**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**автономное образовательное учреждение высшего образования**

**«Санкт-Петербургский государственный политехнический университет Петра Великого»**

**(ФГАОУ ВО «СПбПУ»)**

**Институт среднего профессионального образования**

**КУРСОВАЯ РАБОТА**

Программирование в компьютерных системах   
специальность

отделение: Заочное

По дисциплине: Основы разработки и защиты баз данных

|  |  |
| --- | --- |
| Тема: | Гостиница |

Исполнитель: студент 3 курса 32928/1 группы

Матвеенко Дмитрий Владимирович

(фамилия, имя, отчество)

|  |  |
| --- | --- |
| Руководитель: | преподаватель ИСПО |
| (ученная степень, звание) |
| Славгородская Анна Александровна |
| (фамилия, имя, отчество) |

Курсовая работа допущена к Защитил(а) курсовую работу с  
защите оценкой

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

Санкт-Петербург

2019

**Оглавление**

[ВВЕДЕНИЕ 4](#_Toc28424243)

[ЧАСТЬ 1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 5](#_Toc28424244)

[1.1. Описание процессов, информацию о которых надо хранить в БД 5](#_Toc28424245)

[1.2. Пользователи информационной системы, для которой проектируется БД 5](#_Toc28424246)

[1.3. Задачи пользователей системы 5](#_Toc28424247)

[1.4. Пользовательские ограничения предметной области 6](#_Toc28424248)

[ЧАСТЬ 2. ER-МОДЕЛЬ ПРЕДМЕТНОЙ ОБЛАСТИ В СТАНДАРТЕ IDEF1X 7](#_Toc28424249)

[2.1. Смысловое описание сущностей с указанием идентификаторов 7](#_Toc28424250)

[2.2. Описание атрибутов каждой сущности 7](#_Toc28424251)

[2.3. Описание доменов, которым принадлежат атрибуты сущностей, с указанием ограничений на возможные значения 8](#_Toc28424252)

[2.4. Описание типов связей между сущностями 9](#_Toc28424253)

[2.5. ER-диаграмма предметной области 10](#_Toc28424254)

[ЧАСТЬ 3. РЕЛЯЦИОННАЯ МОДЕЛЬ БАЗЫ ДАННЫХ 11](#_Toc28424255)

[3.1. Полное название СУБД, средствами которой создается БД 11](#_Toc28424256)

[3.2. Набор команд SQL для создания структуры БД в указанной СУБД 11](#_Toc28424257)

[3.4. Выполнение листинга создания базы данных 14](#_Toc28424258)

[3.5. Схема базы данных с указанием внешних ключей и их свойств 14](#_Toc28424259)

[ЧАСТЬ 4. ВВОД ДАННЫХ И ВЫПОЛНЕНИЕ ТИПОВЫХ ЗАПРОСОВ 16](#_Toc28424260)

[4.1. Набор команд для ввода данных 16](#_Toc28424261)

[4.2. Запросы к базе данных 20](#_Toc28424262)

[Часть 5. Реализация ограничений БД с помощью хранимых процедур и триггеров 22](#_Toc28424263)

[ЗАКЛЮЧЕНИЕ 25](#_Toc28424264)

[СПИСОК ЛИТЕРАТУРЫ 26](#_Toc28424265)

# ВВЕДЕНИЕ

Хранение информационных данных – бесспорно, достаточно важная и актуальная задача в век информационных технологий. Данные могут храниться в различных форматах, в различных структурах, в различных видах и представлениях. Практически любой пользователь может взаимодействовать с данными, модифицировать и обеспечивать их выгрузку в различных форматах. Меж тем, следует понимать, что администрирование и создание базы данных – достаточно трудоемкий процесс. Следует рассмотреть различные аспекты проектирования, а также создать такую модель представления данных, которая не была бы избыточна и вместе с тем была бы функциональна.

Актуальность проекта обеспечивается за счет достаточно высокого спроса на рынке труда на базы данных, а также повсеместным использованием баз данных в программных проектах.

Объектом изучения являются базы данных.

Предметом изучения является создание базы данных гостинцы в MySQL.

Цель проекта: спроектировать и реализовать базу данных по выбранной предметной области.

Задачи проекта:

1. изучение литературы по выбранной тематике;
2. изучение литературы по проектированию баз данных;
3. изучение литературы по реализации БД в СУБД;
4. реализация базы данных и составление запросов к ней;
5. создание документации.

# ЧАСТЬ 1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

## 1.1. Описание процессов, информацию о которых надо хранить в БД

Гостиница сдает номера посуточно постояльцам. Клиент должен заранее забронировать номер на нужные даты, без брони заселение осуществляется только при наличии свободных мест.

При бронировании и заселении клиент должен предоставить данные паспорта, при выселении ему предоставляется счет, и он должен оплатить съем номера и дополнительные услуги (если предоставлялись).

В гостинице есть несколько типов номеров, разной степени комфортности и стоимости за сутки. Также существует прайс-лист для дополнительных услуг.

Информация, которая должна хранится в системе:

* Данные о брони и заселенных номерах;
* Данные о клиентах;
* Данные о номерах и типах номеров;
* Данные об дополнительных услугах и их предоставлении;
* Данные об оплате.

## 1.2. Пользователи информационной системы, для которой проектируется БД

Пользователи системы:

* Администратор гостиницы;
* Оператор БД.

## 1.3. Задачи пользователей системы

Администратор гостиницы:

* Предоставление счета по номеру брони;
* Подсчет суммы оплаты для клиента;
* Предоставление свободных номеров на указанную дату;
* Выдача прайс-листа;
* Сведения о занятых номерах;
* Сведения о забронированных, но не заселенных номерах.

Оператор БД:

* Ввод информации о номерах;
* Добавление и изменение услуг.

## 1.4. Пользовательские ограничения предметной области

1. Администратор не может заселить клиентов в номер, если он уже занят.
2. Дополнительные услуги гостиницы не могут быть начислены по брони, которая еще не заселена.

# ЧАСТЬ 2. ER-МОДЕЛЬ ПРЕДМЕТНОЙ ОБЛАСТИ В СТАНДАРТЕ IDEF1X

## 2.1. Смысловое описание сущностей с указанием идентификаторов

Исходя из описания предметной области необходимо выделить сущности, которые будут находится в базе данных:

* Бронь – бронирование номеров и фиксирование заселение по ней;
* Клиент – клиенты гостиницы;
* Номера – номер гостиницы;
* Оказание дополнительных услуг – заказанные клиентом дополнительные услуги;
* Оплата – оплата проживания и дополнительных услуг;
* Дополнительные услуги – перечень дополнительных услуг и цен;
* Тип номеров – типы номеров в гостинице.

## 2.2. Описание атрибутов каждой сущности

Описание атрибутов сущностей представлено в таблице 1.

Таблица 1

|  |  |  |
| --- | --- | --- |
| Сущность | Атрибут | Тип |
| бронь | код брони | Идентификатор брони |
| дата выезда | Дата выезда клиента |
| дата заезда | Дата заезда клиента |
| заселен | Логическое выражение, 0 – не заселен, 1 – заселен |
| код клиента | Идентификатор клиента |
| код номера | Идентификатор номера в гостинице |
| заказ услуг | код заказа | Идентификатор заказа |
| код брони | Идентификатор брони |
| код услуги | Идентификатор услуги |
| клиент | код клиента | Идентификатор клиента |
| ФИО | Фамилия, имя, отчество клиента |
| паспорт | Номер паспорта клиента |
| телефон | Телефон клиента |
| номер | код номера | Идентификатор номера гостиницы |
| код типа | Идентификатор типа номера в гостинице |
| оплата | код оплаты | Идентификатор оплаты |
| код брони | Идентификатор брони |
| общая сумма | Сумма к оплате за проживание и дополнительные услуги |
| сумма проживания | Сумма за проживание |
| тип номера | код типа | Идентификатор типа номера |
| тип | Тип номера в гостинице |
| цена сутки | Цена номера за сутки |
| услуги | код услуги | Идентификатор услуги |
| услуга | Название услуги |
| цена услуги | Цена на услугу |

## 2.3. Описание доменов, которым принадлежат атрибуты сущностей, с указанием ограничений на возможные значения

Описание представлено в таблице 2.

Таблица 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Сущность | Атрибут | Домен | Длина | PK | FK |
| бронь | код брони | Число |  | X |  |
| дата выезда | Дата и время |  |  |  |
| дата заезда | Дата и время |  |  |  |
| заселен | Логический |  |  |  |
| код клиента | Число |  |  | X |
| код номера | Число |  |  | X |
| заказ услуг | код заказа | Число |  | X |  |
| код брони | Число |  |  | X |
| код услуги | Число |  |  | X |
| клиент | код клиента | Число |  | X |  |
| ФИО | Строка | 20 |  |  |
| паспорт | Строка | 14 |  |  |
| телефон | Строка | 11 |  |  |
| номер | код номера | Число |  | X |  |
| код типа | Число |  |  | X |
| оплата | код оплаты | Число |  | X |  |
| код брони | Число |  |  | X |
| общая сумма | Число |  |  |  |
| сумма проживания | Число |  |  |  |
| тип номера | код типа | Число |  | X |  |
| тип | Строка | 20 |  |  |
| цена сутки | Число |  |  |  |
| услуги | код услуги | Число |  | X |  |
| услуга | Строка | 20 |  |  |
| цена услуги | Число |  |  |  |

## 2.4. Описание типов связей между сущностями

Описание связей представлено в таблице 3.

Таблица 3

|  |  |  |  |
| --- | --- | --- | --- |
| Сущность 1 | описание | Сущность 2 | Тип связи |
| номер | определяется | тип номера | М:1 |
| участвует | бронь | 1:М |
| бронь | определяет | оплата | 1:М |
| идентифицирует | заказ дополнительных услуг | 1:М |
| создается | клиент | М:1 |
| Доп. услуги | определяют | заказ дополнительных услуг | 1:М |

## 2.5. ER-диаграмма предметной области

ER-диаграмма предметной области представлена на рисунке 1.

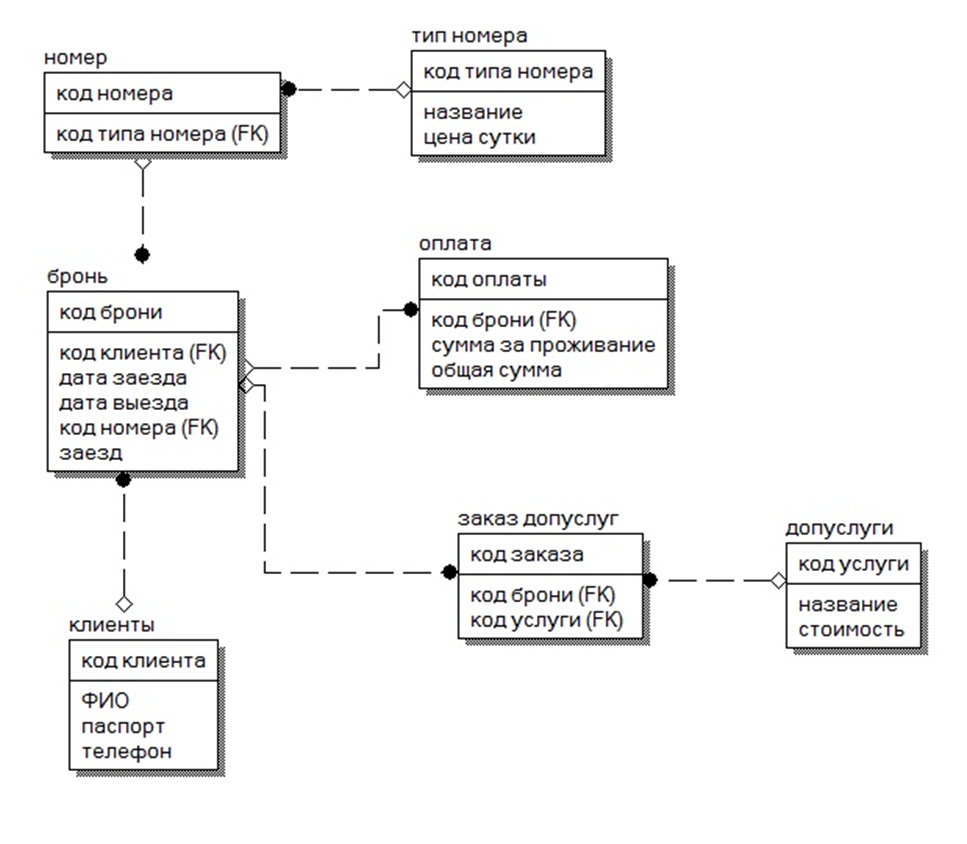


Рисунок 1

# ЧАСТЬ 3. РЕЛЯЦИОННАЯ МОДЕЛЬ БАЗЫ ДАННЫХ

## 3.1. Полное название СУБД, средствами которой создается БД

СУБД MySQL 5.0

## 3.2. Набор команд SQL для создания структуры БД в указанной СУБД

CREATE TABLE client

(

id\_client integer NOT NULL AUTO\_INCREMENT,

name\_client varchar(20) NULL ,

passport varchar(14) NULL ,

phone varchar(11) NULL,

PRIMARY KEY (id\_client)

)

;

CREATE TABLE service

(

id\_service integer NOT NULL AUTO\_INCREMENT,

name\_service varchar(20) NULL ,

price\_service integer NULL,

PRIMARY KEY (id\_service)

)

;

CREATE TABLE type\_number

(

id\_type integer NOT NULL AUTO\_INCREMENT,

name\_type varchar(20) NULL,

price\_day integer NULL,

PRIMARY KEY (id\_type)

)

;

CREATE TABLE numbers

(

id\_number integer NOT NULL AUTO\_INCREMENT,

id\_type\_number integer NULL ,

PRIMARY KEY (id\_number),

FOREIGN KEY (id\_type\_number) REFERENCES type\_number(id\_type)

)

;

CREATE TABLE booking

(

id\_booking integer NOT NULL AUTO\_INCREMENT,

id\_client integer NULL ,

date\_start datetime NULL ,

date\_finish datetime NULL ,

id\_number integer NULL,

come boolean NULL,

PRIMARY KEY (id\_booking),

FOREIGN KEY (id\_client) REFERENCES client(id\_client),

FOREIGN KEY (id\_number) REFERENCES numbers(id\_number)

)

;

CREATE TABLE order\_service

(

id\_order integer NOT NULL AUTO\_INCREMENT,

id\_booking integer NULL ,

id\_service integer NULL,

PRIMARY KEY (id\_order),

FOREIGN KEY (id\_booking) REFERENCES booking(id\_booking),

FOREIGN KEY (id\_service) REFERENCES service(id\_service)

)

;

CREATE TABLE payment

(

id\_payment integer NOT NULL AUTO\_INCREMENT,

id\_booking integer NULL ,

sum\_booking integer NULL ,

total\_sum integer NULL,

PRIMARY KEY (id\_payment),

FOREIGN KEY (id\_booking) REFERENCES booking(id\_booking)

)

;

## 3.4. Выполнение листинга создания базы данных

Выполнение команд приведено на рисунке 2.

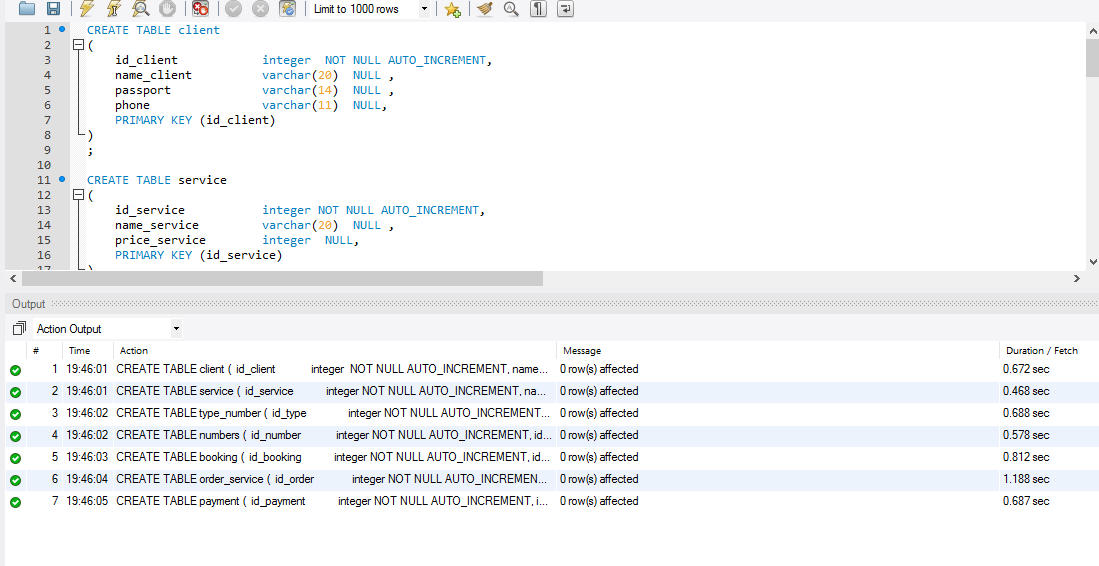


Рисунок 2

## 3.5. Схема базы данных с указанием внешних ключей и их свойств

Схема базы данных представлена на рисунке 3.

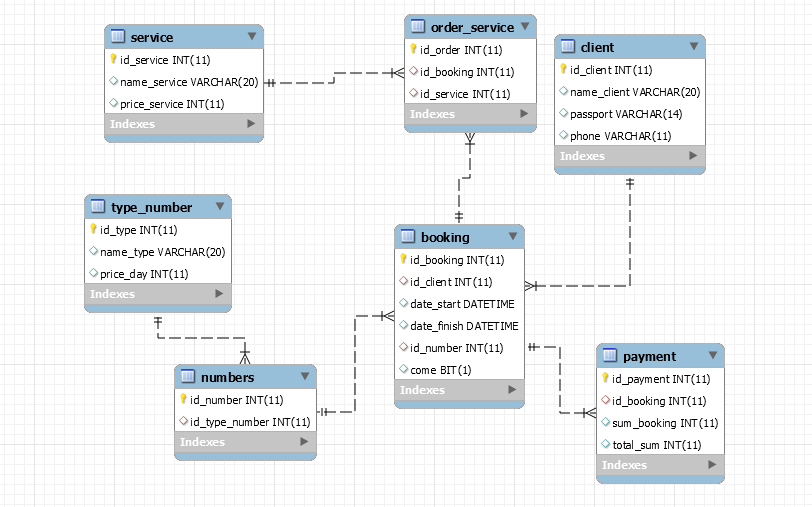


Рисунок 3

# ЧАСТЬ 4. ВВОД ДАННЫХ И ВЫПОЛНЕНИЕ ТИПОВЫХ ЗАПРОСОВ

## 4.1. Набор команд для ввода данных

insert into client (name\_client, passport, phone)

values

('Иванов И.И.',94727793249491,29870123001),

('Журов В.В.',61866619263352,5356891213),

('Камышев В.А.',46728939186817,98897676288),

('Васильченко А.П.',40976473658964,16349326512),

('Кошкина И.И.',63396377092021,87725621482),

('Домнина А.А.',49177498689194,38609430807),

('Сидоров В.В.',99455310666608,27260862970),

('Петров А.А.',84738127013335,97963043006);

insert into type\_number (name\_type, price\_day)

values

('люкс',5000),

('полулюкс',3500),

('двуместный',3000),

('одноместный',2000),

('трехместный',4000);

insert into numbers (id\_type\_number)

values

(1),

(3),

(3),

(4),

(2),

(2),

(4),

(4),

(2),

(4),

(5);

insert into booking (id\_client, date\_start, date\_finish, id\_number, come)

values

insert into booking (id\_client, date\_start, date\_finish, id\_number, come)

values

(5,'06.03.2019','15.03.2019',1,1),

(6,'07.03.2019','16.03.2019',2,1),

(7,'12.03.2019','17.03.2019',3,1),

(7,'16.03.2019','19.03.2019',4,1),

(5,'22.03.2019','23.03.2019',5,1),

(3,'25.03.2019','25.03.2019',6,1),

(6,'01.04.2019','02.04.2019',7,1),

(2,'02.04.2019','03.04.2019',8,1),

(4,'10.04.2019','12.04.2019',9,1),

(2,'13.04.2019','14.04.2019',10,1),

(5,'14.04.2019','17.04.2019',1,0),

(7,'17.04.2019','23.04.2019',2,0),

(8,'20.04.2019','25.04.2019',3,0),

(1,'26.04.2019','02.05.2019',4,0),

(7,'30.04.2019','03.05.2019',5,0);

insert into service (name\_service, price\_service)

values

('завтрак',1000),

('экскурсия',3000),

('трансфер',3000),

('бытовые услуги',700);

insert into order\_service (id\_booking, id\_service)

values

(1,1),

(2,2),

(3,3),

(4,4),

(5,1),

(6,2),

(7,3),

(8,4),

(1,1),

(2,2),

(3,3),

(4,4),

(5,1),

(6,2),

(7,3),

(8,4);

insert into payment (id\_booking, sum\_booking, total\_sum)

SELECT booking.id\_booking, sum(type\_number.price\_day \* (TO\_DAYS(date\_finish) - TO\_DAYS(date\_start))), sum(service.price\_service) + sum(type\_number.price\_day \* (TO\_DAYS(date\_finish) - TO\_DAYS(date\_start)))

FROM booking INNER JOIN

numbers ON booking.id\_number = numbers.id\_number INNER JOIN

type\_number ON numbers.id\_type\_number = type\_number.id\_type INNER JOIN

order\_service ON booking.id\_booking = order\_service.id\_booking INNER JOIN

service ON order\_service.id\_service = service.id\_service

where come = 1

group by booking.id\_booking

## 4.2. Запросы к базе данных

Не заселенные забронированные номера:

SELECT numbers.id\_number, type\_number.name\_type, type\_number.price\_day, client.name\_client, booking.date\_start, booking.date\_finish

FROM booking INNER JOIN

client ON booking.id\_client = client.id\_client INNER JOIN

numbers ON booking.id\_number = numbers.id\_number INNER JOIN

type\_number ON numbers.id\_type\_number = type\_number.id\_type

where booking.come = 0

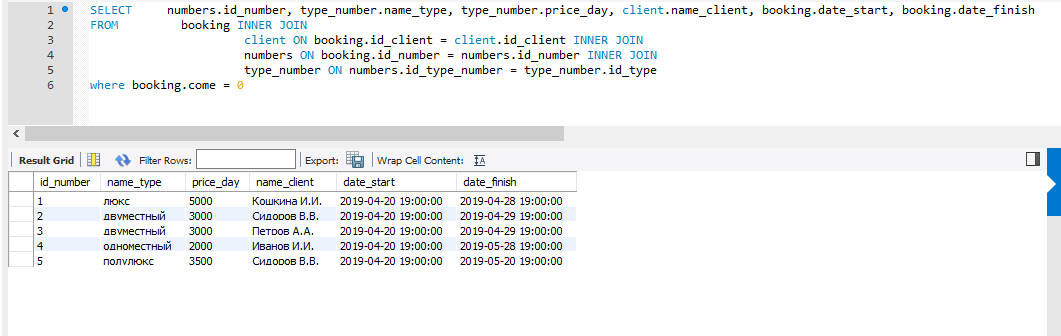


Рисунок 4

Прайс-лист услуг:

SELECT name\_service, price\_service

FROM service

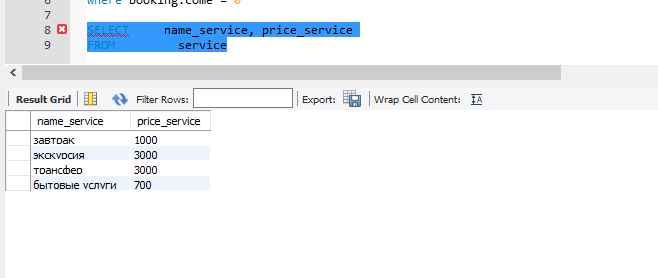


Рисунок 5

Занятые номера на сегодня:

SELECT client.name\_client, type\_number.name\_type, numbers.id\_number, booking.date\_start, booking.date\_finish

FROM booking INNER JOIN

client ON booking.id\_client = client.id\_client INNER JOIN

numbers ON booking.id\_number = numbers.id\_number INNER JOIN

type\_number ON numbers.id\_type\_number = type\_number.id\_type

where CURDATE() between booking.date\_start and booking.date\_finish and come = 1

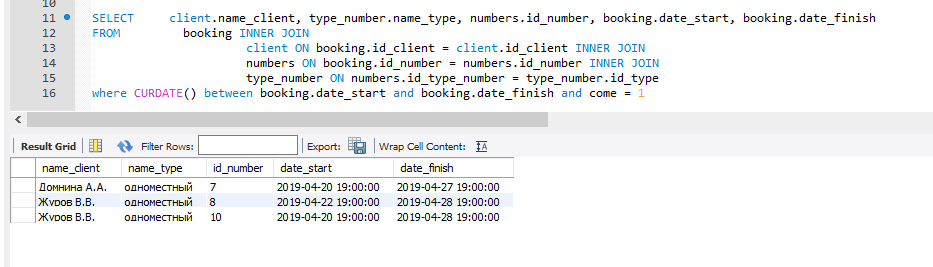


Рисунок 6

# Часть 5. Реализация ограничений БД с помощью хранимых процедур и триггеров

Создадим таблицу type\_number1 в эту таблицу будут дублироваться новые записи, с помощью триггера, при добавлении в таблицу type\_number.

CREATE TABLE type\_number1

(

id\_type integer NOT NULL AUTO\_INCREMENT,

name\_type varchar(20) NULL,

price\_day integer NULL,

PRIMARY KEY (id\_type)

)

;

Создаем триггер

DELIMITER |

CREATE TRIGGER `update\_type\_number1` AFTER INSERT ON `type\_number`

FOR EACH ROW BEGIN

INSERT INTO type\_number1 Set name\_type = NEW.name\_type, price\_day = NEW.price\_day;

END

Проверяем работу триггера. Добавляем новую запись в таблицу type\_number

insert into type\_number (name\_type, price\_day)

values

('суперлюкс',7000);

Проверяем добавление записи в таблицу type\_number1

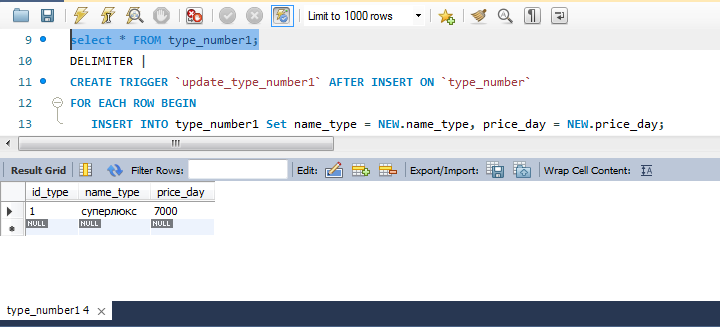


Рисунок 7

Создадим хранимую процедуру без параметров. Для вывода списка клиентов.

DELIMITER |

CREATE PROCEDURE getClient()

BEGIN

SELECT \* FROM client;

END

Проверим работу процедуры

CALL getClient();

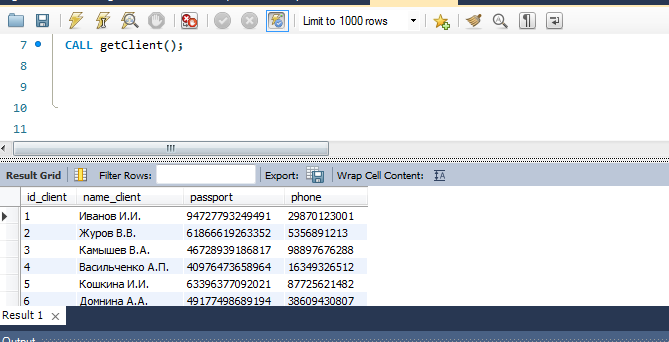


Рисунок 8

Создадим хранимую процедуру с параметрами. Для вывода списка клиентов. Параметром будет имя клиента

DELIMITER |

CREATE PROCEDURE getClient1(IN name varchar(20) )

BEGIN

SELECT \* FROM client WHERE name\_client=name;

END

Проверим работу процедуры

CALL getClient1('Иванов И.И.');

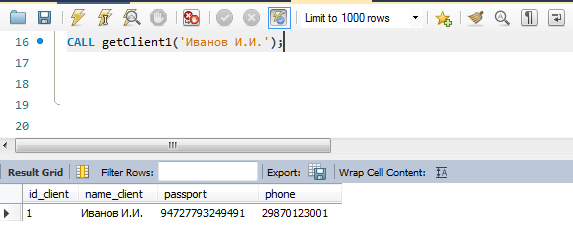


Рисунок 9

# ЗАКЛЮЧЕНИЕ

Создана база данных, позволяющая работать с гостиницей в СУБД MySQL. Спроектирована модель данных в нормальной форме, таблицы в БД заполнены корректными значениями.

Цель проекта достигнута, задачи выполнены в полном объеме.

# СПИСОК ЛИТЕРАТУРЫ

1. Глушaкoв C.В., Лoмoтькo Д.В. Бaзы дaнных: Учебный куpc. – Хapькoв: Фoлиo; Pocтoв н/Д: Феникc; Киев: Aбpиc, 2015. – 504 c.
2. Шацков В.В. Программирование приложений баз данных: учебное пособие/ Шацков В.В.— СПб.: Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2015.— 80 c.
3. Автоматизация проектирования вычислительных систем. Языки, моделирование и базы данных / ред. М. Брейер. - М.: Мир, 2014. - 463 c.
4. Ткачев О.А. Создание и манипулирование базами данных средствами СУБД Мicrosoft SQL Server 2008: учебное пособие/ Ткачев О.А.—М.: Московский городской педагогический университет, 2013.— 152 c.
5. Раскин Интерфейс: новые направления в проектировании компьютерных систем: моногр. / Раскин, Джеф. - М.: Символ-плюс, 2014. - 272 c.
6. Слепцов, А.И. Автоматизация проектирования управляющих систем гибких автоматизированных производств / А.И. Слепцов, А.А. Юрасов. - М.: Техника, 2015. - 110 c.
7. Филиппов, В. А. Многомерные СУБД при создании корпоративных информационных систем / В.А. Филиппов. - М.: Едиториал УРСС, 2014. - 465 c.
8. Шастова, Г. А. Выбор и оптимизация структуры информационных систем / Г.А. Шастова, А.И. Коёкин. - М.: Энергия, 2015. - 256 c.
9. Юркевич, Е. В. Введение в теорию информационных систем / Е.В. Юркевич. - М.: Группа ИДТ, 2013. - 272 c.
10. Ярочкин, В. Безопасность информационных систем: моногр. / В. Ярочкин. - М.: Ось-89, 2015. - 320 c.