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

Задание 1. Создать базу данных, по примеру. Создать приложение, для демонстрации работы с базой данных Листинг программы:

static async Task Main(string[] args)

{

using (SqlConnection sqlConnection = new SqlConnection(ConfigurationManager.ConnectionStrings["Test"].ConnectionString))

{

sqlConnection.Open();

if(sqlConnection.State == System.Data.ConnectionState.Open)

{

Console.WriteLine("Подключение успешно!\n");

SqlCommand getAllTours = new SqlCommand("SELECT \* FROM Tours", sqlConnection);

Console.WriteLine("1. Выводим все туры...");

await PrintTours(getAllTours);

SqlCommand deleteLastTour = new SqlCommand("DELETE FROM Tours WHERE ID=(SELECT MAX(ID) FROM Tours)", sqlConnection);

Console.WriteLine("\nУдаляем последний тур...\n");

deleteLastTour.ExecuteNonQuery();

Console.WriteLine("2. Выводим все туры ещё раз...");

await PrintTours(getAllTours);

Console.WriteLine("3. Выводим туристов и добавляем нового, затем снова выводим туристов..");

SqlCommand getAllTourists = new SqlCommand("SELECT \* FROM Tourists", sqlConnection);

await PrintTourists(getAllTourists);

SqlCommand addNewTourist = new SqlCommand("INSERT INTO [Tourists] (Name, Surname, Last\_Name) VALUES(N'Максим', N'Олехнович', N'Александрович')", sqlConnection);

addNewTourist.ExecuteNonQuery();

Console.WriteLine("\nДобавляем туриста Максима...\n");

await PrintTourists(getAllTourists);

Console.WriteLine("4. Изменяем туриста Максима на Андрея и выводим их заново..");

SqlCommand deleteTourist = new SqlCommand("UPDATE Tourists SET Name = N'Андрей' WHERE Name = N'Максим'", sqlConnection);

deleteTourist.ExecuteNonQuery();

Console.WriteLine("\nТурист изменён...\n");

await PrintTourists(getAllTourists);

}

}

Console.WriteLine("Подключение разорвано!");

}

public static async Task PrintTourists(SqlCommand command)

{

using (SqlDataReader dataReader = await command.ExecuteReaderAsync())

{

if (dataReader.HasRows)

{

string columnName1 = dataReader.GetName(0);

string columnName2 = dataReader.GetName(1);

string columnName3 = dataReader.GetName(2);

string columnName4 = dataReader.GetName(3);

Console.WriteLine("{0,-4} | {1, -15} | {2, -15} | {3, -20}", columnName1, columnName2, columnName3, columnName4);

while (await dataReader.ReadAsync())

{

object id = dataReader.GetValue(0);

object name = dataReader.GetValue(1);

object surname = dataReader.GetValue(2);

object lastname = dataReader.GetValue(3);

Console.WriteLine("{0,-4} | {1, -15} | {2, -15} | {3, -20}", id, name, surname, lastname);

}

}

}

}

public static async Task PrintTours(SqlCommand command)

{

using (SqlDataReader dataReader = await command.ExecuteReaderAsync())

{

if (dataReader.HasRows)

{

string columnName1 = dataReader.GetName(0);

string columnName2 = dataReader.GetName(1);

string columnName3 = dataReader.GetName(2);

string columnName4 = dataReader.GetName(3);

Console.WriteLine("{0,-6} | {1, -15} | {2, -10} | {3, -100}", columnName1, columnName2, columnName3, columnName4);

while (await dataReader.ReadAsync())

{

object id = dataReader.GetValue(0);

object name = dataReader.GetValue(1);

object price = dataReader.GetValue(2);

object info = dataReader.GetValue(3);

Console.WriteLine("{0,-6} | {1, -15} | {2, -10:f2} | {3, -100}", id, name, price, info);

}

}

}

}

Анализ результатов:

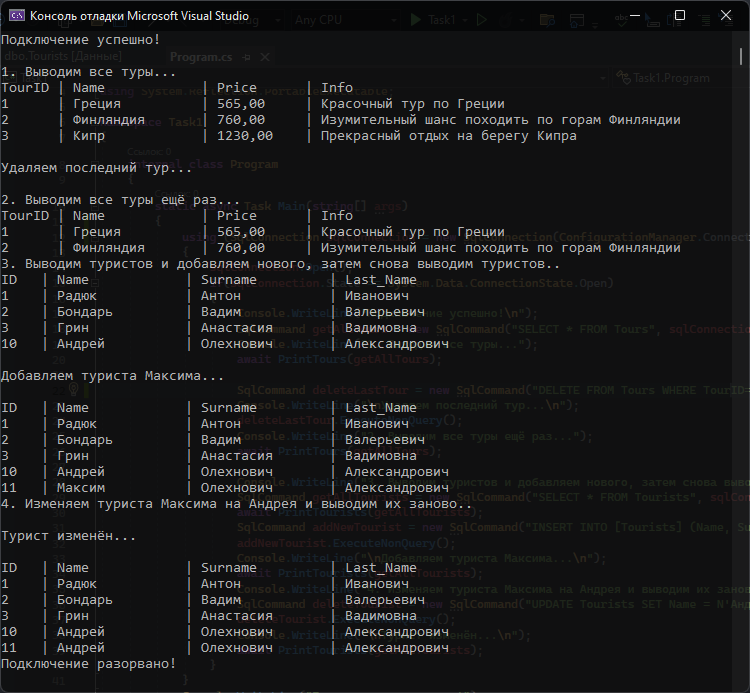


Рисунок 30.1 – Результат работы программы