

Лабораторная работа №45

Создание приложения с БД для чтения данных

1Цель работы

1.1 Научиться создавать приложения для чтения данных из БД.

2Литература

2.1 <https://metanit.com/sharp/adonetcore> гл.1,3

3Подготовка к работе

3.1 Повторить теоретический материал (см. п.2).

3.2 Изучить описание лабораторной работы.

4Основное оборудование

4.1 Персональный компьютер.

5Задание

Создать решение, которое будет использоваться в нескольких лабораторных для работы с БД.

5.1 Создание слоя доступа к данным и формирование строки подключения

5.1.1 Создать набор проектов:

- проект библиотеки для работы с БД MSSQL (пакет Microsoft.Data.SqlClient),
- проект библиотеки для работы с БД Sqlite (пакет Microsoft.Data.Sqlite),
- консольное или оконное клиентское приложение (для проверки работы запросов и переключения между СУБД).

5.1.2 В каждом проекте библиотеки создать статический класс DataAccessLayer (DAL), в котором будет храниться логика работы с БД.

Добавить в классы DAL:

- открытые строковые автосвойства для хранения настроек подключения к БД (для MSSQL: названия сервера, БД, логина и пароля пользователя, для Sqlite: имя файла БД), присвоить им значения по умолчанию,
- открытое строковое свойство ConnectionString только на чтение, возвращающее строку подключения к БД (будет использоваться в остальных методах). Для реализации использовать значения настроек подключения и ПровайдерConnectionStringBuilder.

5.1.3 Проверить работу созданного свойства, вызвав его в приложении, полученную строку отобразить на экране.

5.2 Создание метода для выборки одного значения

5.2.1 Добавить в классы DAL открытый метод GetScalarValue, который:

- принимает строку с текстом SQL-команды,
- выполняет SQL-команду,
- возвращает полученное значение как object.

Для реализации использовать ПровайдерCommand.ExecuteScalar().

5.2.2 Проверить работу созданного метода, вызвав его в приложении.

Текст SQL-команды должен вводиться пользователем, полученное значение должно отображаться на экране.

5.2.3 При вызове обработать исключения.

5.3 Создание метода для выполнения команды на выборку с параметрами

5.3.1 Добавить в классы DAL открытый метод `GetGamesByPrice`, который возвращает информацию об играх с ценой из диапазона min-max (передаются в параметрах метода) как `List<Game>`.

`Game` – пользовательский класс с автосвойствами `int Id`, `string Title`, `double Price`.

Для реализации использовать `ПровайдерCommand.ExecuteReader()`.

Перед выполнением добавить в команду параметры.

5.3.2 Проверить работу созданного метода, вызвав его в приложении.

Результат выполнения должен отображаться на экране.

5.3.3 При вызове обработать исключения.

5.4 Создание свойства для выборки набора значений с использованием `Dapper`

5.4.1 Добавить в классы DAL пакет `Dapper` и открытое свойство на чтение `Games`, которое возвращает информацию об играх как `List<Game>`.

`Game` – пользовательский класс с автосвойствами `int Id`, `string Title`, `double Price`.

Для реализации использовать `Dapper`.

5.4.2 Проверить работу созданного метода, вызвав его в приложении.

Результат выполнения должен отображаться на экране.

5.4.3 При вызове обработать исключения.

6 Порядок выполнения работы

6.1 Запустить MS Visual Studio и создать на C# оконное приложение.

6.2 Выполнить все задания из п.5. При разработке считать, что пользователь ввел данные требуемого типа, остальные возможные ошибки обрабатывать. Исключения должны обрабатываться в основной программе. При выполнении заданий использовать минимально возможное количество команд и переменных и выполнять форматирование и рефакторинг кода.

6.3 Ответить на контрольные вопросы.

7 Содержание отчета

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

8 Контрольные вопросы

8.1 Каково назначение элемента `SqlConnection`?

8.2 Как считать из БД одно значение?

8.3 Каково назначение элемента `SqlDataReader`?

8.4 Какие пространства имен требуется подключить для реализации подключения к СУБД MS SQL Server?