Министерство образования и науки Российской Федерации

Севастопольский государственный университет

Институт информационных технологий

Кафедра ИС

# ОТЧЕТ

по лабораторной работе №1

ИЗУЧЕНИЕ ОСНОВ ЯЗЫКА МАНИПУЛИРОВАНИЯ ДАННЫМИ SQL НА БАЗЕ СЕРВЕРА FIREBIRD

Выполнил:

ст. гр. ИС/б-21-2-о

Мовенко К. М.

Проверил:

Гончаренко Д. Г.

Севастополь

2023

# ЦЕЛЬ РАБОТЫ

Приобрести практические навыки по работе с реляционными базами данных, изучить архитектуру сервера Firebird, основы языка запросов SQL, научиться создавать таблицы и осуществлять элементарные выборки.

# ВАРИАНТ ЗАДАНИЯ

Вариант задания – 12. Необходимо создать ряд таблиц, содержащих данные об осуждённых лицах. Структура системы представлена ниже (Рисунок 2.1).



Рисунок 2.1 – Вариант задания

# ХОД РАБОТЫ

## Создание тестовой БД

Перед началом работы над заданием по варианту была создана тестовая база данных test.fdb. Затем в ней была создана таблица Employee, содержащая информацию о сотрудниках некоторой компании (Рисунок 3.1). Данные о колонках таблицы были взяты из методических указаний.

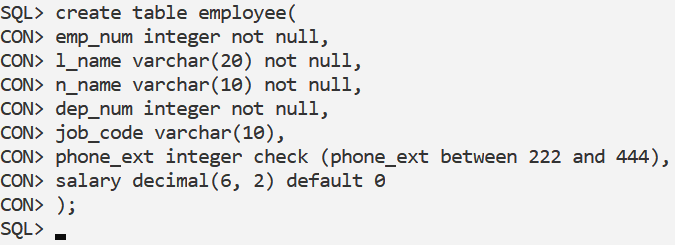


Рисунок 3.1 – Создание таблицы

Далее с помощью функции insert в таблицу было внесено десять новых кортежей (Рисунок 3.2), после чего все строки таблицы были выведены на экран (Рисунок 3.3).

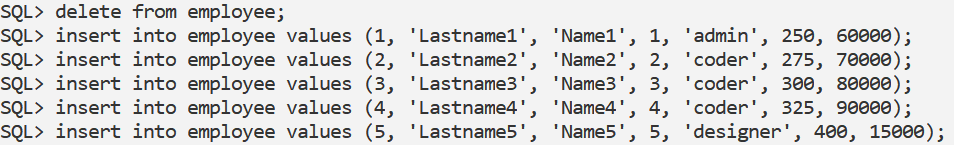


Рисунок 3.2 – Ввод кортежей

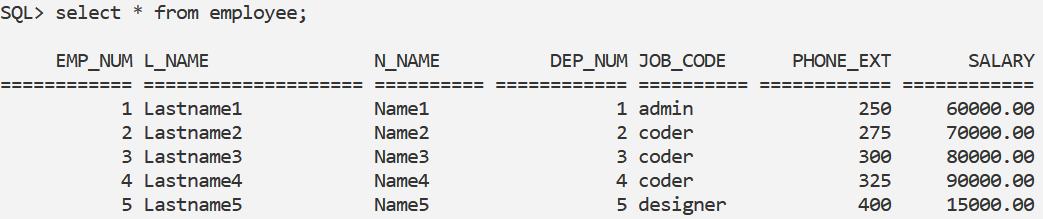


Рисунок 3.3 – Вывод таблицы Employee

## Задание по варианту

Была создана новая база данных lab1.fdb, в ней была создана таблица Person (Рисунок 3.4), содержащая часть данных о личностях заключённых.

Поля таблицы были выбраны в соответствии с вариантом задания.

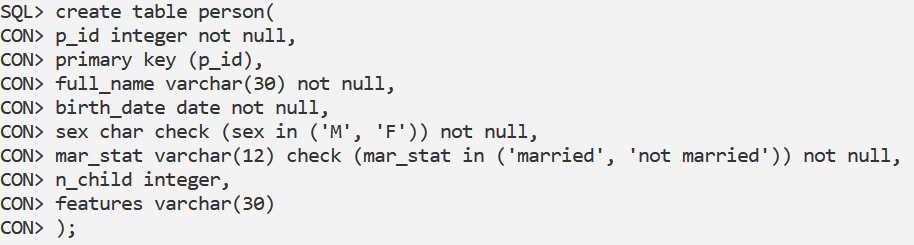


Рисунок 3.4 – Создание таблицы Person (Персона)

В таблицу Person было занесено десять новых кортежей (Рисунок 3.5), после чего все они были выведены с помощью запроса select \*, выводящего все строки таблицы (Рисунок 3.6).

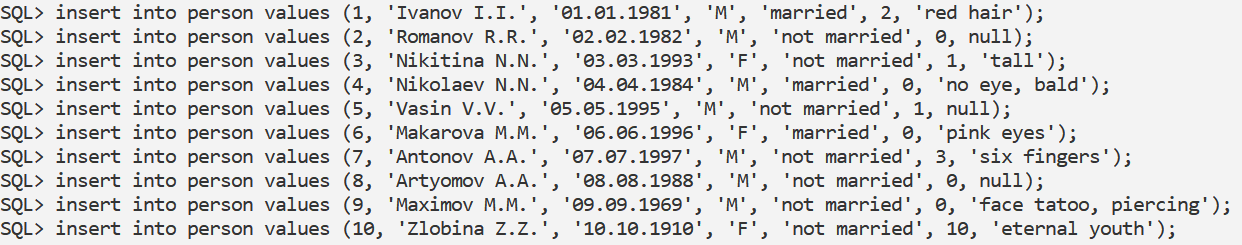


Рисунок 3.5 – Ввод данных о заключённых

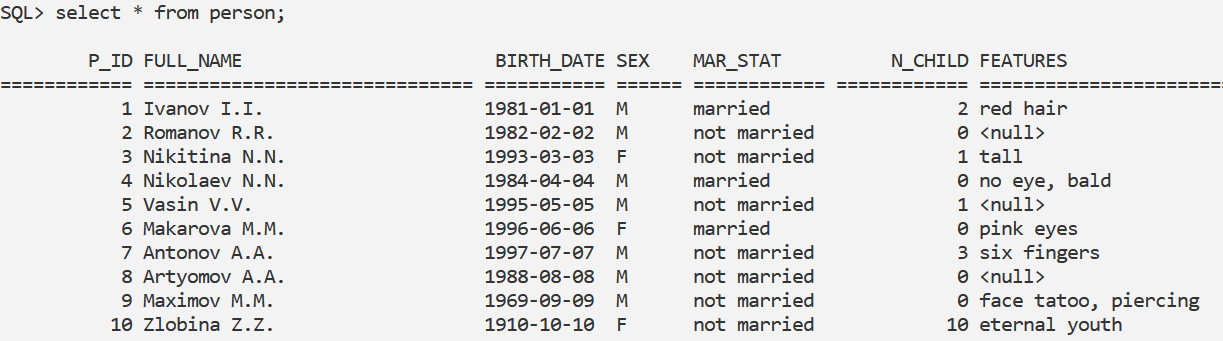


Рисунок 3.6 – Вывод таблицы Person

Далее была создана таблица Cases (Дела), в которой содержатся описания уголовных дел заключённых в виде строк. Для наглядности заполненная таблица была выведена на экран (Рисунок 3.7).

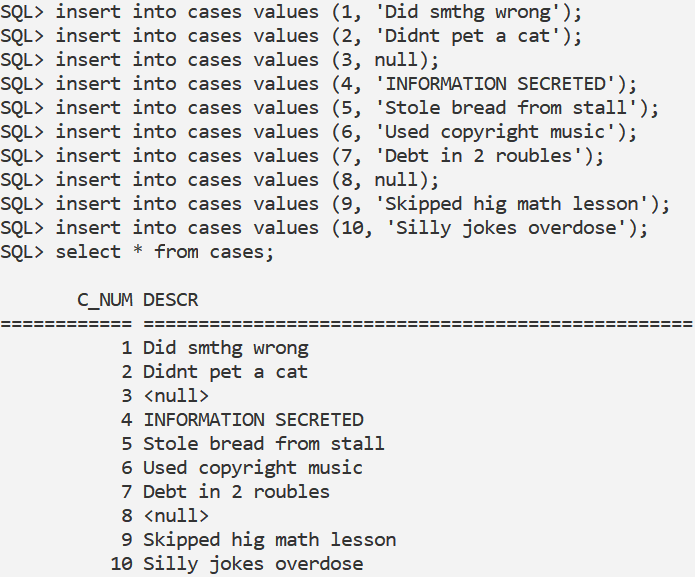


Рисунок 3.7 – Заполнение и вывод таблицы Cases

Те же действия были проведены и для таблицы Alias (Псевдоним), в которой собраны псевдонимы заключённых (Рисунок 3.8).

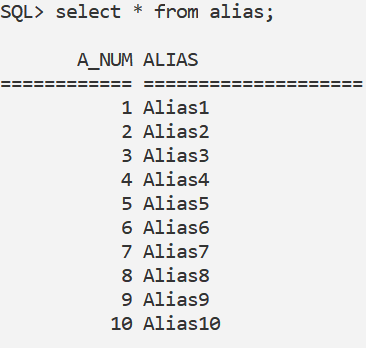


Рисунок 3.8 – Таблица Alias

## Запросы

Для демонстрации различных видов запросов была выбрана таблица Person. После запроса select \* на вывод всех строк таблицы был сделан запрос на вывод той же таблицы, но с порядком столбцов, отличным от исходного (Рисунок 3.9).

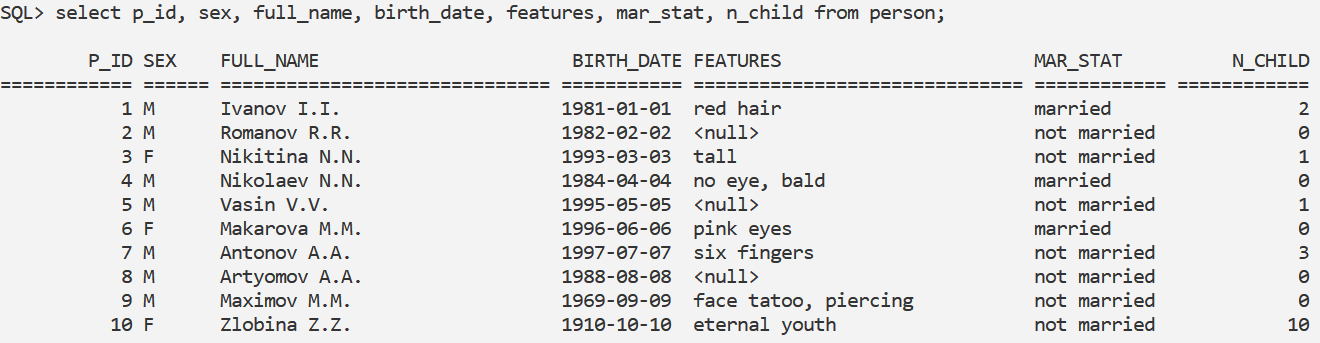


Рисунок 3.9 – Запрос с заданными столбцами

Далее было продемонстрировано действие модификатора DISTINCT. Был сделан запрос на вывод всех уникальных особых примет (Рисунок 3.10).

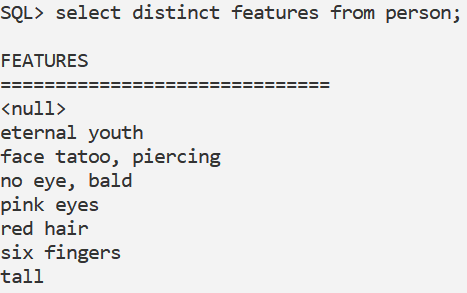


Рисунок 3.10 – Запрос с модификатором distinct

Далее были сделаны запросы с предложением WHERE. Сначала было задано простое условие – вывести данные о всех заключённых, у кого есть дети (Рисунок 3.11). Затем условие было усложнено – вывести всех заключенных, не состоящих в браке, но имеющих хотя бы одного ребёнка (Рисунок 3.12).

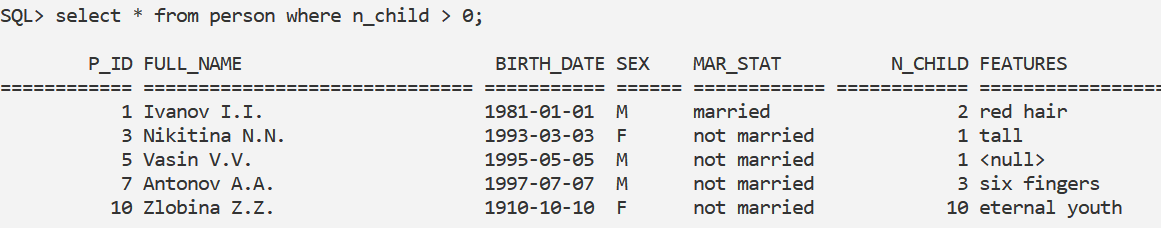


Рисунок 3.11 – Запрос с простым условием

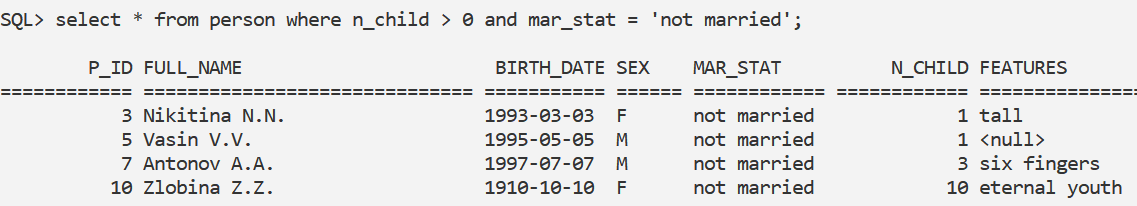


Рисунок 3.12 – Запрос с составным условием

Для демонстрации действия функции BETWEEN был сделан запрос на вывод всех заключённых, родившихся в период 1980-1990 гг. (Рисунок 3.13).

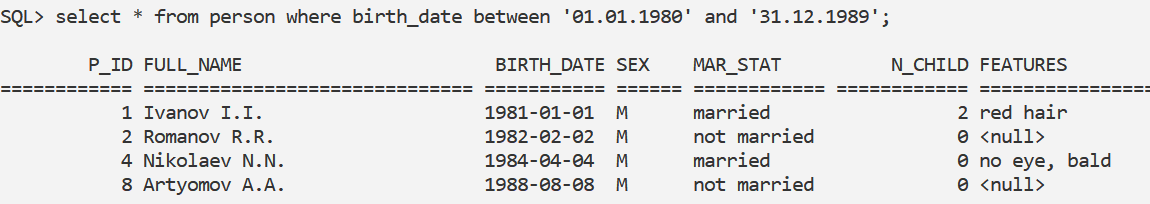


Рисунок 3.13 – Запрос с функцией BETWEEN

Для демонстрации действия функции LIKE был сделан запрос на вывод всех заключённых, у которых указаны две особые приметы, разделённые запятой (Рисунок 3.14).

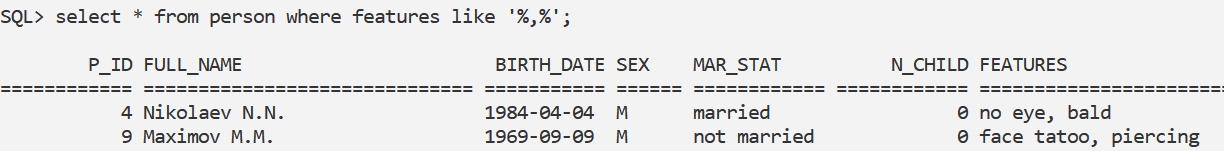


Рисунок 3.14 – Запрос с функцией LIKE

Для демонстрации действия функции IS NULL был сделан запрос на вывод всех заключённых, у которых не указано особых примет (Рисунок 3.15).

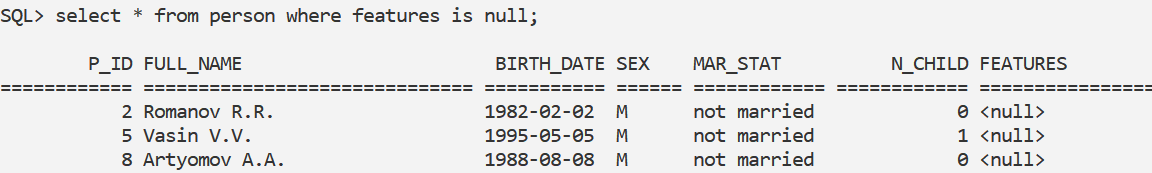


Рисунок 3.15 – Запрос с функцией IS NULL

Для демонстрации действия функции IN был сделан запрос на вывод всех заключённых cо значениями ID 1, 2, 4 или 8 (Рисунок 3.16).

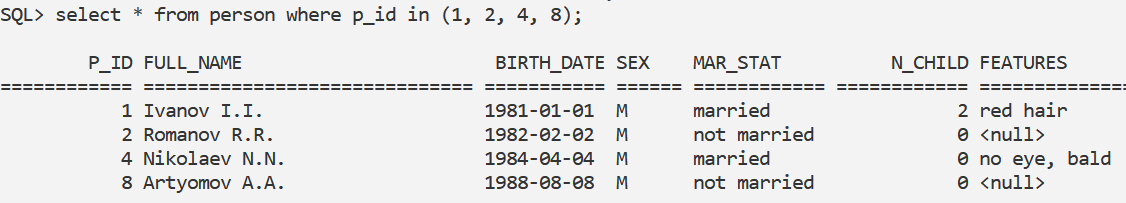


Рисунок 3.16 – Запрос с функцией IN

Наконец, была продемонстрирована работа условия NOT в условии запроса. Для этого был составлен запрос, исключающий из выборки женщин и бездетных людей, т.е. выводящий всех мужчин с детьми (Рисунок 3.17).

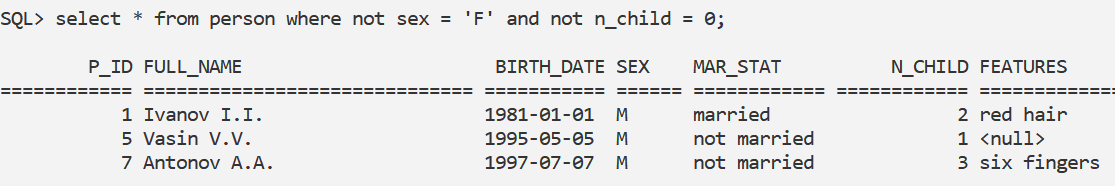


Рисунок 3.17 – Запрос с условием NOT

# КОНТРОЛЬНЫЕ ВОПРОСЫ

## Отличие архитектуры SuperServer и Classic

В архитектуре Classic на каждое клиентское соединение на компьютере-сервере запускается серверный процесс, который обслуживает одного клиента. Количество процессов зависит от числа клиентов, установивших соединение.

В архитектуре SuperServer все клиентские соединения обслуживаются одним серверным процессом, где каждым конкретным клиентом занимаются отдельные потоки (threads).

Архитектура SuperServer считается более производительной (обработка потоков происходит проще), а Classic более надёжной.

## Какая информация указывается при создании базы данных?

Создание новой базы данных производится с помощью оператора CREATE DATABASE. При этом указывается имя файла базы данных (с путём хранения на диске), имя пользователя-создателя, пароль, размер страницы базы данных (например, 4096) и набор символов по умолчанию.

## Базовое понятие многопользовательских систем — «транзакция»

Транзакция – это последовательность действий с базой данных, в которой либо все действия выполняются успешно, либо не выполняется ни одно из них. Транзакция является атомарной, т.е. выполняется как единое целое.

На языке SQL транзакция представляют собой группу операций, помещённых между операторов BEGIN и COMMIT. Также транзакцию можно откатить (оператор ROLLBACK), вернув базу данных в исходное состояние.

## Модели данных. Реляционная модель данных

Модель данных − это совокупность структур данных и операций их обработки. Основные типы моделей: иерархическая, сетевая и реляционная.

В реляционной модели данных объекты и связи между ними представляются в виде таблиц. Каждая реляционная таблица представляет собой двумерный массив. Все строки, составляющие таблицу в реляционной базе данных, должны иметь первичный ключ, быть однородными и не иметь дубликатов. Каждый столбец должен иметь уникальное имя. Все современные средства СУБД поддерживают реляционную модель данных.

## Назначение СУБД

СУБД — это набор программ, позволяющий организовывать, контролировать и администрировать базы данных. Доступ к данным базы любые программы получают через СУБД.

СУБД создает сложные структуры, необходимые для хранения данных, освобождая пользователя от определения и программирования их физических свойств. Также СУБД выполняет структурирование вводимых данных, преобразуя логический формат данных в физический, форму, удобную для хранения.

## Основные понятия реляционных БД: отношение, атрибут, кортеж, домен, схема отношения

Отношение - множество кортежей, соответствующих одной схеме отношения.

Атрибут – это информационное отображение свойства объекта, ненулевая ячейка в электронной таблице или соединение столбца и строки.

Кортеж – элемент отношения, строка таблицы; упорядоченный набор из *N* элементов.

Домен – тип данных с дополнительными условиями (ограничивающими допустимый набор значений).

Схема отношения - именованное множество пар (имя атрибута – имя домена или типа).

## Правила задания таблиц

Таблицы создаются оператором CREATE TABLE. Необходимо указать имя таблицы и определить все столбцы. В определение столбца входит его имя, тип данных (или домен) и, при желании, дополнительные ограничения.

## Ключевой атрибут, первичный ключ

Ключевой атрибут — это атрибут, который определяет столбцы в таблице, используемые в связях внешнего ключа с таблицей фактов.

Первичный ключ — особенное поле в SQL-таблице, которое позволяет однозначно идентифицировать каждую запись в ней

## Требования к вводу данных оператором INSERT INTO

При вводе данных в таблицу с помощью оператора INSERT INTO необходимо указать имя таблицы, в которую будет добавлен кортеж, и предложение VALUES. В нём должны содержаться список столбцов в любом их порядке (если не задавать его, то будет выбран порядок по умолчанию) и значения для всех столбцов строки (в заданном порядке).

Если часть значений будет упущена, то не достающие столбцы будут по возможности заполнены как NULL. Также вводимые значения должны соответствовать типам и ограничениям столбцов таблицы.

# ВЫВОД

В ходе работе были изучены основы организации и настройки сервера Firebird, приобретены навыки работы с реляционными базами данных, изучены базовые понятия языка запросов SQL. В результате была создана база данных, содержащая набор несвязанных между собой таблиц. Их столбцам были даны различные типы и ограничения. В базе данных был осуществлён ряд запросов с простыми и составными условиями. На ряде примеров было продемонстрировано действие таких функций и операторов языка SQL, как WHERE, LIKE, IN, BETWEEN, IS NULL и др.