

ЛАБОРАТОРНАЯ РАБОТА № 4

Разработка базы данных и подключение desktop-клиента логгирования исключений в распределенной информационной системе

Цель работы: Изучить принципы построения приложений, являющихся источниками данных в распределенных информационных системах, научиться использовать методы проектирования приложений доступа к данным, базируясь на принципах Code-First. Освоить механизмы Entity Framework для проектирования desktop-клиентов РИС.

Порядок выполнения работы:

- 1) Изучить материалы методических рекомендаций к ЛР 4 по дисциплине «Распределенные информационные системы», размещенные в личном кабинете.
- 2) В соответствии с заявленной в методических рекомендациях функциональностью, разработать клиент-серверную БД и подключить к ней desktop-клиент, являющийся комплексным проектом логгирования исключений, возникающих в ходе эксплуатации различных форм проекта поддержки лабораторных работ по дисциплинам в области информационной безопасности.
- 3) Desktop-клиент в зависимости от действий пользователя должен уметь переходить в режим записи исключений в базу данных системы AirLogger (сущность UserException) в реальном масштабе времени.
- 4) Разработать дополнительный функционал проекта в соответствии с индивидуальным вариантом.
- 3) Оформить отчет по лабораторной работе, содержащий: титульный лист, название, номер и цель работы, постановку задачи, алгоритм решения для каждого программного модуля, листинг программных модулей, распечатку результатов, распечатку изображения форм, используемых в программе

Задания к лабораторной работе №4

Вариант 1

Перенести реализацию основных операций по работе с набором данных (CRUD – create, read, update, delete) из модуля специализированной формы (form6) в контекст данных, оставив на данной форме только функции отображения данных в dataGridView.

<https://metanit.com/sharp/entityframeworkcore/2.1.php>

Вариант 2

Реализовать конфигурацию подключений к базе данных с помощью JSON-файла. <https://metanit.com/sharp/entityframeworkcore/2.2.php>

Вариант 3

Реализовать логгирование операций по работе с базой данных с помощью собственного интерфейса-провайдера. Лог операций выводить на главную форму проекта (textBox или richTextBox) и записывать в базу данных системы AirLogger.

<https://metanit.com/sharp/entityframeworkcore/2.12.php>

Вариант 4

Реализовать глобальную установку логгирования в контексте данных с настройкой логгирования. Лог операций выводить на главную форму проекта (textBox или richTextBox) и записывать в базу данных системы AirLogger.

<https://metanit.com/sharp/entityframeworkcore/2.12.php>

Вариант 5

Реализовать логгирование операций по работе с базой данных, используя встроенный провайдер Microsoft. Лог операций выводить на главную форму проекта (textBox или richTextBox) и записывать в базу данных системы AirLogger.

<https://metanit.com/sharp/entityframeworkcore/2.12.php>

Вариант 6

Перенести реализацию основных операций по работе с набором данных (CRUD – create, read, update, delete) из модуля специализированной формы (form6) в контекст данных, оставив на данной форме только функции отображения данных в dataGridView.

<https://metanit.com/sharp/entityframeworkcore/2.1.php>

Вариант 7

Реализовать конфигурацию подключений к базе данных с помощью JSON-файла. <https://metanit.com/sharp/entityframeworkcore/2.2.php>

Вариант 8

Реализовать логгирование операций по работе с базой данных с помощью собственного интерфейса-провайдера. Лог операций выводить на главную форму проекта (textBox или richTextBox) и записывать в базу данных системы AirLogger.

<https://metanit.com/sharp/entityframeworkcore/2.12.php>

Вариант 9

Реализовать глобальную установку логгирования в контексте данных с настройкой логгирования. Лог операций выводить на главную форму проекта (textBox или richTextBox) и записывать в базу данных системы AirLogger.

<https://metanit.com/sharp/entityframeworkcore/2.12.php>

Вариант 10

Реализовать логгирование операций по работе с базой данных, используя встроенный провайдер Microsoft. Лог операций выводить на главную форму проекта (textBox или richTextBox) и записывать в базу данных системы AirLogger.

<https://metanit.com/sharp/entityframeworkcore/2.12.php>

Вариант 11

Перенести реализацию основных операций по работе с набором данных (CRUD – create, read, update, delete) из модуля специализированной формы (form6) в контекст данных, оставив на данной форме только функции отображения данных в dataGridView.

<https://metanit.com/sharp/entityframeworkcore/2.1.php>

Вариант 12

Реализовать конфигурацию подключений к базе данных с помощью JSON-файла. <https://metanit.com/sharp/entityframeworkcore/2.2.php>

Вариант 13

Реализовать логгирование операций по работе с базой данных с помощью собственного интерфейса-провайдера. Лог операций выводить на главную форму проекта (textBox или richTextBox) и записывать в базу данных системы AirLogger.

<https://metanit.com/sharp/entityframeworkcore/2.12.php>

Вариант 14

Реализовать глобальную установку логгирования в контексте данных с настройкой логгирования. Лог операций выводить на главную форму проекта (textBox или richTextBox) и записывать в базу данных системы AirLogger.

<https://metanit.com/sharp/entityframeworkcore/2.12.php>

Вариант 15

Реализовать логгирование операций по работе с базой данных, используя встроенный провайдера Microsoft. Лог операций выводить на главную форму проекта (textBox или richTextBox) и записывать в базу данных системы AirLogger.

<https://metanit.com/sharp/entityframeworkcore/2.12.php>

Вариант 16

Перенести реализацию основных операций по работе с набором данных (CRUD – create, read, update, delete) из модуля специализированной формы (form6) в контекст данных, оставив на данной форме только функции отображения данных в dataGridView.

<https://metanit.com/sharp/entityframeworkcore/2.1.php>