

ГУАП

КАФЕДРА № 41

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

Доц, к.т.н.

должность, уч. степень, звание

подпись, дата

Турнецкая Е.Л

инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ

ТЕСТИРОВАНИЕ БАЗЫ ДАННЫХ

по курсу: Инженер по тестированию. Основы тестирования программного обеспечения

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ гр. №

2346

подпись, дата

А. М. Воронцова

Е. Н. Костынюк

С. А. Себелев

М. Н. Федорук

инициалы, фамилия

Санкт-Петербург 2025

1. Цель работы

Приобретение практических навыков составления тестовых запросов к реляционной БД.

2. Описание предметной области

MySQL — это реляционная система управления базами данных (СУБД) с открытым исходным кодом, разработанная компанией Oracle. Она является одной из самых популярных и широко используемых СУБД в мире благодаря своей надежности, производительности и простоте использования.

3. Ход работы

В нашем варианте выделено 3 сущности: Пациент(Patient), Врач(Doctor), Обращение к врачу(Contacting). Для удобства составления инфологической модели (рис. 1) были составлены таблицы 1.1, 1.2, 1.3.

В таблицах приняты следующие сокращения:

NN (Not Null) – поле с обязательным заполнением,

AI (Avto Increment) – поле типа счетчик,

ПК – первичный ключ,

ВК – внешний ключ.

Таблица 1.1

Определение атрибутов сущности Пациент (Patient)

Атрибуты	Тип данных	Размер(бит)	Комментарий
patientId	INT	32	Код пациента, NN, AI, ПК
patientSurname	VARCHAR	45	Фамилия, NN
patientName	VARCHAR	45	Имя, NN
patientSex	VARCHAR	45	Пол
patientBirthDate	DATE	24	Год рождения

Таблица 1.2

Определение атрибутов сущности Врач (Doctor)

Атрибуты	Тип данных	Размер(бит)	Комментарий
docId	INT	32	Код врача, NN, AI, ПК
docSurname	VARCHAR	45	Фамилия, NN
docSpecialization	VARCHAR	45	Специализация, NN
docCategory	VARCHAR	45	Категория, NN

Таблица 1.3

Определение атрибутов сущности Обращение к врачу (Contacting)

Атрибуты	Тип данных	Размер(бит)	Комментарий

contactId	INT	32	Код обращения, NN, AI
patientId	INT	32	Код пациента, NN, BK
docId	INT	32	Код врача, NN, BK
contactDate	DATETIME	48	Дата обращения, NN
contactPrice	DECIMAL	10, 2	Оплата за прием, NN

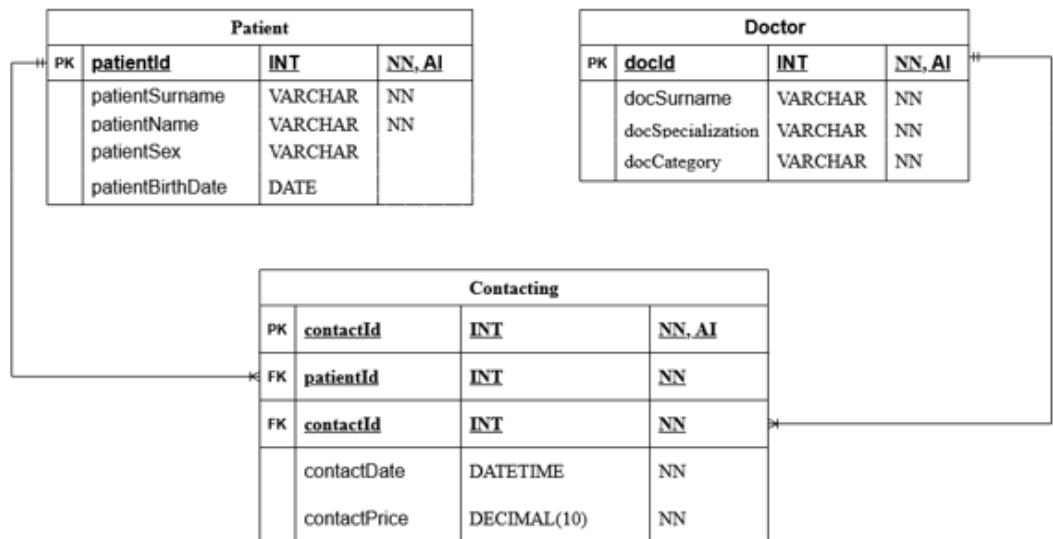


Рис. 1 - инфологическая модель

В качестве среды разработки была выбран IDE VScode с расширением SQLtools, потому что оно позволяет произвести бесконечное число запросов в отличие от онлайн платформа <https://sqlonline.com/>.

Ниже код для создания БД

```

CREATE TABLE Patient(
    patientId INT AUTO_INCREMENT PRIMARY KEY,
    patientName VARCHAR(45) Not Null,
    patientSurname VARCHAR(45) Not Null,
    patientSex VARCHAR(45),
    patientBirthDate DATE
);

CREATE TABLE Doctor(
    docId INT AUTO_INCREMENT PRIMARY KEY,
    docSurname VARCHAR(45) Not Null,
    docSpecializationn VARCHAR(45) Not Null,
    docCategory VARCHAR(45) Not Null
);

CREATE TABLE Contacting(
    contactId INT AUTO_INCREMENT PRIMARY KEY,
    docId INT Not Null,
    patientId INT Not Null,
    contactPrice DECIMAL(10) Not Null,

```

```

        contactDate DATETIME Not Null,
        FOREIGN KEY (patientId) REFERENCES Patient(patientId) ON
DELETE CASCADE,
        FOREIGN KEY (docId) REFERENCES Doctor(docId) ON DELETE
CASCADE
);

```

Увидеть созданные таблицы можно в MySQL Workbench рис. 2.

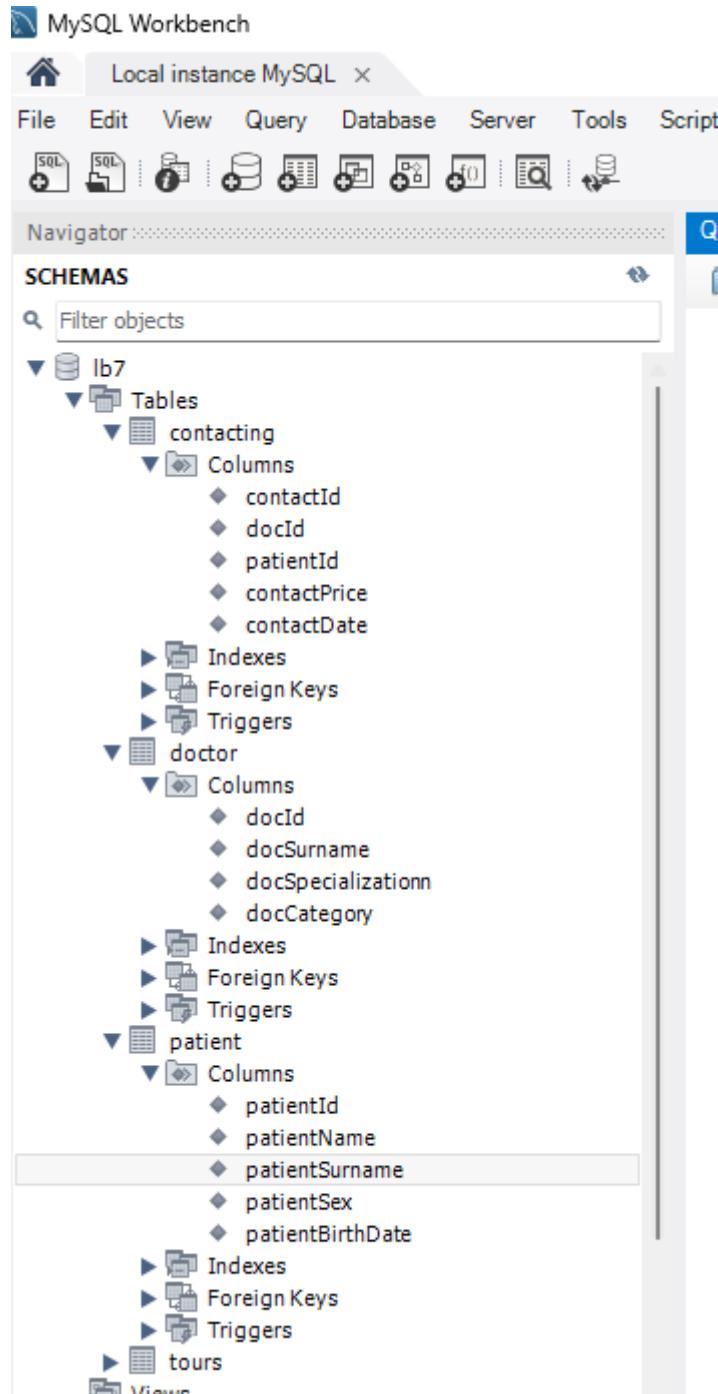


Рис. 2 – создание таблиц

Таблицы надо заполнить. Код для заполнения:

```

INSERT INTO Patient (patientId, patientName, patientSurname,
patientSex, patientBirthDate)
VALUES

```

```

(1, 'Денис', 'Амитов', 'Мужской', '2003-02-01'),
(2, 'Ксения', 'Васенина', 'Женский', '2012-01-10'),
(3, 'София', 'Воронцова', 'Женский', '1999-03-20'),
(4, 'Анна', 'Воронцова', 'Женский', '2004-06-17'),
(5, 'Георгий', 'Герасимов', 'Мужской', '1980-06-25'),
(6, 'Карина', 'Голант', 'Женский', '2000-11-25'),
(7, 'Дарья', 'Ильина', '', '2005-06-01');

INSERT INTO Doctor (docId, docSurname, docSpecializationn,
docCategory)
VALUES
(1, 'Исаева', 'Психиатр', 'Вторая'),
(2, 'Козлова', 'Кардиолог', 'Первая'),
(3, 'Костынюк', 'Спортивный врач', 'Высшая'),
(4, 'Кроян', 'Дерматолог', 'Высшая'),
(5, 'Лобанова', 'Невролог', 'Первая'),
(6, 'Лысак', 'Невролог', 'Вторая'),
(7, 'Матюшенко', 'Анестезиолог', 'Вторая');

INSERT INTO Contacting (contactId, docId, patientId, contactPrice,
contactDate)
VALUES
(1, 3, 4, 300, '2025-11-23 12:00:00'),
(2, 3, 5, 900, '2025-11-23 12:30:00'),
(3, 6, 3, 500, '2025-11-23 12:30:00'),
(4, 5, 1, 999, '2025-11-23 10:15:00'),
(5, 4, 2, 891, '2025-11-23 17:45:00'),
(6, 1, 7, 333, '2025-11-23 14:15:00'),
(7, 2, 6, 888, '2025-11-23 16:20:00');

```

Для первичного осмотра можно вызвать таблицы из VScode (рис. 3) или просмотреть в MySQL Workbench (рис. 4).

The screenshot shows the VSCode interface with four tables displayed as grids:

- Contacting:** Columns: contactId, docId, patientId, contactPrice, contactDate. Rows: 1, 2, 3, 4, 5, 6, 7.
- Doctor:** Columns: docId, docSurname, docSpecializationn, docCategory. Rows: 1, 2, 3, 4, 5, 6, 7.
- Patient:** Columns: contactId, docId, patientId, contactPrice, contactDate. Rows: 1, 2, 3, 4, 5, 6, 7.
- Doctor:** Columns: docId, docSurname, docSpecializationn, docCategory. Rows: 1, 2, 3, 4, 5, 6, 7.

Рис. 3 – таблица в VScode

The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' tree view is expanded to show the 'lb7' schema, which contains four tables: 'contacting', 'doctor', 'patient', and 'tours'. The 'patient' table is selected. On the right, the 'Result Grid' displays the results of the query 'SELECT * FROM lb7.patient;'. The grid has columns: patientId, patientName, patientSurname, patientSex, and patientBirthDate. The data consists of 7 rows:

	patientId	patientName	patientSurname	patientSex	patientBirthDate
1	Денис	Амитов	Мужской	2003-02-01	
2	Ксения	Васенина	Женский	2012-01-10	
3	София	Воронцова	Женский	1999-03-20	
4	Анна	Воронцова	Женский	2004-06-17	
5	Георгий	Герасимов	Мужской	1980-06-25	
6	Карина	Голант	Женский	2000-11-25	
7	Дарья	Ильина		2005-06-01	

Рис. 4 – таблица в MySQL Workbench

Теперь для тестирования БД создадим 5 запросов. Результаты их исполнения видны на рис. 4-8. Код приведён ниже:

```
-- Подсчитать общее количество приемов врачей
SELECT
    count(contactId) AS 'Количество приемов врачей'
FROM
    contacting;

-- нахождение пациентов Воронцовых
SELECT
    patientSurname,
    patientName
FROM
    patient
WHERE
    patientSurname like 'Воронцова';

-- Вывести услуги врачей стоимостью большее 400
SELECT
    d.docSurname AS 'Врач',
    d.docSpecializationn AS 'Специализация',
    p.patientSurname AS 'Пациент',
    c.contactPrice AS 'Стоимость приёма',
    TIME(c.contactDate) AS 'Время'
FROM
    contacting AS c
INNER JOIN doctor as d ON c.docId = d.docId
INNER JOIN patient as p ON c.patientId = p.patientId
GROUP BY
    d.docSurname,
    d.docSpecializationn,
    p.patientSurname,
```

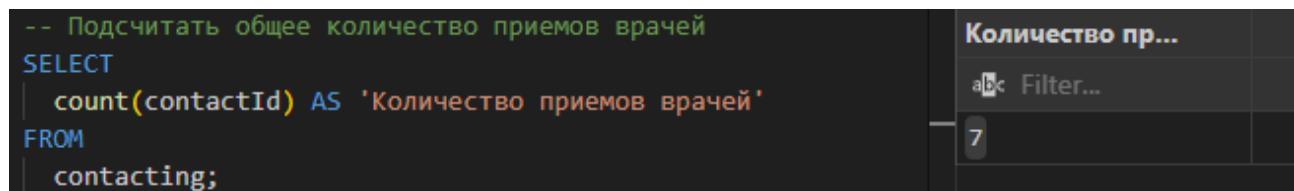
```

c.contactPrice,
c.contactDate
HAVING
    c.contactPrice > 400;

-- Вывести услуги врачей стоимостью большее 400 и принимают позже 12:30
SELECT
    d.docSurname AS 'Врач',
    d.docSpecialization AS 'Специализация',
    p.patientSurname AS 'Пациент',
    c.contactPrice AS 'Стоимость приёма',
    TIME(c.contactDate) AS 'Время'
FROM
    contacting AS c
    INNER JOIN doctor AS d ON c.docId = d.docId
    INNER JOIN patient AS p ON c.patientId = p.patientId
WHERE
    c.contactPrice > 400
    AND TIME(c.contactDate) > '12:30:00';

-- Врачи с максимальным количеством приемов
SELECT
    d.docId,
    d.docSurname AS 'Фамилия врача',
    d.docSpecialization AS 'Специализация',
    COUNT(c.contactId) AS 'Количество приёмов'
FROM
    contacting AS c
    INNER JOIN doctor AS d ON c.docId = d.docId
GROUP BY
    d.docId,
    d.docSurname,
    d.docSpecialization
HAVING
    COUNT(c.contactId) = (
        SELECT
            MAX(appointment_count)
        FROM
            (
                SELECT
                    COUNT(contactId) AS appointment_count
                FROM
                    contacting
                GROUP BY
                    docId
            ) AS tmp
    );

```



Количество пр...
abc Filter...
7

Рис. 4 – Первый запрос: подсчитать общее количество приемов врачей

```
-- нахождение пациентов Воронцовых
SELECT
    patientSurname,
    patientName
FROM
    patient
WHERE
    patientSurname like 'Воронцова';
```

patientSurname	patientName
Воронцова	София
Воронцова	Анна

Рис. 5 – Второй запрос: нахождение пациентов Воронцовых

```
-- Вывести услуги врачей стоимостью больше 400
SELECT
    d.docSurname AS 'Врач',
    d.docSpecializationn AS 'Специализация',
    p.patientSurname AS 'Пациент',
    c.contactPrice AS 'Стоймость приёма',
    TIME(c.contactDate) AS 'Время'
FROM
    contacting AS c
INNER JOIN doctor as d ON c.docId = d.docId
INNER JOIN patient as p ON c.patientId = p.patientId
GROUP BY
    d.docSurname,
    d.docSpecializationn,
    p.patientSurname,
    c.contactPrice,
    c.contactDate
HAVING
    c.contactPrice > 400;
```

Врач	Специализация	Пациент	Стоймость при...	Время
Костынок	Спортивный врач	Герасимов	900	+12:30:00.000000
Лысак	Невролог	Воронцова	500	+12:30:00.000000
Лобанова	Невролог	Амитов	999	+10:15:00.000000
Кроян	Дерматолог	Васенина	891	+17:45:00.000000
Козлова	Кардиолог	Голант	888	+16:20:00.000000

Рис. 6 – Третий запрос: вывести услуги врачей стоимостью больше 400

```
-- Вывести услуги врачей стоимостью больше 400 и принимают позже 12:30
SELECT
    d.docSurname AS 'Врач',
    d.docSpecializationn AS 'Специализация',
    p.patientSurname AS 'Пациент',
    c.contactPrice AS 'Стоймость приёма',
    TIME(c.contactDate) AS 'Время'
FROM
    contacting AS c
INNER JOIN doctor as d ON c.docId = d.docId
INNER JOIN patient as p ON c.patientId = p.patientId
WHERE
    c.contactPrice > 400
    AND TIME(c.contactDate) > '12:30:00';
```

Врач	Специализация	Пациент	Стоймость при...	Время
Кроян	Дерматолог	Васенина	891	+17:45:00.000000
Козлова	Кардиолог	Голант	888	+16:20:00.000000

Рис. 7 – Четвёртый запрос: Вывести услуги врачей стоимостью большее 400 и принимают позже 12:30

```
-- Врачи с максимальным количеством приемов
SELECT
    d.docId,
    d.docSurname AS 'Фамилия врача',
    d.docSpecializationn AS 'Специализация',
    COUNT(c.contactId) AS 'Количество приёмов'
FROM
    contacting AS c
INNER JOIN doctor AS d ON c.docId = d.docId
GROUP BY
    d.docId,
    d.docSurname,
    d.docSpecializationn
HAVING
    COUNT(c.contactId) =
        (
            SELECT
                MAX(appointment_count)
            FROM
                (
                    SELECT
                        COUNT(contactId) AS appointment_count
                    FROM
                        contacting
                    GROUP BY
                        docId
                ) AS tmp
        );
```

docId	Фамилия врача	Специализация	Количество приёмов
3	Костынок	Спортивный врач	2

Рис. 8 – Пятый запрос: врачи с максимальным количеством приемов

4. Вывод

В ходе выполнения практической работы по тестированию базы данных были успешно приобретены и закреплены практические навыки работы с реляционными базами

данных на всех этапах их жизненного цикла.

Полученные и закреплённые навыки:

- Проектирование БД с помощью таблиц со сущностями и инфологической модели.
- Реализация БД MySQL на персональном компьютере посредством IDE VsCode
- Наполнение БД
- Тестирование запросами

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. DeepSeek – URL: <https://chat.deepseek.com>
2. “MySQL. Сборник рецептов” - Поль Дюбуа
3. Учим Базы Данных за 1 час! #От Профессионала - Хауди Хо™ - Просто о мире IT!
– URL: <https://youtu.be/ZKU7-ktaa2o?si=tGcuWa9-hWp8kBIZ>