

МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Казанский национальный исследовательский технологический университет»

(ФГБОУ ВО «КНИТУ»)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Кафедра | | Интеллектуальных систем и управления информационными | | |
| ресурсами | | | | |
| Направление | | | | 02.03.03 «Математическое обеспечение и администрирование |
| информационных систем» | | | | |
| Профиль | | | Информационные системы и базы данных | |
| Группа | 4301-21 | | | |

**КУРСОВОЙ ПРОЕКТ**

по дисциплине: «Базы данных и системы управления базами данных»

на тему: «Проектирование базы данных туристической фирмы»

|  |  |  |  |
| --- | --- | --- | --- |
| Исполнитель |  |  | Шайдуллина К.Ш. |
|  | (дата, подпись) |  | (Ф.И.О.) |
| Руководитель |  |  |  |
|  | (дата, подпись) |  | (Ф.И.О.) |
| Нормоконтролер |  |  |  |
|  | (дата, подпись) |  | (Ф.И.О.) |

Работа защищена с оценкой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата, подпись)

КАЗАНЬ, 2022

## **ЗАДАНИЕ**

На курсовой проект студенту кафедры: Шайдуллиной Карине Шафкатовне, 4301-21.

Тема проекта: Проектирование базы данных туристической фирмы.

Исходные данные к работе: требуется разработать базу данных для хранения и обработки информации согласно предметной области, указанной в теме курсового проекта. Для реализации применяется реляционная СУБД MySQL. Проект должен включать не менее четырех таблиц, характерные запросы к базе данных, функции для автоматизации вычислений, процедуры для автоматизации рутинных действий пользователей базы данных, а также триггеры, позволяющие поддерживать целостность базы данных.

Содержание расчетно-пояснительной записки (включая перечень подлежащих разработке вопросов, включая вопросы стандартизации и контроля качества)

1. Изучение предметной области, постановка задачи.
2. Концептуальное, логическое и физическое моделирование базы данных.
3. Описание объектов базы данных: запросов, пользовательских функций, хранимых процедур и триггеров.

Перечень графического материала (схемной документации)

1. Модель «сущность-связь» базы данных в нотации П.Чена (1 рис.).

2. EER-модель базы данных в нотации IDEF1X (для объектов) и Crow’s Foot (для связей) (1 рис.).

Консультанты по проекту (с указанием относящихся к ним разделов)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата выдачи задания «19» сентября 2022 г.

Руководитель работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Томилова М.Н.)

(подпись) (фамилия, инициалы)

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 4](#_Toc122599997)

[1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 6](#_Toc122599998)

[2. КОНЦЕПТУАЛЬНОЕ МОДЕЛИРОВАНИЕ БАЗЫ ДАННЫХ 8](#_Toc122599999)

[3. ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ 15](#_Toc122600000)

[4. ФИЗИЧЕСКОЕ ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ 18](#_Toc122600001)

[5. ОПИСАНИЕ ЗАПРОСОВ К БАЗЕ ДАННЫХ 22](#_Toc122600002)

[6. ОПИСАНИЕ ОБЪЕКТОВ ПРОЦЕДУРНОГО SQL 25](#_Toc122600003)

[ЗАКЛЮЧЕНИЕ 34](#_Toc122600004)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 35](#_Toc122600005)

# **ВВЕДЕНИЕ**

На сегодняшний день, базы данных являются частью современных информационных систем, они широко применимы в разнообразных сферах человеческой деятельности [1].

Основные задачи баз данных — это хранение информации, а также лёгкий доступ к ней [1].

Для того чтобы реализовать БД на компьютере имеются особые программы – системы управления базами данных (СУБД). Они предназначены для структурирования информации, размещения ее в таблицах, а также манипулирования данными [1].

В настоящий момент имеется более 50 разных СУБД. В наибольшей степени распространение получили следующие СУБД: MS SQL Server, Oracle Database, Informix, Sybase, IBM DB2, FoxPro, MS Access, бесплатные MySQL, Postgresql, FireBird и др. [1].

По своей структуре системы управления базами данных делятся на сетевые реляционные и иерархические [1].

В реляционной СУБД все данные представляются в виде совокупности взаимосвязанных двумерных таблиц. Все действия с базой данных являются манипуляциями над таблицами. Таблица представляется в виде набора столбцов и строк, а также имеет свое имя. Она отображает тип объекта реального мира (сущность), а каждая строка хранит в себе информацию, которая относится к конкретному объекту [2].

Одной из самых известных реляционных СУБД считается MySQL. Она наиболее универсальна и предпочтительна во многих случаях, т.к. работа с ней по большей части не вызывает трудностей [3].

Цель курсовой работы: разработать базу данных для хранения и обработки информации согласно предметной области.

Чтобы достигнуть поставленную цель следует решить следующие задачи:

* Изучить предметную область
* Произвести концептуальное, логическое и физическое моделирование базы данных.
* Сделать описание объектов базы данных: запросов, пользовательских функций, хранимых процедур и триггеров.

# **1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ**

Туризм является одной из самых больших и динамичных сфер экономики.

Туристическое агентство занимается продажей туров и дополнительных услуг, а также заключением договоров по оказанным услугам. Задача туристической фирмы заключается в том, чтобы правильно забронировать выбранные туристами туры и услуги, а также своевременно оплатить тур оператору, предоставляющему этот тур. Туристический оператор должен предоставить туристам все необходимые услуги [4].

Для повышения производительности труда используют автоматизированные информационные системы, тем самым перекладывая рутинные обязанности персонала на компьютер. База данных позволит хранить данные в более компактном и доступном виде.

Для реализации предоставления туров понадобится информация о возможных турах и дополнительных услугах, которые могут быть оказаны клиентам. Туры составляются туроператорами, после чего передаются туристической фирме. Все услуги будет предоставлять турагент, который будет заключать договор с клиентом, на основы выбранного тура и услуг. Для договора также понадобятся паспортные данные клиента, включая и загранпаспорт [5].

Один турагент может оказывать услуги сразу нескольким клиентам. Однако между клиентом и турагентом может быть заключен только один договор за раз, что не запрещает заключить другой договор с этим же турагентом в следующий раз.

Турагент может производить следующие действия:

* внесение данных о клиентах
* внесение данных о договорах

Турагент может запрашивать данные:

* полную информацию о турах
* информацию о доступных турах
* полную информацию о дополнительных услугах
* полную информацию о клиентах
* данные о договорах
* стоимость дополнительных услуг
* окончательную стоимость (тур и дополнительные услуги)

Туроператор может производить следующие действия:

* добавлять и удалять туры
* назначать цену турам
* добавлять и изменять информацию об отелях
* добавлять дату прибытия и отбытия
* добавлять и удалять дополнительные услуги
* назначать цену дополнительным услугам

Туроператор может запрашивать данные:

* полную информацию о турах
* полную информацию о дополнительных услугах

# **2. КОНЦЕПТУАЛЬНОЕ МОДЕЛИРОВАНИЕ БАЗЫ ДАННЫХ**

Задачей концептуального моделирования является формализованное описание предметной области, которое будет называться концептуальной моделью БД. Концептуальное моделирование состоит из анализа корректности схемы разрабатываемой БД, однако оно не зависит от СУБД, поэтому ее выбор будет сделан позднее [6].

Основные понятия данного метода — это сущность и связи между сущностями. Сущность – это объект или класс однотипных объектов, информация о которых содержится в БД. У каждой сущности есть уникальное имя. Существуют также атрибуты, являющиеся характеристиками, определяющими свойства сущности [6].

Основываясь на информации, приобретенной в ходе анализа предметной области, можно выделить следующие сущности:

1. Клиенты (clients),
2. Турагенты (travel\_agent),
3. Договор (treaty),
4. Туры (tours),
5. Дополнительные услуги (additional\_services),
6. Договор\_Включает\_Дополнительные\_Услуги (treaty\_has\_additional\_services).

Сущность клиенты содержит паспортные данные всех клиентов, включая загранпаспорт, а также номер телефона. Связана с Договором, один клиент может заключать несколько договоров. Детальное представление атрибутов сущности отображено в таблице 1 [7].

Сущность Турагенты содержит информацию о всех турагентах фирмы. Сущность связана с Договором, один турагент способен заключать несколько договоров. Детальное представление атрибутов сущности отображено в таблице 2 [7].

Сущность Договор содержит информацию о всех договорах, включая клиента и турагента, подписывающих этот договор. Он связан с клиентами, турагентами, турами и дополнительными услугами. В одном договоре может быть информация только об одном турагенте и клиенте, а также об одном туре. Детальное представление атрибутов сущности отображено в таблице 3 [7].

Сущность Туры содержит всю информацию о турах: страну, город, отель, тип питания, стоимость на одного человека, а также дату прибытия и отбытия. Туры связаны с договором. Один тур может быть прописан сразу в нескольких договорах. Детальное представление атрибутов сущности отображено в таблице 4 [7].

Сущность Дополнительные услуги содержит информацию о наименовании и стоимости дополнительных услуг. Она связана с Договором и Договор\_Включает\_Дополнительные\_Услуги, в одном договоре может быть указано несколько дополнительных услуг. Детальное представление атрибутов сущности отображено в таблице 5 [7].

Сущность Договор\_Включает\_Дополнительные\_Услуги содержит информацию о количестве человек отправляющихся в тур по конкретному договору. Она связана с Договором и Дополнительными услугами. Детальное представление атрибутов сущности отображено в таблице 6 [7].

На основе результатов анализа предметной области построена модель «сущность-связь» в нотации Чена, которая представлена на рисунке 1.

*Таблица 1. Сущность Клиенты (clients)*

| Название | Ограничение | Описание | Тип данных | Ограничения на тип данных | Домен | Значение по умолчанию |
| --- | --- | --- | --- | --- | --- | --- |
| idClients | Первичный ключ | Идентификатор | Целое число | Not null  Unsigned  Unique | Натуральные числа |  |
| Surname |  | Фамилия | Строка переменной длины | Not null | Только буквы русского алфавита |  |
| Name |  | Имя | Строка переменной длины | Not null | Только буквы русского алфавита |  |
| Patronymic |  | Отчество | Строка переменной длины | Not null | Только буквы русского алфавита |  |
| Phone\_number |  | Номер телефона | Строка переменной длины | Not null | Символ + и 11 натуральных чисел |  |
| Passport\_series |  | Серия паспорта | Целое число | Not null  Unsigned  Unique | Натуральные числа |  |
| Passport\_number |  | Номер паспорта | Целое число | Not null  Unsigned  Unique | Натуральные числа |  |
| Date\_of\_passport\_issue |  | Дата выдачи паспорта | Дата | Not null | Даты формата: ГГГГ-ММ-ДД |  |
| Place\_of\_passport\_issue |  | Кем выдан паспорт | Строка переменной длины | Not null | Только буквы |  |
| Unit\_code |  | Код подразделения | Строка переменной длины | Not null |  |  |
| Gender |  | Пол | Перечисление | Not null | 'м', 'ж' |  |
| Birth\_date |  | Дата рождения | Дата | Not null | Даты формата: ГГГГ-ММ-ДД |  |
| Bith\_place |  | Место рождения | Строка переменной длины | Not null | Только буквы |  |
| Address |  | Адрес прописки | Строка переменной длины | Not null | Буквы, цифры и / |  |
| International\_pas\_number |  | Номер загранпаспорта | Строка переменной длины | Not null  Unique | Только натуральные числа |  |
| Code\_of\_issuing\_state |  | Код государства | Строка переменной длины | Not null | Только латинские буквы |  |
| International\_surname |  | Фамилия загранпаспорт | Строка переменной длины | Not null | Только латинские буквы |  |
| Nationality |  | Гражданство | Строка переменной длины | Not null | Только латинские буквы |  |
| Inter\_pas\_birth\_place |  | Место рождения загранпаспорт | Строка переменной длины | Not null | Буквы кириллицы и латинского алфавита и / |  |
| Inter\_pas\_gender |  | Пол загранпаспорт | Перечисление | Not null | 'm', 'f' |  |
| Date\_of\_inter\_pas\_issue |  | Дата выдачи загранпаспорта | Дата | Not null | Даты формата: ГГГГ-ММ-ДД |  |
| Date\_of\_expiry |  | Дата окончания срока загранпаспорта | Дата | Not null | Даты формата: ГГГГ-ММ-ДД |  |

*Таблица 2. Сущность Турагенты (travel\_agent)*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Название | Ограничение | Описание | Тип данных | Ограничения на тип данных | Домен | Значение по умолчанию |
| idTravel\_Agent | Первичный ключ | Идентификатор турагента | Целое число | Not null  Unique  Unsigned | Натуральные числа |  |
| Surname |  | Фамилия | Строка переменной длины | Not null | Только буквы русского алфавита |  |
| Name |  | Имя | Строка переменной длины | Not null | Только буквы русского алфавита |  |
| Patronymic |  | Отчество | Строка переменной длины | Not null | Только буквы русского алфавита |  |
| Phone\_number |  | Номер телефона | Строка переменной длины | Not null | Символ + и 11 натуральных чисел |  |

*Таблица 3. Сущность Договор (treaty)*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Название | Ограничение | Описание | Тип данных | Ограничения на тип данных | Домен | Значение по умолчанию |
| idTreaty | Первичный ключ | Идентификатор договора | Целое число | Not null  Unique  Unsigned | Натуральные числа |  |
| Date\_of\_signing |  | Дата заключения договора | Дата | Not null  Unsigned | Дата |  |
| Final\_ad\_sv\_price |  | Итоговая сумма | Целое число | Unsigned | Натуральные числа |  |
| Total cost |  | Итоговая цена дополнительных услуг | Целое число | Unsigned | Натуральные числа |  |
| Clients\_idClients | Внешний ключ для clients | Идентификатор, ссылка на клиентов | Целое число | Not null  Unsigned | Натуральные числа |  |
| Travel\_Agent\_idTravel\_Agent | Внешний ключ для travel\_agent | Идентификатор, ссылка на турагентов | Целое число | Not null  Unsigned | Натуральные числа |  |
| Tours\_idTours | Внешний ключ для tours | Идентификатор, ссылка на туры | Целое число | Not null  Unsigned | Натуральные числа |  |

*Таблица 4. Сущность Туры (tours)*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Название | Ограничение | Описание | Тип данных | Ограничения на тип данных | Домен | Значение по умолчанию |
| idTours | Первичный ключ | Идентификатор | Целое число | Not null  Unique  Unsigned | Натуральные числа |  |
| Сountry |  | Страна | Строка переменной длины | Not null | Только буквы русского алфавита |  |
| City |  | Город | Строка переменной длины | Not null | Только буквы русского алфавита |  |
| Hotel |  | Отель | Строка переменной длины | Not null | Только латинские буквы и знак & |  |
| Hotel\_category |  | Категория отеля | Целое число | Not null  Unsigned | Натуральные числа |  |
| Meals |  | Питание | Строка переменной длины | Not null | Только буквы русского алфавита |  |
| Departure\_date |  | Дата прибытия | Дата | Not null | Даты формата: ГГГГ-ММ-ДД |  |
| Arrival\_date |  | Дата отправления | Дата | Not null | Даты формата: ГГГГ-ММ-ДД |  |
| Tour\_price |  | Цена тура | Целое число | Not null  Unsigned | Натуральные числа |  |

*Таблица 5. Сущность Дополнительные услуги (additional\_services)*

| Название | Ограничение | Описание | Тип данных | Ограничения на тип данных | Домен | Значение по умолчанию |
| --- | --- | --- | --- | --- | --- | --- |
| idAdditional\_Services | Первичный ключ | Идентификатор | Целое число | Not null  Unique  Unsigned | Натуральные числа |  |
| Title |  | Название дополнительных услуг | Строка переменной длины | Not null | Только буквы русского алфавита |  |
| Price |  | Цена | Целое число | Not null  Unsigned | Натуральные числа |  |

*Таблица 6. Сущность Договор\_Включает\_Дополнительные\_Услуги (treaty\_has\_additional\_services)*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Название | Ограничение | Описание | Тип данных | Ограничения на тип данных | Домен | Значение по умолчанию |
| Treaty\_idTreaty | Внешний ключ для treaty | Идентификатор, ссылка на договор | Целое число | Not null  Unsigned | Натуральные числа |  |
| Additional\_Services\_idAdditional\_Services | Внешний ключ для additional\_services | Идентификатор, ссылка на дополнительные услуги | Целое число | Not null  Unsigned | Натуральные числа |  |
| Number\_of\_people |  | Количество человек | Целое число | Not null  Unsigned | Натуральные числа |  |

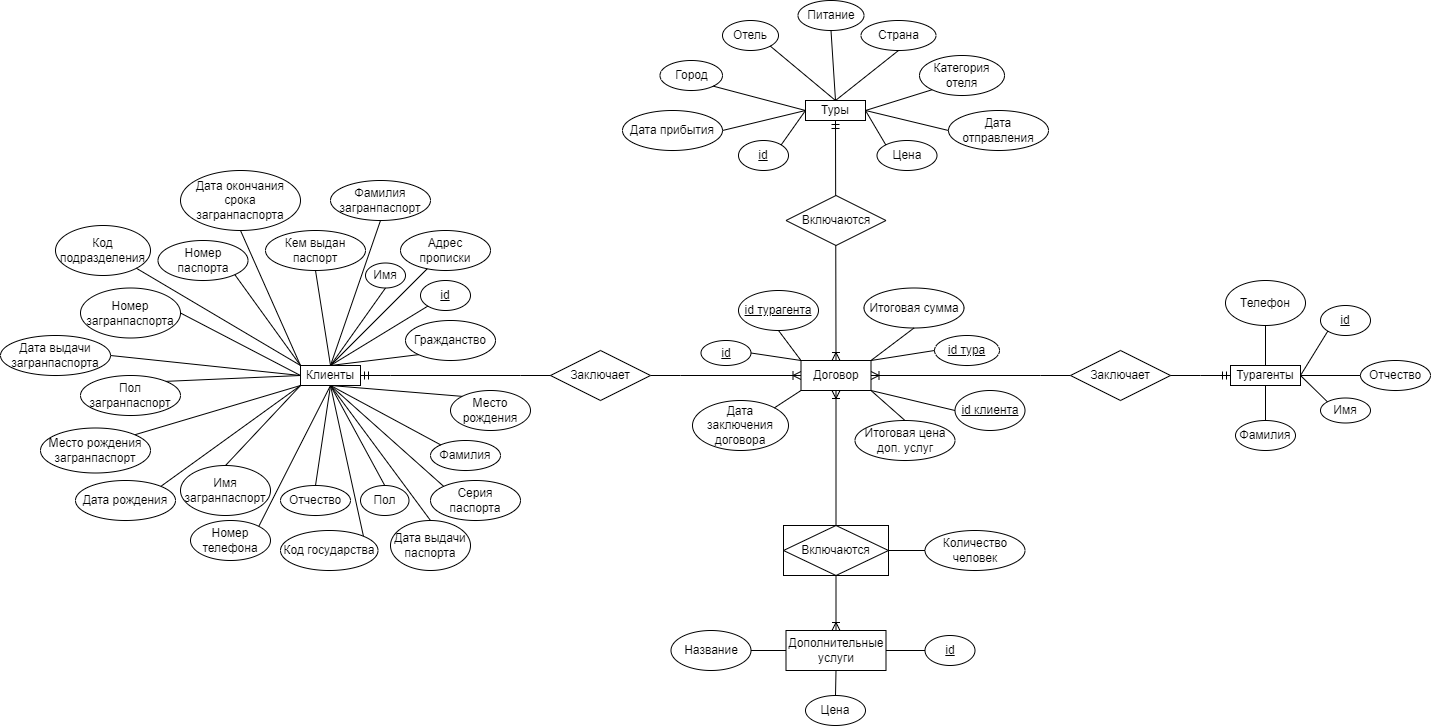


Рисунок 1 - Модель БД «сущность-связь» в нотации Чена

# **3. ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ**

При логическом проектировании необходимо определить все информационные единицы и связи между ними, также типы данных. На этом этапе концептуальную модель преобразовывают в модель, которую поддерживает выбранная СУБД [6].

Формируя логическую схему, нужно выделить сущности путем логической группировки, для этого следует найти функциональны зависимости между атрибутами сущности путем приведения их к 3 нормальной форме [6].

Третьей нормальной формой называется та форма, в которой значения любого поля, которые не входят в первичный ключ не зависят от другого поля, также не входящего в первичный ключ. Т.е. любой атрибут зависит только от первичного ключа данной сущности (ПК-2) [8].

Рисунок 2 иллюстрирует функциональные зависимости сущности Турагенты (travel\_agent). Так как неключевые атрибуты взаимонезависимы, и не входят в первичный ключ, а значения полей неделимы, то таблица Турагенты находится в 3НФ [9].

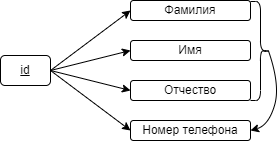


Рисунок 2 - Функциональные зависимости сущности Турагенты (travel\_agent)

Рисунок 3 иллюстрирует функциональные зависимости сущности Дополнительные услуги (additional\_services). Так как неключевые атрибуты взаимонезависимы, и не входят в первичный ключ, а значения полей неделимы, то таблица Дополнительные услуги находится в 3НФ [9].



Рисунок 3 - Функциональные зависимости сущности Дополнительные услуги (additional\_services)

Рисунок 4 иллюстрирует функциональные зависимости сущности Договор (treaty). Так как неключевые атрибуты взаимонезависимы, и не входят в первичный ключ, а значения полей неделимы, то таблица Договор находится в 3НФ [9].

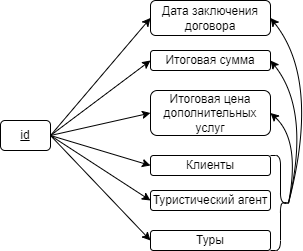


Рисунок 4 - функциональные зависимости сущности Договор (treaty)

Рисунок 5 иллюстрирует функциональные зависимости сущности Туры (tours). Так как неключевые атрибуты взаимонезависимы, и не входят в первичный ключ, а значения полей неделимы, то таблица Туры находится в 3НФ [9].

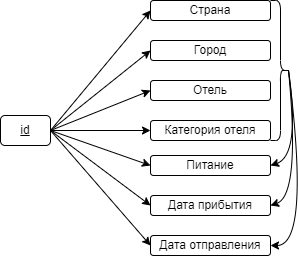


Рисунок 5 - Функциональные зависимости сущности Туры (tours)

Рисунок 6 иллюстрирует функциональные зависимости сущности Клиенты (clients). Так как неключевые атрибуты взаимонезависимы, и не входят в первичный ключ, а значения полей неделимы, то таблица Клиенты находится в 3НФ [9].

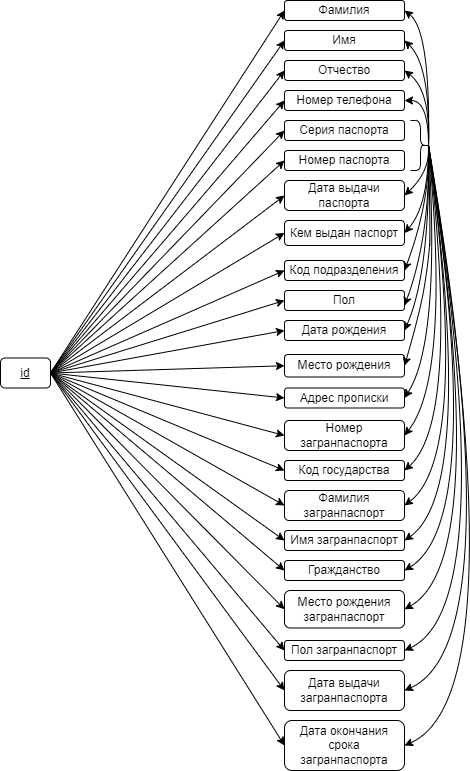


Рисунок 6 - Функциональные зависимости сущности Клиенты (clients)

Рисунок 7 иллюстрирует функциональные зависимости сущности Договор\_Включает\_Дополнительные\_Услуги (treaty\_has\_additional\_services). Так как неключевые атрибуты взаимонезависимы, и не входят в первичный ключ, а значения полей неделимы, то таблица Дополнительные услуги находится в 3НФ [9].

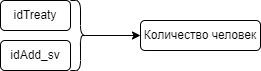


Рисунок 7 - Функциональные зависимости сущности Договор\_Включает\_Дополнительные\_Услуги (treaty\_has\_additional\_services)

# **4. ФИЗИЧЕСКОЕ ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ**

Все СУБД различным образом организуют методы хранения и доступа к данным, поэтому необходимо выбрать наиболее подходящую.

Сегодня наиболее популярной является бесплатная СУБД с открытым исходным кодом под названием MySQL. Широкое распространение она получила благодаря простоте и полной поддержке языка SQL. Также имеется возможность работы не только в текстовом режиме, но и в графическом [3].

Достоинства MySQL:

1. Гибкость, несложность в использовании. При создании и поддержке БД тратиться меньше времени. Необходим небольшой уровень компетенций для полноценной работы с MySQL и реализации ее потенциала.
2. Открытый исходный код, вследствие чего легкость в модификации.
3. Высокий уровень защиты. Система построена таким образом, что большое количество встроенных функций безопасности работают по умолчанию.
4. Скорость. Высокую производительность обеспечивает упрощение некоторых стандартов, которые используются в СУБД.
5. Высокая производительность системы обеспечивается за счет упрощения некоторых используемых в ней стандартов [3].

В таблицах 7 – 12 представлены примеры данных для таблиц, описанных ранее (таблицы 1 - 6).

*Таблица 7. Пример для сущности Клиенты (clients)*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| idClients | Surname | Name | Patronymic | Phone\_number | Passport\_series | Passport\_number | Date\_of\_passport\_issue | Place\_of\_passport\_issue | Unit\_code | Gender | Birth\_date | Bith\_place | Address | International\_pas\_number | Code\_of\_issuing\_state | International\_surname | International\_name | Nationality | Inter\_pas\_birth\_place | Inter\_pas\_gender | Date\_of\_inter\_pas\_issue | Date\_of\_expiry |
| 1 | Прокофьева | Арина | Ивановна | +79364205214 | 3451 | 561980 | 2014-03-21 | МВД по Республике Татарстан | 123-567 | ж | 2000-05-13 | г.Бугульма Республика Татарстан | Энергетиков 3 | 32 1234951 | RUS | Prokofeva | Arina | Russian Federation | Республика Татарстан / Russia | f | 2002-07-12 | 2012-07-12 |
| 2 | Петров | Максим | Львович | +79742014685 | 1209 | 123097 | 2021-08-01 | УФМС России по Краснодарскому краю | 109-383 | м | 1998-11-02 | г.Сочи Краснодарский край | Пригородная 6/10 | 12 2340562 | RUS | Petrov | Maksim | Russian Federation | Краснодарский край / Russia | m | 2012-02-05 | 2022-02-05 |
| 3 | Громов | Степан | Эмильевич | +78451249632 | 1478 | 354729 | 2016-09-07 | МВД по Республике Татарстан | 112-574 | м | 2005-04-16 | г.Казань Республика Татарстан | Проспект Ямашева 61Б | 32 3776540 | RUS | Gromov | Stepan | Russian Federation | Республика Татарстан / Russia | m | 2003-12-01 | 2013-12-01 |
| 4 | Чернова | Кира | Антоновна | +79854123675 | 4598 | 178944 | 2000-03-14 | МВД по Республике Чувашия | 543-088 | ж | 1980-07-15 | г.Чебоксары Республика Чувашия | Проспект Мира 62Б | 45 8745987 | RUS | Chernova | Kira | Russian Federation | Республика Чувашия / Russia | f | 2015-10-12 | 2025-10-12 |

*Таблица 8. Пример для сущности Турагенты (travel\_agent)*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| idTravel\_Agent | Surname | Name | Patronymic | Phone\_number |
| 1 | Белякова | Ульяна | Дмитреевна | +79842314574 |
| 2 | Казаков | Владимир | Егорович | +79841236584 |
| 3 | Филатов | Виталий | Васильевич | +74562384056 |

*Таблица 9. Пример для сущности Договор (treaty)*

| idTreaty | Date\_of\_signing | Final\_ad\_sv\_price | Total\_cost | Clients\_idClients | Travel\_Agent\_idTravel\_Agent | Tours\_idTours |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 2022-01-15 | 82000 | 584227 | 1 | 3 | 2 |
| 2 | 2022-01-02 | 4000 | 502128 | 5 | 2 | 5 |
| 3 | 2019-04-17 | 127000 | 640324 | 3 | 4 | 12 |

*Таблица 10. Пример для сущности Туры (tours)*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| idTours | Сountry | City | Hotel | Hotel\_category | Meals | Departure\_date | Arrival\_date | Tour\_price |
| 1 | Индия | Нью Дели | Fortuna Delhi | 3 | Завтрак и ужин | 2022-10-12 | 2022-10-19 | 35779 |
| 2 | Вьетнам | Нанчанг | Vinpearl Nha Trang Resort | 5 | Завтрак, обед и ужин | 2022-02-03 | 2022-02-10 | 125032 |
| 3 | Шри-Ланка | Бентота | Heritance Ahungalla | 5 | Все включено | 2021-05-14 | 2021-05-24 | 112578 |
| 4 | Мексика | Канкун | Grand Oasis Palm | 5 | Все включено | 2021-01-15 | 2021-01-22 | 168311 |

*Таблица 11. Пример для сущности Дополнительные услуги (additional\_services)*

|  |  |  |
| --- | --- | --- |
| idAdditional\_Services | Title | Price |
| 1 | Оформление визы | 8000 |
| 2 | Личный переводчик или гид | 80000 |
| 3 | Трансфер в аэропорт | 2000 |

*Таблица 12. Пример для сущности Договор\_Включает\_Дополнительные\_Услуги (treaty\_has\_additional\_services)*

|  |  |  |
| --- | --- | --- |
| Treaty\_idTreaty | Additional\_Services\_idAdditional\_Services | Number\_of\_people |
| 1 | 2 | 3 |
| 1 | 3 | 3 |
| 2 | 6 | 4 |

База данных была спроектирована в MySQL Workbench при помощи прямого инжиниринга (forward engineering). MySQL Workbench является кроссплатформенным инструментом, с помощью которого можно проектировать реляционные базы данных. Он позволяет добавлять функциональность и упрощать разработку MySQL и SQL. В MySQL Workbench объединяются проектирование, разработка, создание, администрирования и обслуживание SQL, а также возможность создания графического интерфейса для работы с базой данных (ПК-2) [10].

Для представления схемы данных, сущностей и их связей предназначен редактор EER-диаграмм. Для визуального представления выбраны нотации для объектов – IDEF1X и для связей – Crow’s Foot [10].

На рисунке 8 представлена модель базы данных в виде EER-диаграммы.

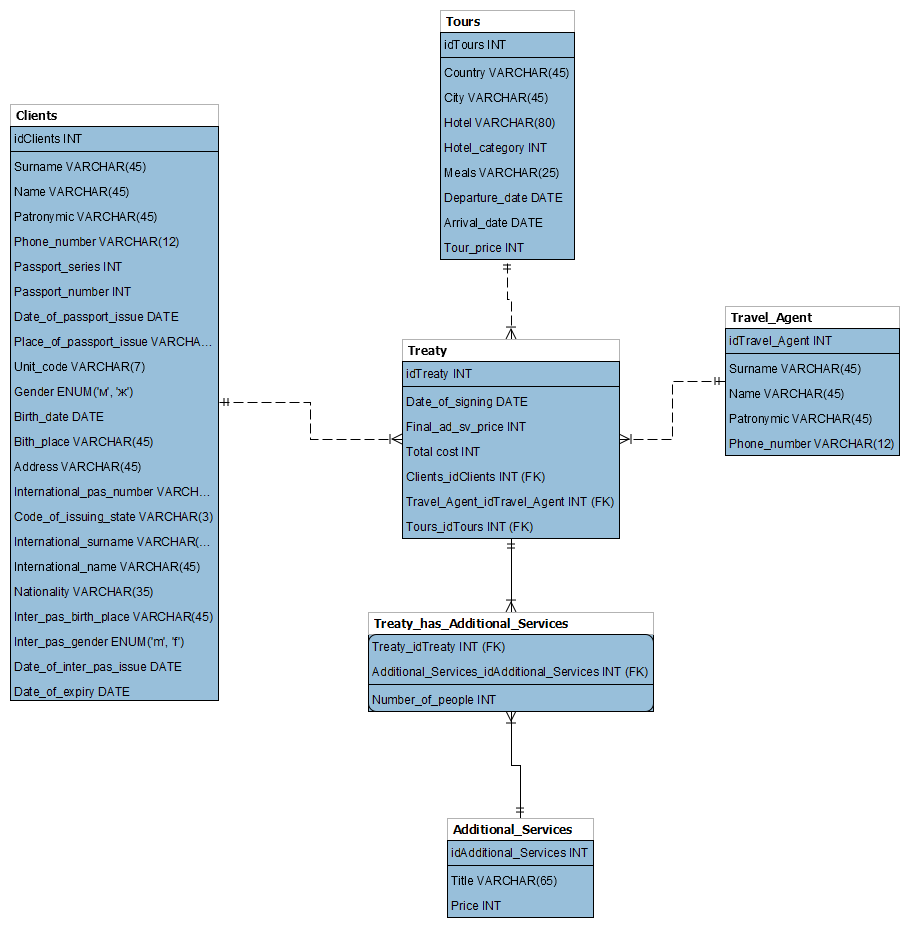


Рисунок 8 - Модель базы данных в виде EER-диаграммы.

# **5. ОПИСАНИЕ ЗАПРОСОВ К БАЗЕ ДАННЫХ**

1. Запрос на вывод краткой информации о клиентах и их турах.

Запрос выводит ФИО, возраст, телефон клиентов, а также страну, город и продолжительность забронированного тура для тех клиентов, кто родился после 2015. Запрос сортируется по возрасту.

*Код запроса:*

select concat(Surname,' ',Name,' ',Patronymic) as 'ФИО',

date\_format(from\_days(to\_days(now()) - to\_days(Birth\_date)), '%Y') + 0 as 'Возраст',

Phone\_number as 'Телефон',

Сountry as 'Страна',

City as 'Город',

(Arrival\_date-Departure\_date) as 'Количество\_дней'

from clients join treaty on idClients=Clients\_idClients join tours on idTours=Tours\_idTours

where Birth\_date<'2005-01-01' and Surname like '%о%'

order by Возраст;

1. Запрос на вывод информации об отеле с ценой тура больше средней.

Запрос выводит имя и дату рождения клиентов, страну, отель вместе с категорией, а также цену тура. Только для тех туров, где цена тура выше средней цены всех туров. Запрос сортируется по дате подписания договора.

*Код запроса:*

select Name as 'Имя',

Birth\_date as 'Дата\_рождения',

Сountry as 'Страна',

concat(Hotel,' ',Hotel\_category) as 'Отель',

Tour\_price as 'Цена\_тура'

from clients join treaty on idClients=Clients\_idClients join tours on idTours=Tours\_idTours

where Tour\_price>(select avg(Tour\_price) from tours)

order by Date\_of\_signing;

1. Запрос на подсчет количества клиентов у каждого турагента.

Запрос выводит ФИО турагента, телефон и количество клиентов у каждого турагента. Но только тех, у кого категория отеля больше 3 и дата подписания договора между 2021 и текущим годом. Запрос сортируется и группируется по id турагента.

*Код запроса:*

select concat(travel\_agent.Surname,' ',travel\_agent.Name) as 'Туристический\_агент',

travel\_agent.Phone\_number as 'Телефон',

count(idClients) as 'Количество\_клиентов'

from clients join treaty on idClients=Clients\_idClients join tours on idTours=Tours\_idTours join travel\_agent on Travel\_Agent\_idTravel\_Agent=idTravel\_Agent

where Hotel\_category>3 and Date\_of\_signing between '2021-01-01' and now()

group by idTravel\_Agent

order by idTravel\_Agent;

1. Запрос на вывод паспортных данных клиента.

Запрос выводит ФИО клиента, серию, номер, дату выдачи, место выдачи паспорта, код подразделения, пол, дату рождения, место рождения, адрес прописки по паспорту. Где год подписания договора равен текущему. Запрос сортируется по ФИО.

*Код запроса:*

select concat(Surname,' ',Name,' ',Patronymic) as 'ФИО',

Passport\_series as 'Серия',

Passport\_number as 'Номер',

Date\_of\_passport\_issue as 'Дата\_выдачи',

Place\_of\_passport\_issue as 'Место\_выдачи',

Unit\_code as 'Код\_подразделения',

Gender as 'Пол',

Birth\_date as 'Дата\_рождения',

Bith\_place as 'Место\_рождения',

Address as 'Адрес\_прописки'

from clients join treaty on idClients=Clients\_idClients

where year(Date\_of\_signing)=year(now())

order by ФИО;

1. Запрос на вывод информации об отелях.

Запрос выводит название и категорию отеля, питание, дату отправления и прибытия, окончательную цену. Только для тех отелей, где окончательная сумма по договору больше средней цены всех клиентов. Запрос сортируется по Отелю.

*Код запроса:*

select Hotel as 'Отель',

Hotel\_category as 'Категория\_отеля',

Meals as 'Питание',

Departure\_date as 'Дата\_отправления',

Arrival\_date as 'Дата\_прибытия',

Total\_cost as 'Окончательная\_цена'

from tours join treaty on Tours\_idTours=idTours

where Total\_cost > (select avg(Total\_cost) from treaty)

order by Отель;

1. Запрос на вывод информации по загранпаспорту.

Запрос выводит ФИО клиента, серию загранпаспорта, код государства, национальность, место рождения, пол, дату выдачи и дату истечения срока загранпаспорта. Запрос сортируется по полу, установлен лимит на 5 строк.

*Код запроса:*

select concat (International\_surname,' ',International\_name) as 'ФИО',

International\_pas\_number as 'Серия\_загранпаспорта',

Code\_of\_issuing\_state as 'Код\_государства',

Nationality as 'Национальность',

Inter\_pas\_birth\_place as 'Место\_рождения',

Inter\_pas\_gender as 'Пол',

Date\_of\_inter\_pas\_issue as 'Дата\_выдачи',

Date\_of\_expiry as 'Дата\_истечения\_срока'

from clients

order by Inter\_pas\_gender desc

limit 0,5;

Представление – краткая информация о клиентах, турагентах и договоре.

Представление выводит ID, ФИО, номер телефона, адрес клиента, ID, ФИО, номер телефона турагента и ID договора и окончательную стоимость договора (ПК-2).

*Код представления:*

create or replace view Информация (ID\_клиента, ФИО\_клиента, Телефон\_клиента, Адрес\_клиента, ID\_Турагента, ФИО\_Турагента, Телефон\_турагента, ID\_Договора, Окончательная\_цена) as

select idClients,

concat(clients.Surname,' ',clients.Name,' ',clients.Patronymic),

clients.Phone\_number,

Address,

idTravel\_Agent,

concat(travel\_agent.Surname,' ',travel\_agent.Name,' ',travel\_agent.Patronymic),

travel\_agent.Phone\_number,

idTreaty,

Total\_cost

from clients join treaty on idClients=Clients\_idClients join travel\_agent on Travel\_Agent\_idTravel\_Agent=idTravel\_Agent

order by idClients;

# **6. ОПИСАНИЕ ОБЪЕКТОВ ПРОЦЕДУРНОГО SQL**

**6.1. Функции**

1. Функция для подсчета количества дней тура.

*Синтаксис функции*

count\_days(id int)

*Аргументы функции*

id – id клиента, тип данных int

*Результат*

Количество дней тура тип данных int

Пример реализации изображен на рисунке 8.



Рисунок 9 – пример реализации функции count\_days()

*Описание алгоритма*

1. Вводим id клиента.
2. На основе даты прибытия и отбытия находим и возвращаем количество дней.

*Код функции:*

delimiter $$

create function `count\_days`(id int) returns int

deterministic

begin

declare ar\_d, dp\_d date;

declare count\_d int;

set ar\_d = (select Arrival\_date from tours join treaty on idTours=Tours\_idTours where id=Clients\_idClients);

set dp\_d = (select Departure\_date from tours join treaty on idTours=Tours\_idTours where id=Clients\_idClients);

set count\_d = ar\_d - dp\_d;

return count\_d;

end $$

delimiter ;

1. Функция для подсчета итоговой цены дополнительных услуг.

*Синтаксис функции*

final\_ad\_sv\_price(id int)

*Аргументы функции*

id – id договора, тип данных int

*Результат*

Сумма итоговой стоимости дополнительных услуг в договоре тип данных int.

Пример реализации изображен на рисунке 9.



Рисунок 10 – Пример реализации функции final\_ad\_sv\_price()

*Описание алгоритма*

1. Вводим id договора
2. На основе цен дополнительных услуг выбранных по id находим итоговую стоимость этих услуг.

*Код функции:*

delimiter $$

create function `final\_ad\_sv\_price`(id int) returns int

deterministic

begin

declare final\_price varchar(10);

declare f int;

set final\_price = (select sum(Price) from additional\_services join treaty\_has\_additional\_services on Additional\_Services\_idAdditional\_Services=idAdditional\_Services

join treaty on Treaty\_idTreaty=idTreaty where idTreaty=id);

set f = convert(final\_price,UNSIGNED);

return f;

end $$

delimiter ;

1. Функция для подсчета итоговой суммы по договору.

*Синтаксис функции*

total\_cost(id int)

*Аргументы функции*

id – id договора, тип данных int

*Результат*

Итоговая сумма тура и дополнительных услуг в договоре тип данных int.

Пример реализации изображен на рисунке 10.



Рисунок 11 – Пример реализации функции total\_cost()

*Описание алгоритма*

1. Вводим id договора
2. На основе цены тура и итоговой цены дополнительных услуг выбранных по id находим итоговую сумму.

*Код функции:*

delimiter $$

create function `total\_cost`(id int) returns int

deterministic

begin

declare tour,t\_price, total\_cost,num\_of\_pep, ad\_sv int;

set tour = (select Tours\_idTours from treaty where id=idTreaty);

set t\_price = (select Tour\_price from tours where idTours=tour);

set num\_of\_pep =(select Number\_of\_people from treaty\_has\_additional\_services where id=Treaty\_idTreaty limit 1);

set total\_cost = num\_of\_pep \* t\_price + final\_ad\_sv\_price(id);

return total\_cost;

end $$

delimiter ;

**6.2. Процедуры**

1. Процедура для обновления итоговой цены дополнительных услуг в договоре (ПК-2).

*Синтаксис процедуры*

Ad\_sv\_price(id int)

*Входные данные*

id – id договора, тип данных int

*Результат*

Изменение столбца Final\_ad\_sv\_price в таблице treaty.

Пример реализации изображен на рисунке 11.

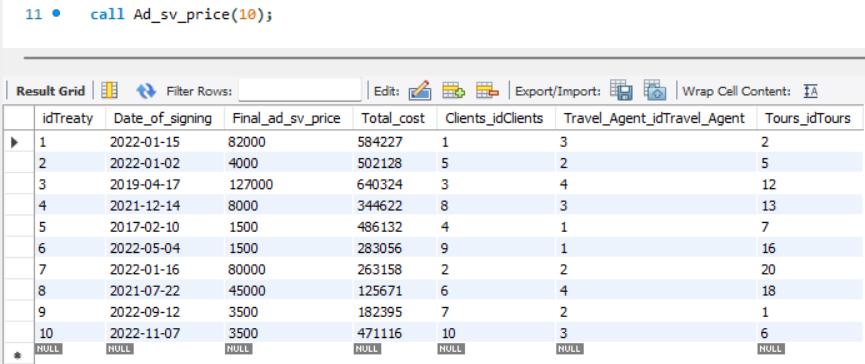


Рисунок 12 – пример реализации процедуры Ad\_sv\_price()

*Описание алгоритма*

1. Вводим id договора
2. Изменяем столбец Final\_ad\_sv\_price в таблице treaty.

*Код процедуры:*

delimiter $$

create procedure `Ad\_sv\_price`(id int)

deterministic

begin

update treaty

set Final\_ad\_sv\_price=final\_ad\_sv\_price(id) where id=idTreaty;

end $$

delimiter ;

1. Процедура для обновления итоговой суммы в договоре (ПК-2).

*Синтаксис процедуры*

Total\_cost(id int)

*Входные данные*

id – id договора, тип данных int

*Результат*

Изменение столбца Total\_cost в таблице treaty.

Пример реализации изображен на рисунке 12.

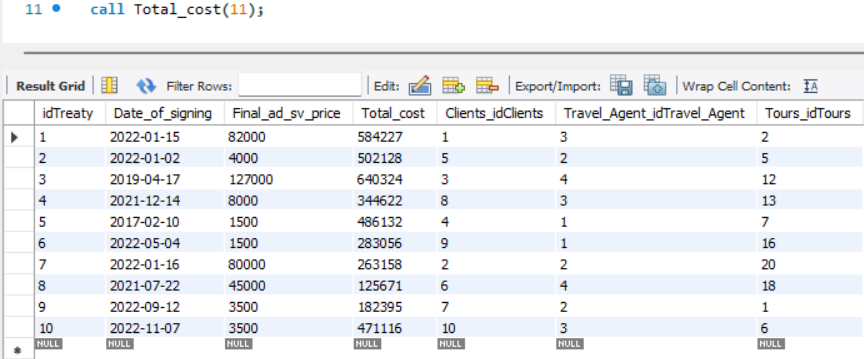


Рисунок 13 – пример реализации процедуры Total\_cost()

*Описание алгоритма*

1. Вводим id договора
2. Изменяем столбец Total\_cost в таблице treaty.

*Код процедуры:*

delimiter $$

create procedure `Total\_cost`(id int)

deterministic

begin

update treaty

set Total\_cost=total\_cost(id) where id=idTreaty;

end $$

delimiter ;

1. Процедура для вывода клиентов с определенным типом питания в отеле.

*Синтаксис процедуры*

Client\_for\_meal(meal varchar(25))

*Входные данные*

meal – тип питания, тип данных varchar(25)

*Результат*

Вывод ФИО клиентов, названия и категории отеля, страны, города и количества дней.

Пример реализации изображен на рисунке 13.

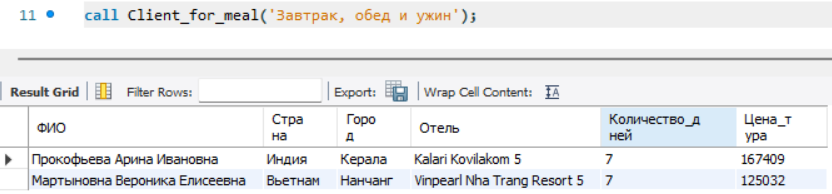


Рисунок 14 – пример реализации процедуры Client\_for\_meal()

*Описание алгоритма*

1. Вводим тип питания
2. Вывод необходимой информации на основе типа питания.
3. Код процедуры:

delimiter $$

create procedure `Client\_for\_meal`(meal varchar(25))

deterministic

begin

select concat(Surname,' ',Name,' ',Patronymic) as 'ФИО',

Сountry as 'Страна',

City as 'Город',

concat(Hotel,' ',Hotel\_category) as 'Отель',

count\_days(idClients) as 'Количество\_дней',

Tour\_price as 'Цена\_тура'

from clients join treaty on idClients=Clients\_idClients join tours on Tours\_idTours=idTours

where Meals=meal;

end $$

delimiter ;

1. Процедура для нахождения клиентов выбранного турагента.

*Синтаксис процедуры*

Clients\_from\_agent(id int)

*Входные данные*

id – id турагента, тип данных int

*Результат*

Вывод ФИО турагента, id клиента и ФИО клиента.

Пример реализации изображен на рисунке 14.

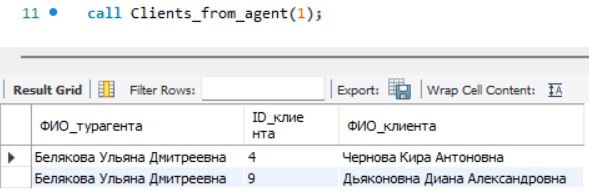


Рисунок 15 – пример реализации процедуры Clients\_from\_agent()

*Описание алгоритма*

1. Вводим id турагента
2. Вывод клиентов выбранного турагента.

Код процедуры:

delimiter $$

create procedure `Clients\_from\_agent`(id int)

deterministic

begin

select

concat(travel\_agent.Surname,' ',travel\_agent.Name,' ',travel\_agent.Patronymic) as 'ФИО\_турагента',

Clients\_idClients as 'ID\_клиента',

concat(clients.Surname,' ',clients.Name,' ',clients.Patronymic) as 'ФИО\_клиента'

from clients join treaty on Clients\_idClients=idClients join travel\_agent on idTravel\_Agent=Travel\_Agent\_idTravel\_Agent

where id=Travel\_Agent\_idTravel\_Agent;

end $$

delimiter ;

1. Процедура для вывода клиента и даты подписания договора для турагента на следующий месяц.

*Синтаксис процедуры*

next\_month\_signing()

*Входные данные*

id – id турагента, тип данных int

*Результат*

Вывод ФИО клиента, даты заключения договора на следующий месяц.

Пример реализации изображен на рисунке 15.

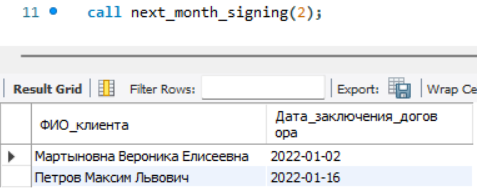


Рисунок 16 – пример реализации процедуры next\_month\_signing()

*Описание алгоритма*

1. Вводим id турагента
2. Вывод клиента и даты заключения договора у заданного турагента на следующий месяц.

*Код процедуры:*

delimiter $$

create procedure `next\_month\_signing`(id int)

deterministic

begin

select concat(clients.Surname,' ',clients.Name,' ',clients.Patronymic) as 'ФИО\_клиента',

Date\_of\_signing as 'Дата\_заключения\_договора'

from clients join treaty on Clients\_idClients=idClients join travel\_agent on idTravel\_Agent=Travel\_Agent\_idTravel\_Agent

where id=idTravel\_Agent and month(Date\_of\_signing)=month(date\_add(now(), interval 1 month));

end $$

delimiter ;

1. Процедура для поиска информации о туре по стране на текущий год.

*Синтаксис процедуры*

tours\_for\_country(country varchar(17))

*Входные данные*

country – страна тура, тип данных varchar(17)

*Результат*

Вывод id тура, города, отеля вместе с категорией, типа питания, даты прибытия и отправления, цены тура.

Пример реализации изображен на рисунке 16.

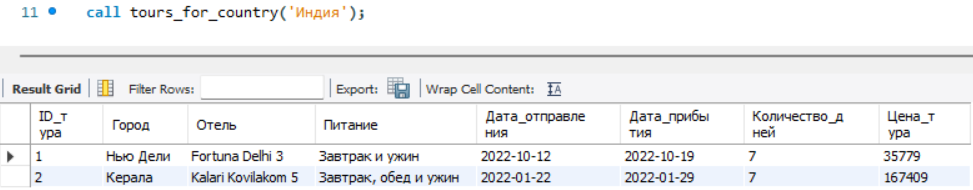


Рисунок 17 – пример реализации процедуры tours\_for\_country()

*Описание алгоритма*

1. Вводим название страны
2. Вывод данных о туре по выбранной стране.

*Код процедуры:*

delimiter $$

create procedure `tours\_for\_country`(country varchar(17))

deterministic

begin

select idTours as 'ID\_тура',

City as 'Город',

concat(Hotel,' ',Hotel\_category) as 'Отель',

Meals as 'Питание',

Departure\_date as 'Дата\_отправления',

Arrival\_date as 'Дата\_прибытия',

count\_days(Clients\_idClients) as 'Количество\_дней',

Tour\_price as 'Цена\_тура'

from tours join treaty on idTours=Tours\_idTours where Сountry=country and year(Departure\_date)=year(now());

end $$

delimiter ;

**6.3. Триггеры**

1. Триггер для обновления итоговой цены дополнительных услуг при изменении дополнительных услуг (ПК-2).

*Название*: ad\_sv\_BEFORE\_UPDATE

*Алгоритм*: при изменении дополнительных услуг в таблице treaty\_has\_additional\_services в таблице treaty обновляется итоговая стоимость дополнительныз услуг.

*Код триггера:*

delimiter $$

create trigger `ad\_sv\_BEFORE\_UPDATE`

before update

on `treaty\_has\_additional\_services` for each row

begin

if (old.Additional\_Services\_idAdditional\_Services<>new.Additional\_Services\_idAdditional\_Services) then

update treaty

set `Final\_ad\_sv\_price` = Ad\_sv\_price(new.Treaty\_idTreaty) WHERE idTreaty = NEW.Treaty\_idTreaty;

end if;

end $$

delimiter ;

1. Триггер для обновления итоговой цены дополнительных услуг при удалении дополнительных услуг (ПК-2).

*Название*: ad\_sv\_BEFORE\_DELETE

*Алгоритм*: при удалении дополнительных услуг в таблице treaty\_has\_additional\_services в таблице treaty обновляется итоговая стоимость дополнительныз услуг.

*Код триггера:*

delimiter $$

create trigger `ad\_sv\_BEFORE\_DELETE`

before delete

on `treaty\_has\_additional\_services` for each row

begin

update treaty

set `Final\_ad\_sv\_price` = Ad\_sv\_price(old.Treaty\_idTreaty) WHERE idTreaty = old.Treaty\_idTreaty;

end $$

delimiter ;

1. Триггер для обновления итоговой цены дополнительных услуг после внесения дополнительных услуг (ПК-2).

*Название*: ad\_sv\_AFTER\_INSERT

*Алгоритм*: после внесения дополнительных услуг в таблицу treaty\_has\_additional\_services в таблице treaty обновляется итоговая стоимость дополнительныз услуг.

*Код триггера:*

delimiter $$

create trigger `ad\_sv\_AFTER\_INSERT`

after insert

on `treaty\_has\_additional\_services` for each row

begin

update treaty

set `Final\_ad\_sv\_price` = Ad\_sv\_price(new.Treaty\_idTreaty) WHERE idTreaty = NEW.Treaty\_idTreaty;

end $$

delimiter ;

# **ЗАКЛЮЧЕНИЕ**

В рамках данного курсового проекта была создана функционирующая база данных туристической фирмы. В данном проекте реализован частичный функционал подобной системы, было создано шесть процедур, три функции, три триггера и одно представление. Реализованы пользовательские функции для:

* для подсчета количества дней тура - count\_days();
* для подсчета итоговой цены дополнительных услуг – final\_ad\_sv\_price();
* для подсчета итоговой суммы по договору – total\_cost();

Созданы процедуры:

* для обновления итоговой цены дополнительных услуг в договоре – Ad\_sv\_price();
* для обновления итоговой суммы в договоре – Total\_cost();
* для вывода клиентов с определенным типом питания в отеле – Client\_for\_meal();
* для нахождения клиентов выбранного турагента – Clients\_from\_agent();
* для вывода клиента и даты подписания договора для турагента на следующий месяц – next\_month\_signing().

Разработанные триггеры позволяют в автоматическом режиме обновлять итоговую цену дополнительных услуг.

Для всех таблиц БД были выявлены и проанализированы функциональные зависимости атрибутов, таблицы БД были приведены к 3НФ, чтобы избежать возможных аномалий и для поддержания целостности базы данных. В ходе выполнения были закреплены навыки работы в MySQL Workbench, а также навыки разработки реляционных баз данных – создания функций, процедур, триггеров, запросов, связей и таблиц. Были выполнены все этапы реализации БД.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Основы баз данных: Учебно-методическое пособие к выполнению самостоятельной работы / Е. А. Сидорова, А.В. Долгова; Омский гос. ун-т путей сообщения. Омск, 2020. 23 с. - URL: https://e.lanbook.com/book/165700 (дата обращения: 04.06.2021). - Режим доступа: для авторизированных пользователей.
2. Муравьев А.И. Базы данных: Учебное пособие. - - Томск: Томский государственный университет систем управления и радиоэлектроники. 136 с. - URL: https://e.lanbook.com/book/11788 (дата обращения: 04.06.2021). - Режим доступа: для авторизированных пользователей.
3. Нурмагомедова Н.Х., Исаева Г.Г. WEB- технологии. Курс лекций. - Махачкала: ДГПУ, АЛЕФ, 2022 - 81с. - URL: https://e.lanbook.com/book/262442 (дата обращения: 04.06.2021). - Режим доступа: для авторизированных пользователей.
4. Tourfaq.net [сайт] - URL: http://tourfaq.net/travel-business/turisticheskaya-industriya-i-ee-sostavlyayushhie/ (дата обращения: 04.06.2021).
5. Ikirov.ru [сайт] - URL: https://www.ikirov.ru/news/4583-chem-otlichaetsya-turfirma-turagentstvo-turoperator (дата обращения: 04.06.2021).
6. Шнырев С.Л. Базы данных: Учебное пособие. М.: НИЯУ МИФИ, 2011. - 224 с. - Текст: электронный // Лань: электронно-библиотечная система. - URL: https://e.lanbook.com/book/75809 (дата обращения: 04.06.2021). - Режим доступа: для авторизированных пользователей.
7. Осипов Д. Л. Технологии проектирования баз данных. - М.: ДМК Пресс, 2019. - 498 с.: ил. - URL: https://e.lanbook.com/book/131692 (дата обращения: 04.06.2021). - Режим доступа: для авторизированных пользователей.
8. Токмаков, Геннадий Петрович Базы данных: Модели и структуры данных, язык SQL, программирование баз данных: учебное пособие / Г. П. Токмаков. - Ульяновск: УлГТУ, 2021. - 362 с. - Текст: электронный // Лань: электронно-библиотечная система. - URL: https://e.lanbook.com/book/259706 (дата обращения: 04.06.2021). - Режим доступа: для авторизированных пользователей.
9. Фешина Е. В. Базы данных : учебник / Е. В. Фешина, В. В. Ткаченко. - Краснодар: КубГАУ, 2020. - 172 с. - URL: https://e.lanbook.com/book/254261 (дата обращения: 04.06.2021). - Режим доступа: для авторизированных пользователей.
10. Мартишин, С. А. Проектирование и реализация баз данных в СУБД MySQL с использованием MySQL Workbench: учеб. пособие / С.А. Мартишин, В.Л. Симонов, М.В. Храпченко. — М.: ИД «ФОРУМ»: ИНФРА-М, 2019. – 160 с. – Текст: электронный. – URL: https://znanium.com/catalog/product/1007949 (дата обращения: 04.06.2021). – Режим доступа: по подписке.