

### 30 Работа с базами данных. Технология ADO.NET

Задание 1. В любой СУБД создайте базу данных DBTur\_firm. Создайте .Net приложение (консольное, WinForm, WPF любое на выбор) и выполните подключение к разработанной базе данных. Создайте и выполните SQL запросы:

Выберите все туры;

Удалите тур;

Добавьте туриста;

Измените туриста.

Листинг программы:

```
namespace Task
{
    public partial class MainWindow : Window
    {
        private string connectionString;

        public MainWindow()
        {
            InitializeComponent();

            // Получение пути к файлу базы данных
            string databasePath =
                System.IO.Path.Combine(AppDomain.CurrentDomain.BaseDirectory,
                "DBTur_firm.accdb");

            // Создание строки подключения
            connectionString = $"Provider=Microsoft.ACE.OLEDB.12.0;Data
            Source={databasePath}";
        }

        private void SelectAllTours_Click(object sender, RoutedEventArgs e)
        {
            // Запрос для выборки всех записей из таблицы "Туры"
            string query = "SELECT * FROM Туры";

            try
            {
                // Создание подключения к базе данных
```

					УП 2-40 01 01.37ТП.227.23.30								
Изм.	Лист	№ док	Подпись	Дата									
Разраб.		Сорокина Е.А.			Работа с базами данных.  Технология ADO.NET				Лит	Лист	Листов		
Проверил.		Новик А.И.									132		
									Гродненский ГКТТид				
Н.контр.													
Утвердил.													

```

using (OleDbConnection connection = new OleDbConnection(connectionString))
{
    // Открытие подключения
    connection.Open();

    // Создание команды с запросом и подключением
    using (OleDbCommand command = new OleDbCommand(query,
connection))
    {
        // Создание объекта для чтения данных
        using (OleDbDataReader reader = command.ExecuteReader())
        {
            // Очистка списка перед загрузкой новых данных
            TourList.Items.Clear();

            // Чтение данных и добавление их в список
            while (reader.Read())
            {
                int tourID = reader.GetInt32(0); // Код тура находится в
первом столбце
                string tourName = reader.GetString(1); // Название тура
находится во втором столбце
                string tourPrice = reader.GetDecimal(2).ToString(); // Цена
тура находится в третьем столбце

                // Создание строки, содержащей все данные о туре
                string tourData = $"ID тура: {tourID} - Название:
{tourName} - Цена: {tourPrice}";

                // Добавление строки с данными о туре в список
                TourList.Items.Add(tourData);
            }
        }
    }
}
catch (Exception ex)
{
    // Обработка ошибок при выполнении запроса или
открытии/закрытии подключения
    MessageBox.Show($"Ошибка: {ex.Message}");
}
}

```

Изм.	Лист	№ док	Подпись	Дата

```

private void DeleteTour_Click(object sender, RoutedEventArgs e)
{
    if (int.TryParse(txtTourIdToDelete.Text, out int tourId))
    {
        try
        {
            using (OleDbConnection connection = new
OleDbConnection(connectionString))
            {
                string query = "DELETE FROM Туры WHERE [Код тура] =
@TourId";
                OleDbCommand command = new OleDbCommand(query,
connection);
                command.Parameters.AddWithValue("@TourId", tourId);

                connection.Open();
                command.ExecuteNonQuery();

                MessageBox.Show("Тур успешно удален.");
            }
        }
        catch (Exception ex)
        {
            MessageBox.Show("Ошибка при выполнении запроса: " +
ex.Message);
        }
    }
    else
    {
        MessageBox.Show("Некорректный ID тура.");
    }
}

private void AddTourist_Click(object sender, RoutedEventArgs e)
{
    string name = txtTouristName.Text;
    string surname = txtTouristSurname.Text;
    string patronymic = txtTouristPatronymic.Text;

    try
    {
        using (OleDbConnection connection = new
OleDbConnection(connectionString))

```

Изм.	Лист	№ док	Подпись	Дата

```

        {
            string query = "INSERT INTO Туристы (Фамилия, Имя,
Отчество) VALUES (@Surname, @Name, @Patronymic)";
            OleDbCommand command = new OleDbCommand(query,
connection);

            command.Parameters.AddWithValue("@Surname", surname);
            command.Parameters.AddWithValue("@Name", name);
            command.Parameters.AddWithValue("@Patronymic", patronymic);

            connection.Open();
            command.ExecuteNonQuery();

            MessageBox.Show("Турист успешно добавлен.");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show("Ошибка при выполнении запроса: " +
ex.Message);
    }
}

private void UpdateTourist_Click(object sender, RoutedEventArgs e)
{
    if (int.TryParse(txtTouristIdToUpdate.Text, out int touristId))
    {
        string newName = txtNewTouristName.Text;
        string newSurname = txtNewTouristSurname.Text;
        string newPatronymic = txtNewTouristPatronymic.Text;

        try
        {
            using (OleDbConnection connection = new
OleDbConnection(connectionString))
            {
                string query = "UPDATE Туристы SET Фамилия =
@NewSurname, Имя = @NewName, Отчество = @NewPatronymic WHERE [Код
туриста] = @TouristId";
                OleDbCommand command = new OleDbCommand(query,
connection);

                command.Parameters.AddWithValue("@NewSurname",
newSurname);

                command.Parameters.AddWithValue("@NewName", newName);

```

```

newPatronymic);

command.Parameters.AddWithValue("@NewPatronymic",
command.Parameters.AddWithValue("@TouristId", touristId);

connection.Open();
command.ExecuteNonQuery();

MessageBox.Show("Турист успешно изменен.");
}
}
catch (Exception ex)
{
    MessageBox.Show("Ошибка при выполнении запроса: " +
ex.Message);
}
}
else
{
    MessageBox.Show("Некорректный ID туриста.");
}
}
}
}
}

```

Анализ результатов:

Выбрать все туры

**Удалите тур:**

ID тура:

**Добавьте туриста:**

Имя:  Фамилия:  Отчество:

**Измените туриста:**

ID туриста:  Новое имя:  Новая фамилия:  Новое отчество:

ID тура: 2 - Название: Утро с пандами - Цена: 2000  
ID тура: 3 - Название: Исторический тур по Риму - Цена: 2500  
ID тура: 4 - Название: Сафари в национальных парках Кении - Цена: 4500  
ID тура: 5 - Название: Путешествие по Японии - Цена: 3900  
ID тура: 6 - Название: Романтический уикенд в Замке Сказок - Цена: 2800  
ID тура: 7 - Название: Тур по историческим достопримечательностям Египта - Цена: 2800

Рисунок 30.1 – Результат работы программы

Изм.	Лист	№ док	Подпись	Дата

УП 2-40 01 01.37ТП.227.23.30

Лист  
136