Open)

{

connection.Close();

}

}

}

private void LoadRequests()

{

Connect();

string query = @"

SELECT

r.RequestID,

r.StartDate,

h.orgTechType AS ModelType,

h.orgTechModel,

r.ProblemDescription,

s.StatusName,

r.CompletionDate,

p.RepairPartName,

u.FullName AS MasterFullName,

ud.FullName AS ClientFullName,

c.Message AS MasterComment

FROM Requests r

INNER JOIN Models h ON r.ModelID = h.ModelID

LEFT JOIN Statuses s ON r.RequestStatusID = s.StatusID

LEFT JOIN Parts p ON r.RepairPartID = p.RepairPartID

LEFT JOIN Users u ON r.MasterID = u.UserID

LEFT JOIN Users ud ON r.ClientID = ud.UserID

LEFT JOIN Comments c ON r.RequestID = c.RequestID AND r.MasterID = c.MasterID

WHERE r.MasterID IS NULL";

SqlCommand command = new SqlCommand(query, connection);

SqlDataAdapter adapter = new SqlDataAdapter(command);

DataTable dt = new DataTable();

adapter.Fill(dt);

dataGridViewRequests.Rows.Clear();

foreach (DataRow row in dt.Rows)

{

string[] rowData = new string[]

{

row["RequestID"].ToString(),

Convert.ToDateTime(row["StartDate"]).ToString("yyyy-MM-dd HH:mm"),

row["ModelType"].ToString(),

row["orgTechModel"].ToString(),

row["ProblemDescription"].ToString(),

row["StatusName"].ToString(),

row["CompletionDate"] != DBNull.Value ? Convert.ToDateTime(row["CompletionDate"]).ToString("yyyy-MM-dd HH:mm") : "",

row["RepairPartName"].ToString(),

row["MasterFullName"].ToString(),

row["ClientFullName"].ToString(),

row["MasterComment"].ToString()

};

dataGridViewRequests.Rows.Add(rowData);

}

labelTotalRequests.Text = "Всего заявок: " + dt.Rows.Count;

connection.Close();

}

private void button5\_Click(object sender, EventArgs e)

{

Close();

Authorization authorization = new Authorization();

authorization.ShowDialog();

}

private void button1\_Click(object sender, EventArgs e)

{

if (dataGridViewRequests.SelectedRows.Count > 0)

{

int requestId = Convert.ToInt32(dataGridViewRequests.SelectedRows[0].Cells["RequestID"].Value);

DateTime start = Convert.ToDateTime(dataGridViewRequests.SelectedRows[0].Cells["StartDate"].Value);

string type1 = dataGridViewRequests.SelectedRows[0].Cells["OrgTechType"].Value.ToString();

string problem = dataGridViewRequests.SelectedRows[0].Cells["ProblemDescription"].Value.ToString();

string status = dataGridViewRequests.SelectedRows[0].Cells["StatusName"].Value.ToString();

string client = dataGridViewRequests.SelectedRows[0].Cells["ClientFullName"].Value.ToString();

string model = dataGridViewRequests.SelectedRows[0].Cells["orgTechModel"].Value.ToString();

AddOper addOperForm = new AddOper(requestId, start, type1, model, problem, status, client);

addOperForm.ShowDialog();

dataGridViewRequests.Rows.Clear();

LoadRequests();

}

else

{

MessageBox.Show("Пожалуйста, выберите заказ.");

}

LoadRequests();

}

}

}

Результат –

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание