Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования «Ивановский государственный энергетический университет имени В.И. Ленина»

Факультет информатики и вычислительной техники

Кафедра программного обеспечения компьютерных систем

Курсовой проект

по дисциплине «Базы данных» на тему:

«Проектирование и разработка базы данных

информационной системы «Аквапарк»

Выполнил:

ст. гр. 2-42 Потонов Г.Е.

Проверила:

Ратманова И.Д.

Иваново 2021

# Введение

**Цель работы:** спроектировать и разработать базу данных тестовой информационной системы под названием «Аквапарк».

**Описание:** аквапарк предлагает своим клиентам хорошо провести время в трех зонах: водные аттракционы, сауны и массажный городок. Зоны различаются стоимостью одной минуты пребывания в них: 5, 7 и 9 рублей соответственно. Местоположение клиента отслеживается с при помощи электронных браслетов. По уходу из аквапарка клиенту выставляется счет на основе проведенного им в аквапарке времени c точностью до минуты с учетом пребывания в разных зонах. При втором посещении аквапарка клиенту приписывается статус постоянного, и для него теперь действует скидка 10%.

# 1 Бизнес-понятия, бизнес-события и концептуальная модель БД

**Бизнес-понятия и их атрибуты:** клиент, электронный браслет, зона

**Бизнес-события:** оплата

**Справочники:** пол, флаг постоянного покупателя

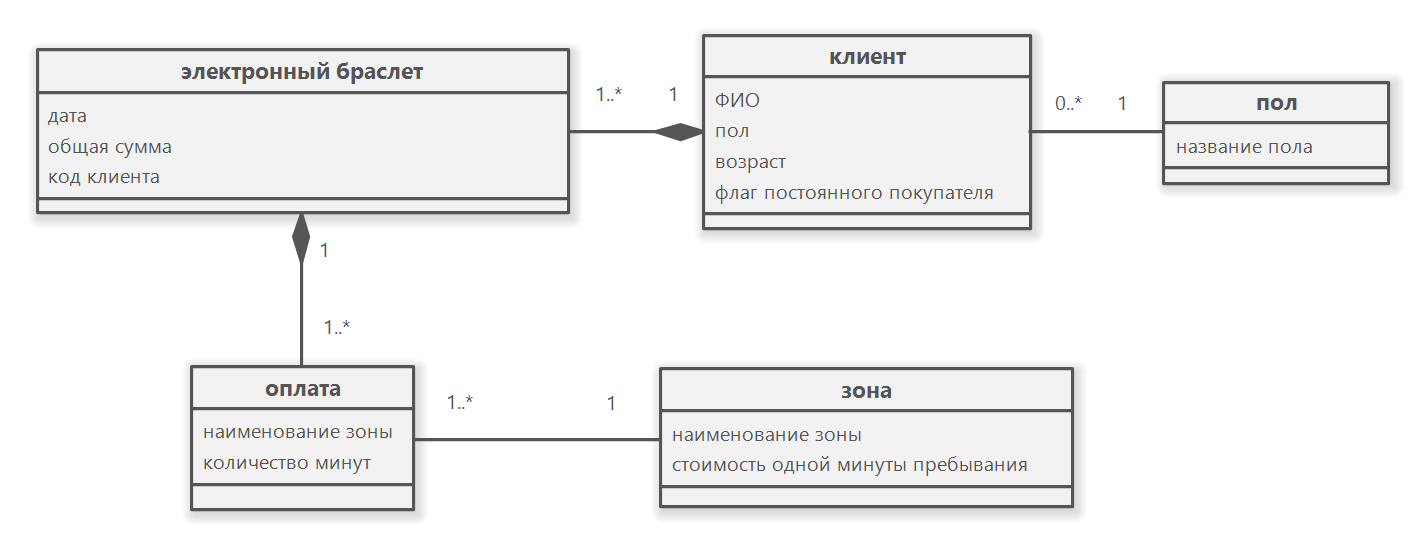


Рис. 1. Концептуальная модель предметной области

# 2 Логическая модель БД

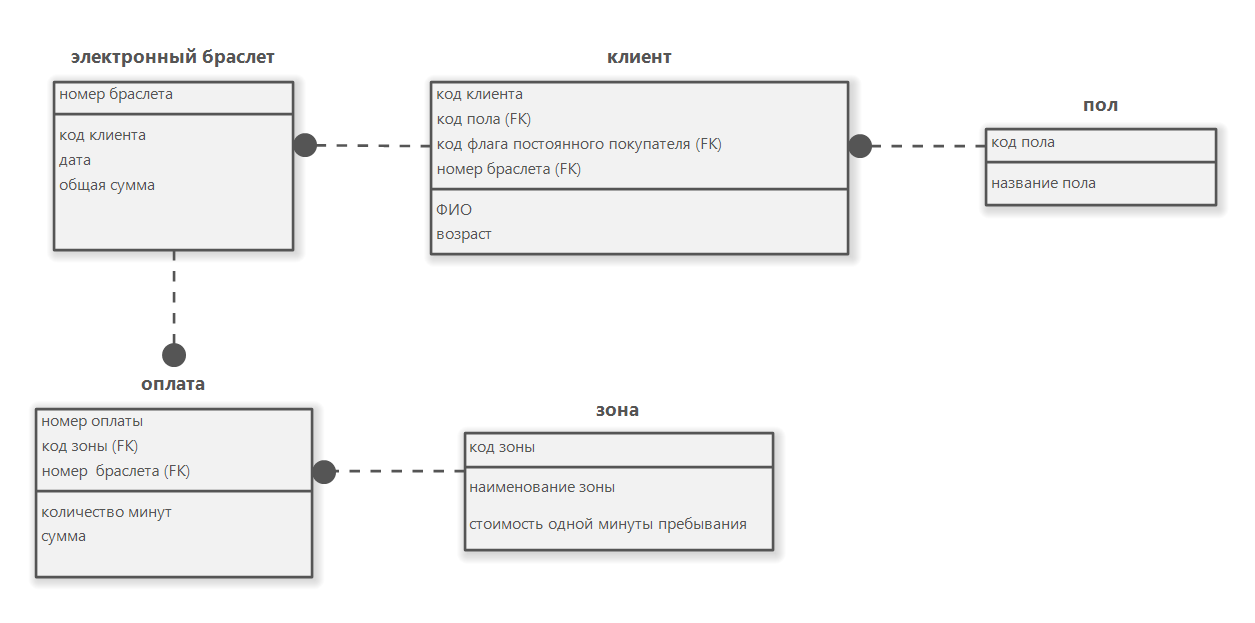
­­­

Рис. 2. Логическая модель реляционной базы данных

# 3 Физическая модель БД и содержание таблиц

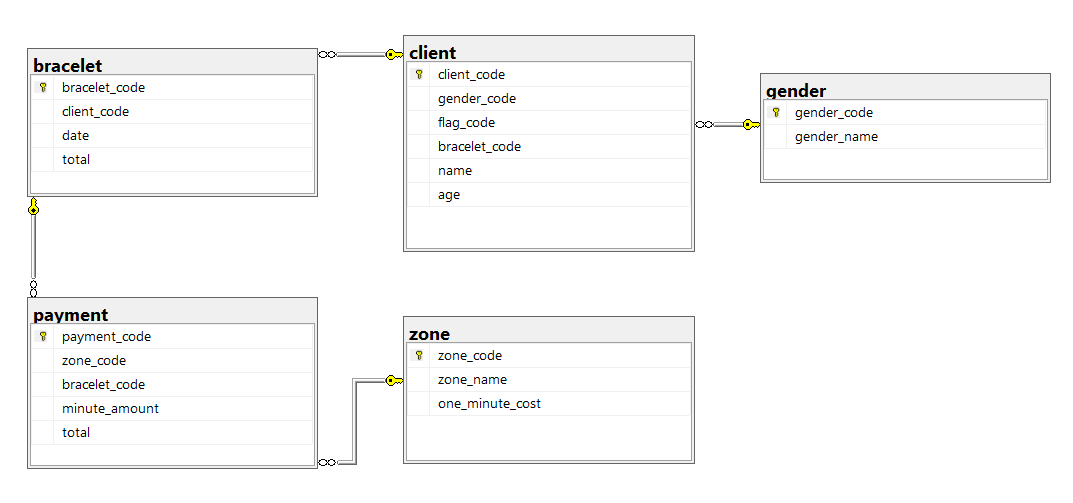


Рис. 3. Физическая диаграмма реляционной базы данных

**Содержание таблиц**

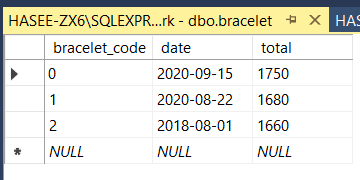


Рис. 3.1. Таблица “электронный браслет”

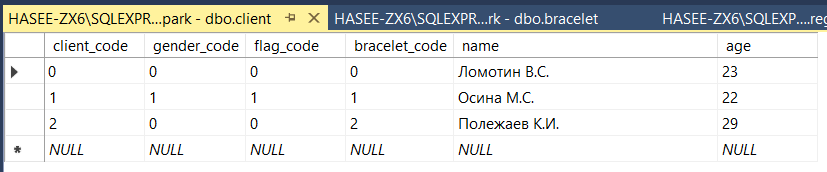


Рис. 3.2. Таблица “клиент”

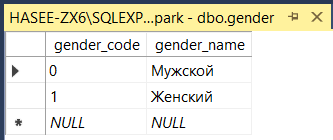


Рис. 3.3. Таблица “пол”

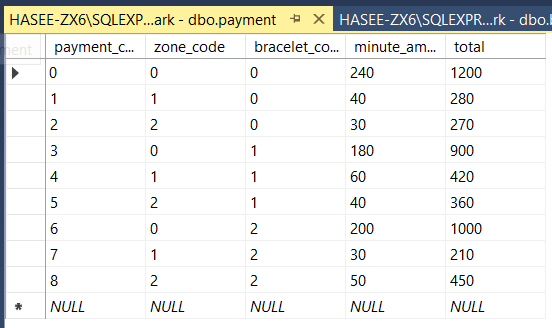


Рис. 3.4. Таблица “оплата”

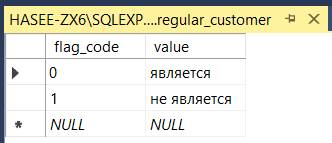


Рис. 3.5. Таблица “флаг постоянного покупателя”

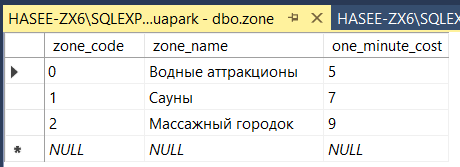


Рис. 3.6. Таблица “зона”

# 4 Ведение базы данных. Запросы на модификацию данных

**Insert**

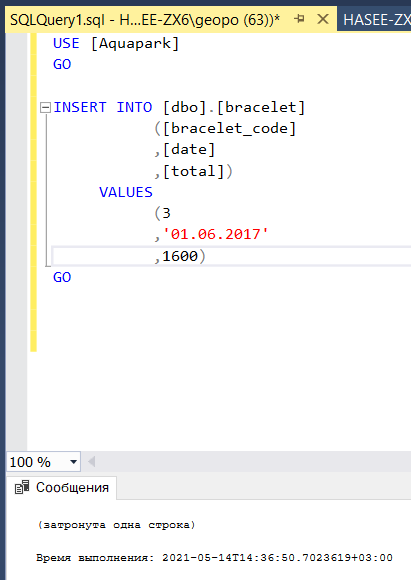


Рис. 4. Запрос Insert для таблицы “электронный браслет”

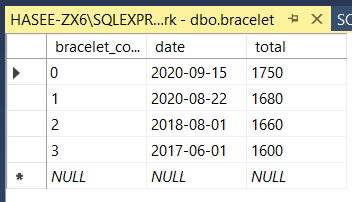


Рис. 4. Результат выполнения запроса Insert

**Update**

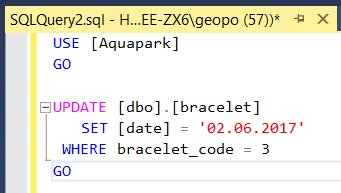


Рис. 4. Запрос Update для таблицы “электронный браслет”

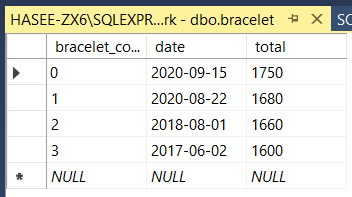


Рис. 4. Результат выполнения запроса Update

**Delete**

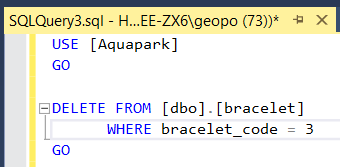


Рис. 4. Запрос Delete для таблицы “электронный браслет”

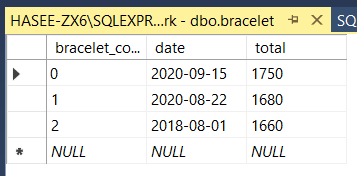


Рис. 4. Результат выполнения запроса Delete

# 5 Запросы к БД

**Проекция (case)**

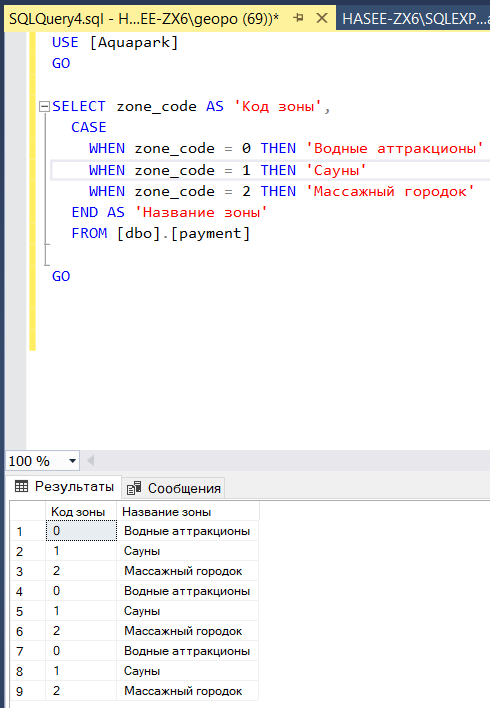


Рис. 5.1. Разделение оплат по зонам

**Фильтрация (where)**

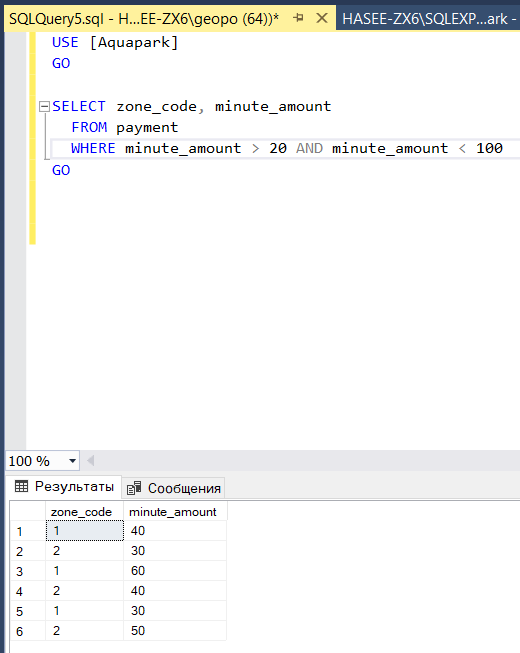


Рис. 5.2. Определение зон, время пребывания в которых находится в интервале от 20 до 100

**Агрегация (group by)**

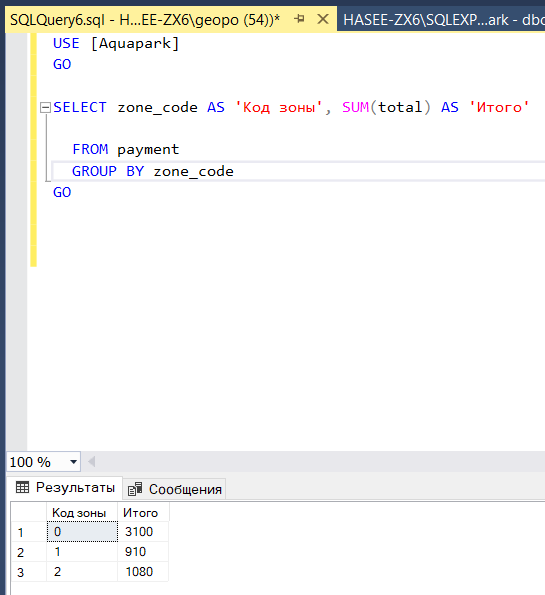


Рис. 5.3. Вывод общей суммы, полученной с каждой зоны

**Навигация по базе (from)**

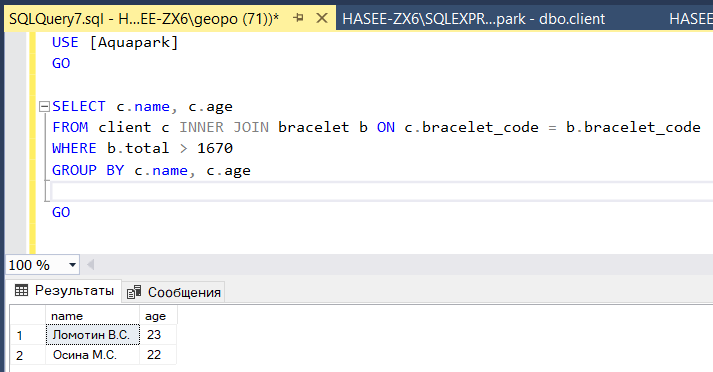


Рис. 5.4. Выбор клиентов, потративших больше 1670 рублей

**Подзапросы**

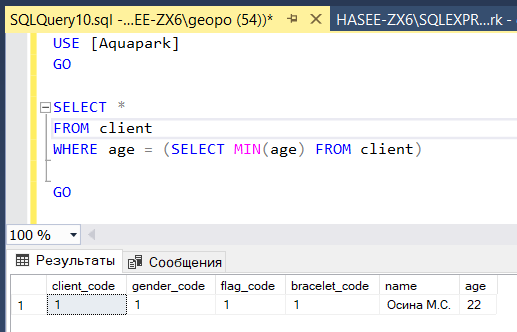


Рис. 5.5. Нахождение самого молодого клиента

# 6 Бизнес-логика приложения

**Хранимые процедуры**

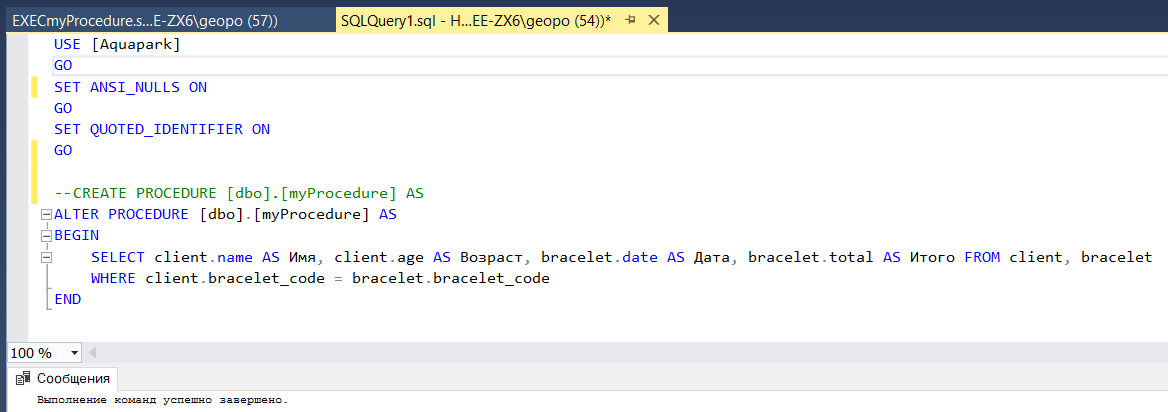


Рис. 6.1.1. Хранимая процедура myProcedure, выводящая полезную информацию из таблиц client и bracelet

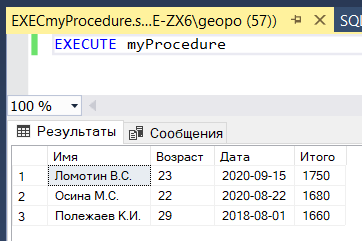


Рис. 6.1.2. Выполнение процедуры myProcedure

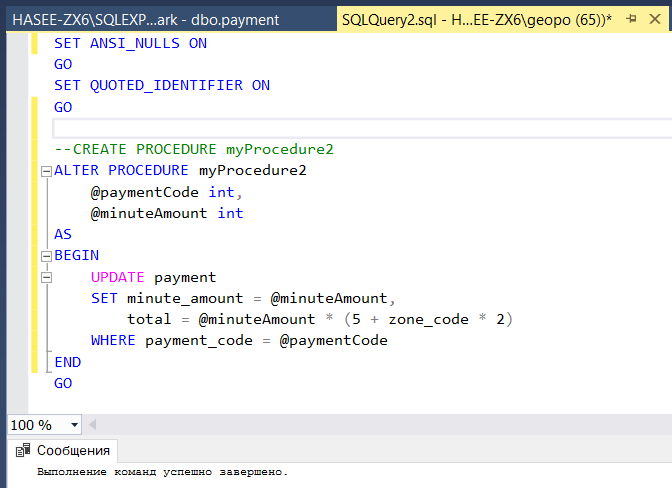


Рис. 6.2.1. Хранимая процедура myProcedure2 для изменения данных о платеже, принимающая 2 параметра

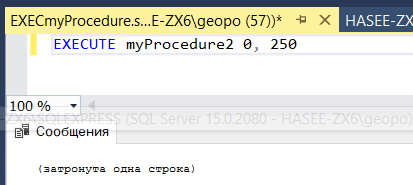


Рис. 6.2.2. Выполнение процедуры myProcedure2

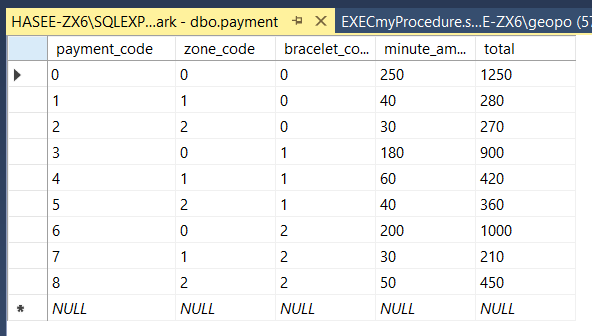


Рис. 6.2.2. Результат выполнения процедуры myProcedure2

**Триггеры**

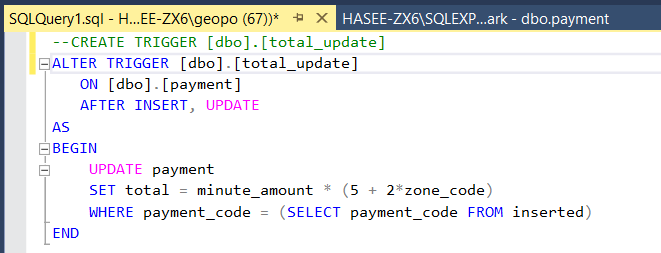


Рис. 6.3.1. Триггер для автоматического просчета стоимости

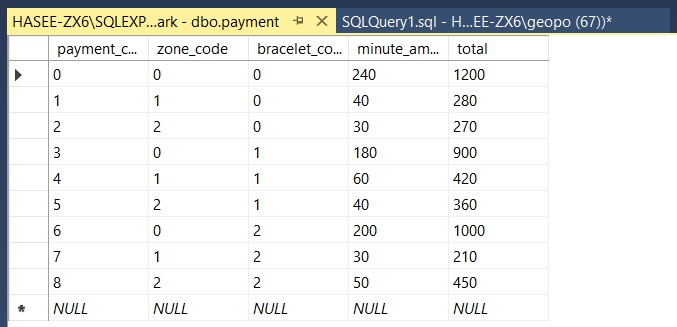


Рис. 6.3.2. При изменении в первой строчке *minute\_amount* с 250 на 240 значение *total* в той же строке меняется с 1250 на 1200

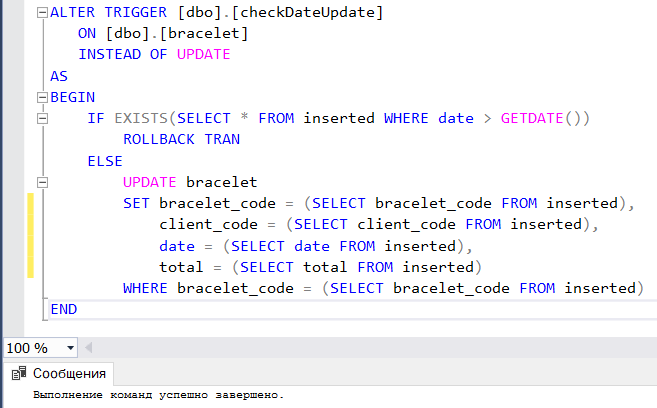


Рис. 6.3.1. Триггер, проверяющий корректность даты при ее изменении.

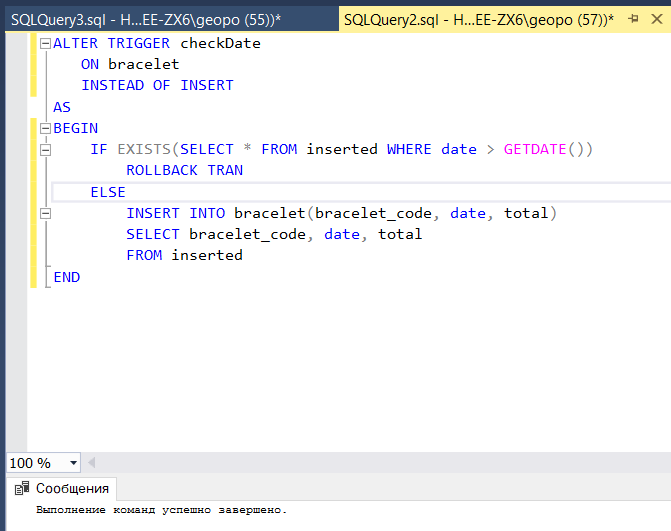


Рис. 6.3.2. Триггер, проверяющий корректность даты, при добавлении новой записи

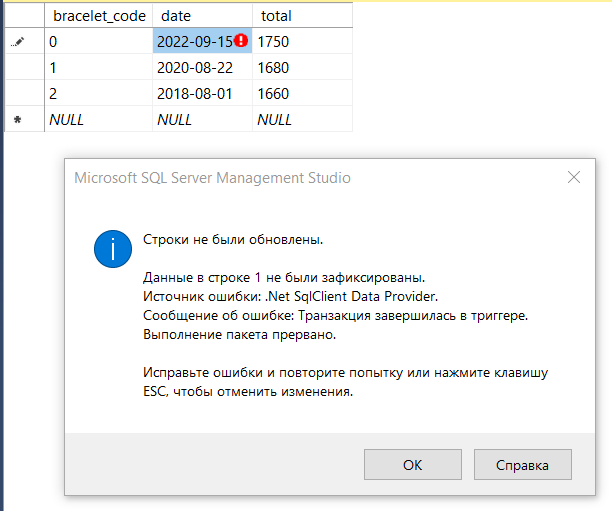


Рис. 6.3.3. Результат работы триггера

# 7 Логическая модель аналитической витрины

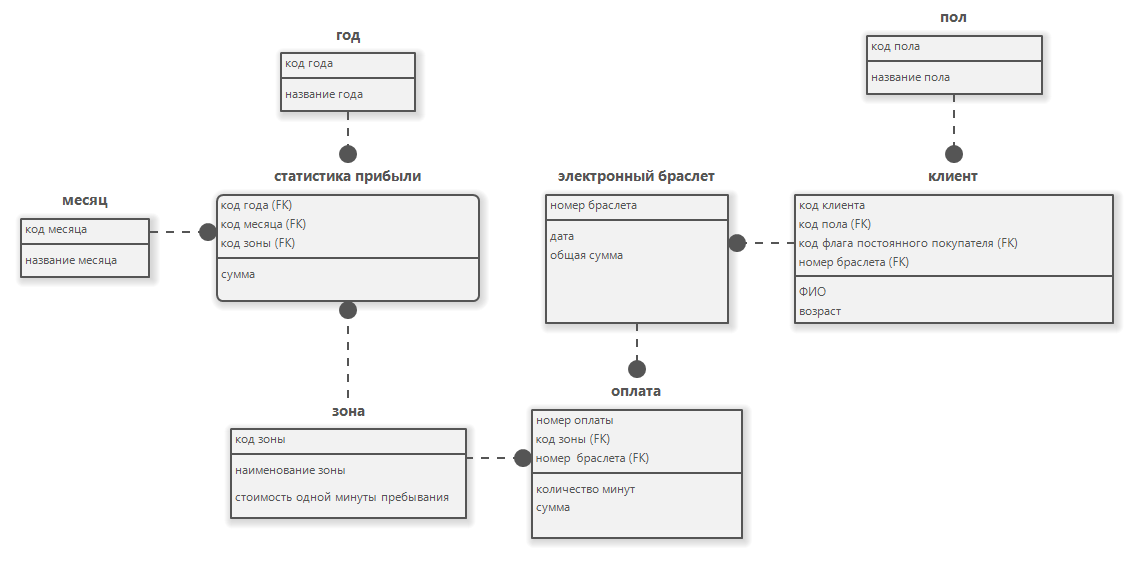


Рис. 7. Логическая модель аналитической витрины ведения статистики прибыли

# 8 Клиентское приложение для выполнения запросов к базе данных с использованием технологии доступа к данным ADO.NET

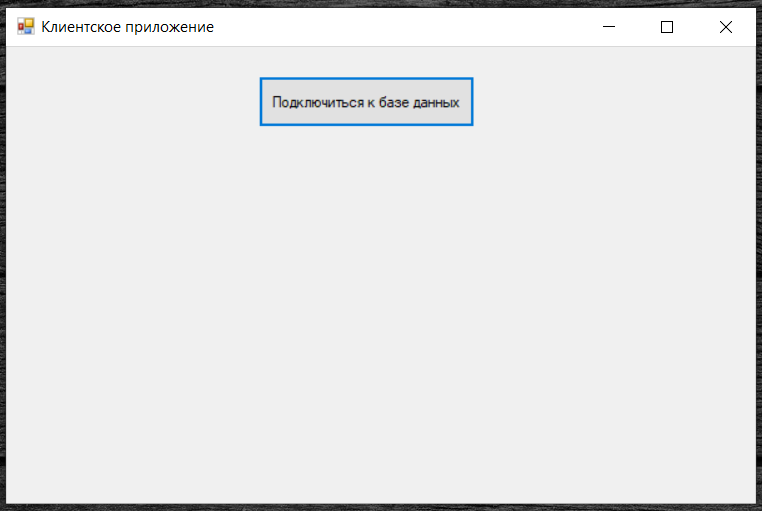
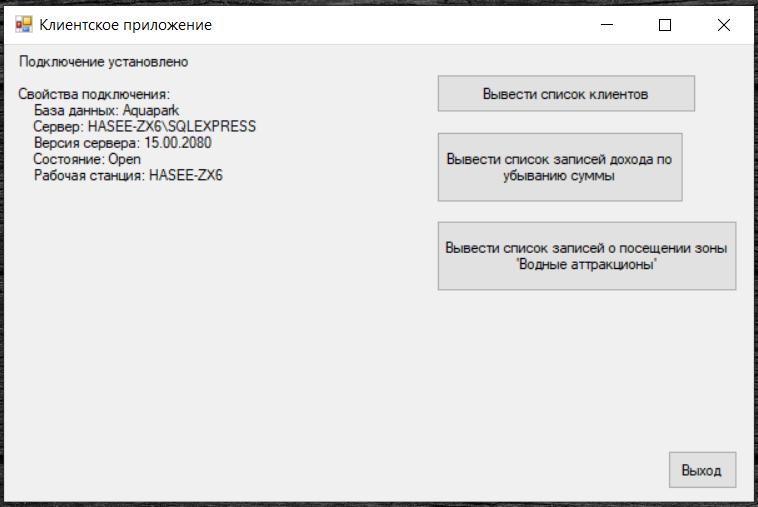
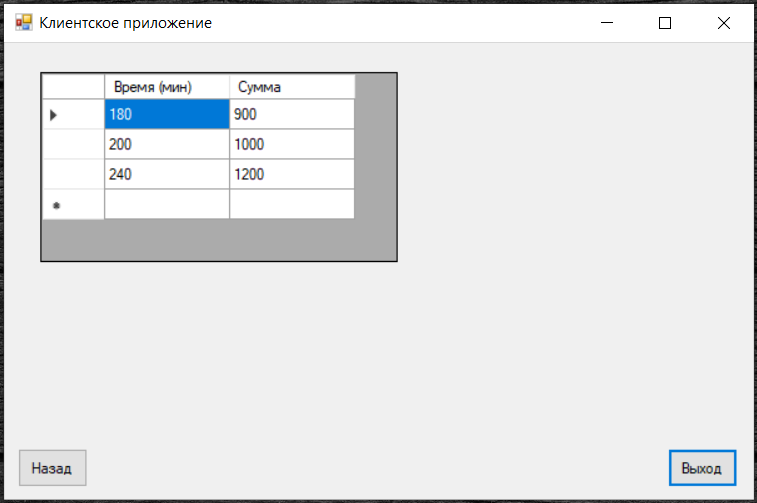


Рисунок 7.1 – Окно при старте программы

Рисунок 7.2 – Главное меню

Рисунок 7.3 – Пример запроса

## Код программы

using System;

using System.Data;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace БазаДанных

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

dataGridView1.AutoSize = true;

button1.Visible = true;

button2.Visible = false;

button3.Visible = false;

button4.Visible = false;

button6.Visible = false;

button7.Visible = false;

label1.Visible = false;

dataGridView1.Visible = false;

dataGridView1.AutoGenerateColumns = true;

}

string connectionString = @"Data Source=HASEE-ZX6\SQLEXPRESS; Initial Catalog=Aquapark; Integrated Security=True";

//запрос: вывести список клиентов

string sqlExpression2 = "SELECT gender\_code AS 'Пол (М-0, Ж-1)', flag\_code AS 'Постоянный клиент', name AS Имя, age AS Возраст FROM client;";

//запрос: вывести список записей дохода по убыванию суммы

string sqlExpression3 = "SELECT date AS Дата, total AS Сумма FROM bracelet ORDER BY total DESC;";

//запрос: вывести список записей о посещении зоны "Водные аттракционы"

string sqlExpression4 = "SELECT minute\_amount AS 'Время (мин)', total AS Сумма FROM payment WHERE zone\_code = 0 ORDER BY minute\_amount;";

private void button1\_Click(object sender, EventArgs e)

{

using (SqlConnection connection = new SqlConnection(connectionString))

{

connection.Open();

button1.Visible = false;

button2.Visible = true;

button3.Visible = true;

button4.Visible = true;

button7.Visible = true;

label1.Visible = true;

//Основные сведения о подключении

label1.Text = "Подключение установлено\n\n" +

"Свойства подключения:\n" +

" База данных: " + connection.Database + "\n" +

" Сервер: " + connection.DataSource + "\n" +

" Версия сервера: " + connection.ServerVersion + "\n" +

" Состояние: " + connection.State + "\n" +

" Рабочая станция: " + connection.WorkstationId + "\n";

}

}

private void button2\_Click(object sender, EventArgs e)

{

using (SqlConnection connection = new SqlConnection(connectionString))

{

button2.Visible = false;

button3.Visible = false;

button4.Visible = false;

button6.Visible = true;

button7.Visible = true;

label1.Visible = false;

dataGridView1.Visible = true;

DataSet ds = new DataSet();

SqlDataAdapter adapter = new SqlDataAdapter(sqlExpression2, connection);

SqlCommandBuilder commandBuilder = new SqlCommandBuilder(adapter);

adapter.Fill(ds, "client");

dataGridView1.DataSource = ds;

dataGridView1.DataMember = "client";

}

}

private void button3\_Click(object sender, EventArgs e)

{

using (SqlConnection connection = new SqlConnection(connectionString))

{

button2.Visible = false;

button3.Visible = false;

button4.Visible = false;

button6.Visible = true;

label1.Visible = false;

dataGridView1.Visible = true;

DataSet ds = new DataSet();

SqlDataAdapter adapter = new SqlDataAdapter(sqlExpression3, connection);

SqlCommandBuilder commandBuilder = new SqlCommandBuilder(adapter);

adapter.Fill(ds, "bracelet");

dataGridView1.DataSource = ds;

dataGridView1.DataMember = "bracelet";

}

}

private void button4\_Click(object sender, EventArgs e)

{

using (SqlConnection connection = new SqlConnection(connectionString))

{

button2.Visible = false;

button3.Visible = false;

button4.Visible = false;

button6.Visible = true;

label1.Visible = false;

dataGridView1.Visible = true;

DataSet ds = new DataSet();

SqlDataAdapter adapter = new SqlDataAdapter(sqlExpression4, connection);

SqlCommandBuilder commandBuilder = new SqlCommandBuilder(adapter);

adapter.Fill(ds, "payment");

dataGridView1.DataSource = ds;

dataGridView1.DataMember = "payment";

}

}

private void button6\_Click(object sender, EventArgs e)

{

//Вернуться назад в меню

button2.Visible = true;

button3.Visible = true;

button4.Visible = true;

button6.Visible = false;

label1.Visible = true;

dataGridView1.Visible = false;

}

private void button7\_Click(object sender, EventArgs e)

{

//Выход

using (SqlConnection connection = new SqlConnection(connectionString))

{

connection.Close();

this.Close();

}

}

}

}

# 9. Вывод

В процессе данной заботы я научился проектировать (создавать концептуальные, логические и физические модели) и создавать базы данных в среде MS SQL Server 2019. Познакомился со средой для управления инфраструктурой SQL Microsoft SQL Server Management Studio 18. Познакомился с языком T-SQL и с его помощью научился выполнять запросы к базам данных, а также создавать процедуры и триггеры.

В среде Microsoft Visual Studio 2019 создал клиентское приложение для выполнения запросов к созданной базе данных.