<https://udemy-certificate.s3.amazonaws.com/image/UC-a4fd27c6-68a1-459b-92df-6e017b8f5d1c.jpg?v=1666958400000>

<https://myvisit.card.uralsk-07.kz/>

**Лабораторная работа №13. Разработка и интеграция модулей проекта**

**Цель работы:**

Закрепление практических навыков работы с **Visual Studio 2019** и **MS SQL Server**, выполнение интеграции программных модулей.

**Выполнение задания**

**1. Создание базы данных в MS SQL Server**

1. **Запуск MS SQL Server Management Studio:**
   * Запустите **SSMS** и подключитесь к серверу, используя свои учетные данные.
     + Сервер: localhost (или другой указанный адрес).
     + Аутентификация: Windows или SQL Server Authentication.
2. **Создание базы данных:**
   * В панели **Object Explorer** щелкните правой кнопкой на **Databases** → **New Database**.
   * Укажите имя базы данных, например, UserManagementDB.
3. **Создание таблиц:**
   * Выберите созданную базу данных и щелкните правой кнопкой на **Tables** → **New Table**.

Таблица Users:

* + Поля:
    - UserID (int, Primary Key, Identity: начальное значение = 1, шаг = 1).
    - UserName (nvarchar(50), Not Null).
    - PasswordHash (nvarchar(64), Not Null).
    - Email (nvarchar(100), Not Null).
    - SecretQuestionID (int, Foreign Key).

Таблица SecretQuestions:

* + Поля:
    - SecretQuestionID (int, Primary Key, Identity).
    - Question (nvarchar(255), Not Null).

1. **Связывание таблиц:**
   * В панели **Database Diagrams** создайте новую диаграмму.
   * Перетащите таблицы Users и SecretQuestions.
   * Создайте связь между SecretQuestionID в Users и SecretQuestionID в SecretQuestions.
2. **Наполнение таблиц тестовыми данными:**
   * Выполните следующий SQL-запрос для добавления данных:

sql

Копировать код

INSERT INTO SecretQuestions (Question)

VALUES

('What is your favorite color?'),

('What is your pet’s name?'),

('What is your mother’s maiden name?');

**2. Создание приложения с окном авторизации**

1. **Создание проекта:**
   * В Visual Studio создайте новый проект:
     + **Тип:** Windows Forms App (.NET Framework).
     + **Имя:** UserAuthApp.
2. **Дизайн окна авторизации:**
   * Добавьте следующие элементы управления:
     + Метка для имени пользователя (lblUsername).
     + Текстовое поле для имени пользователя (txtUsername).
     + Метка для пароля (lblPassword).
     + Текстовое поле для пароля (txtPassword).
     + Кнопка для авторизации (btnLogin).
     + Метка для вывода ошибок (lblError).
   * Настройте свойства элементов, например:
     + Для txtPassword установите UseSystemPasswordChar = True.
3. **Пример интерфейса:**

markdown

Копировать код

Логин: [\_\_\_\_\_\_\_\_\_]

Пароль: [\_\_\_\_\_\_\_\_\_]

[ Войти ]

**3. Создание модели данных**

1. **Подключение Entity Framework:**
   * В Visual Studio добавьте Entity Framework:
     + **Tools** → **NuGet Package Manager** → **Manage NuGet Packages for Solution**.
     + Установите пакет EntityFramework.
2. **Создание модели:**
   * Щелкните правой кнопкой на проекте → **Add** → **New Item** → **ADO.NET Entity Data Model**.
   * Выберите **EF Designer from Database**.
   * Укажите подключение к базе данных UserManagementDB.
   * Импортируйте таблицы Users и SecretQuestions.

**4. Сохранение данных пользователя**

1. **Класс для работы с базой:**
   * Создайте класс UserService:

csharp

Копировать код

using System;

using System.Linq;

public class UserService

{

private readonly UserManagementDBEntities \_db;

public UserService()

{

\_db = new UserManagementDBEntities();

}

public bool Login(string username, string password)

{

string hashedPassword = HashPassword(password);

var user = \_db.Users.FirstOrDefault(u => u.UserName == username && u.PasswordHash == hashedPassword);

return user != null;

}

public void Register(string username, string password, string email, int secretQuestionId)

{

string hashedPassword = HashPassword(password);

var newUser = new User

{

UserName = username,

PasswordHash = hashedPassword,

Email = email,

SecretQuestionID = secretQuestionId

};

\_db.Users.Add(newUser);

\_db.SaveChanges();

}

private string HashPassword(string password)

{

using (var sha256 = System.Security.Cryptography.SHA256.Create())

{

var hashedBytes = sha256.ComputeHash(System.Text.Encoding.UTF8.GetBytes(password));

return BitConverter.ToString(hashedBytes).Replace("-", "").ToLower();

}

}

}

1. **Обработка авторизации:**
   * Добавьте в форму код для кнопки **Войти**:

csharp

Копировать код

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

{

UserService userService = new UserService();

bool isValid = userService.Login(txtUsername.Text, txtPassword.Text);

if (isValid)

{

MessageBox.Show("Login successful!");

}

else

{

lblError.Text = "Invalid username or password.";

lblError.Visible = true;

}

}

**5. Создание отчета**

1. **Требования к оформлению:**
   * Шрифт: Times New Roman, 12.
   * Межстрочный интервал: 1,5.
   * Отступ слева: 1,5 см.
   * Выравнивание: по ширине.
2. **Структура отчета:**
   * **Название работы:** Лабораторная работа №13.
   * **Цель работы:** Закрепление практических навыков работы с Visual Studio и MS SQL Server.
   * **Ход работы:**
     + Создание базы данных.
     + Разработка приложения.
     + Создание модели данных.
     + Интеграция и тестирование.
   * **Вывод:** Приложение успешно разработано и протестировано.
   * **Ответы на вопросы:**
     + **Что такое интеграция?** Интеграция — это процесс объединения модулей системы для обеспечения их совместной работы.
     + **Как реализовывается создание базы данных?** База данных создается в MS SQL Server Management Studio путем создания таблиц, связей и заполнения данными.
     + **Что такое модель данных?** Модель данных — это объектное представление данных, используемое для работы с базой данных.
     + **Как передаются данные в MSSQL Server?** Через запросы SQL или ORM, такие как Entity Framework.