

Технологии программирования

ЛАБОРАТОРНАЯ РАБОТА № 4

Содержание

ЛАБОРАТОРНАЯ РАБОТА № 4.....	1
Требования к отчёту.....	8
Критерии оценивания.....	9
Задание 1. Установка sqlite в macOS.....	10
Способ 1. Установить sqlite из исходников.....	10
Способ 2. Установить с помощью утилиты Homebrew.....	11
Способ 3. Установить из порта.....	11
Задание 2. Управление базой данных из консоли.....	11
Упражнение 2.1. Изучить видео.....	11
Упражнение 2.2. Изучить примеры.....	11
Упражнение 2.3. Создать базу данных и выполнить запросы.....	11
Упражнение 2.4. Выполнить запросы по вариантам.....	12
Задание 3. Управление базой данных в SQLite Database Manager.....	12
Упражнение 3.1. Изучить примеры.....	12
Упражнение 3.2. Выполнить учебные запросы по вариантам.....	12
Упражнение 3.3. Создать БД и выполнить запросы по вариантам.....	12
Задание 4. Изучение примеров приложений на C подключения и запросов к базе данных.....	12
Дополнительные инструкции.....	12
Задание 5. Создание приложения на C для подключения к базе данных и запросы к базе данных.....	13
Варианты.....	13
Вариант 1.....	13
Задание 2.....	13
Упражнение 2.3.....	13
Упражнение 2.4.....	13
Задание 3.....	14
Упражнение 3.2.....	14

Вариант 2.....	14
Задание 2.....	14
Упражнение 2.3.....	14
Упражнение 2.4.....	14
Задание 3.....	14
Упражнение 3.2.....	14
Вариант 3.....	15
Задание 1.....	15
Упражнение 2.3.....	15
Упражнение 2.4.....	15
Задание 3.....	15
Упражнение 3.2.....	15
Вариант 4.....	15
Задание 1.....	15
Упражнение 2.3.....	15
Упражнение 2.4.....	15
Задание 3.....	16
Упражнение 3.2.....	16
Вариант 5.....	16
Задание 2.....	16
Упражнение 2.3.....	16
Упражнение 2.4.....	16
Задание 3.....	16
Упражнение 3.2.....	16
Вариант 6.....	17
Задание 2.....	17
Упражнение 2.3.....	17
Упражнение 2.4.....	17
Задание 3.....	17
Упражнение 3.2.....	17
Вариант 7.....	17
Упражнение 2.3.....	17
Упражнение 2.4.....	17
Задание 3.....	18

Упражнение 3.2.....	18
Вариант 8.....	18
Задание 2.....	18
Упражнение 2.3.....	18
Упражнение 2.4.....	18
Задание 3.....	18
Упражнение 3.2.....	18
Вариант 9.....	19
Задание 2.....	19
Упражнение 2.3.....	19
Упражнение 2.4.....	19
Задание 3.....	19
Упражнение 3.2.....	19
Вариант 10.....	19
Задание 2.....	19
Упражнение 2.3.....	19
Упражнение 2.4.....	19
Задание 3.....	20
Упражнение 3.2.....	20
Вариант 11.....	20
Задание 2.....	20
Упражнение 2.3.....	20
Упражнение 2.4.....	20
Задание 3.....	20
Упражнение 3.2.....	20
Вариант 12.....	21
Задание 2.....	21
Упражнение 2.3.....	21
Упражнение 2.4.....	21
Задание 3.....	21
Упражнение 3.2.....	21
Вариант 13.....	21
Задание 2.....	21
Упражнение 2.3.....	21

Упражнение 2.4.....	21
Задание 3.....	22
Упражнение 3.2.....	22
Вариант 14.....	22
Задание 2.....	22
Упражнение 2.3.....	22
Упражнение 2.4.....	22
Задание 3.....	22
Упражнение 3.2.....	22
Вариант 15.....	23
Задание 2.....	23
Упражнение 2.3.....	23
Упражнение 2.4.....	23
Задание 3.....	23
Упражнение 3.2.....	23
Вариант 16.....	23
Задание 2.....	23
Упражнение 2.3.....	23
Упражнение 2.4.....	23
Задание 3.....	24
Упражнение 3.2.....	24
Вариант 17.....	24
Задание 2.....	24
Упражнение 2.3.....	24
Упражнение 2.4.....	24
Задание 3.....	24
Упражнение 3.2.....	24
Вариант 18.....	25
Задание 2.....	25
Упражнение 2.3.....	25
Упражнение 2.4.....	25
Задание 3.....	25
Упражнение 3.2.....	25
Вариант 19.....	25

Задание 2.....	25
Упражнение 2.3.....	25
Упражнение 2.4.....	25
Задание 3.....	26
Упражнение 3.2.....	26
Вариант 20.....	27
Задание 2.....	27
Упражнение 2.3.....	27
Упражнение 2.4.....	27
Задание 3.....	27
Упражнение 3.2.....	27
Вариант 21.....	27
Задание 2.....	27
Упражнение 2.3.....	27
Упражнение 2.4.....	27
Задание 3.....	28
Упражнение 3.2.....	28
Вариант 22.....	28
Задание 2.....	28
Упражнение 2.3.....	28
Упражнение 2.4.....	28
Задание 3.....	29
Упражнение 3.2.....	29
Вариант 23.....	29
Задание 2.....	29
Упражнение 2.3.....	29
Упражнение 2.4.....	29
Задание 3.....	29
Упражнение 3.2.....	29
Вариант 24.....	30
Задание 2.....	30
Упражнение 2.3.....	30
Упражнение 2.4.....	30
Задание 3.....	30

Упражнение 3.2.....	30
Вариант 25.....	30
Задание 2.....	30
Упражнение 2.3.....	30
Упражнение 2.4.....	31
Задание 3.....	31
Упражнение 3.2.....	31
Вариант 26.....	31
Задание 2.....	31
Упражнение 2.3.....	31
Упражнение 2.4.....	31
Задание 3.....	32
Упражнение 3.2.....	32
Вариант 27.....	32
Задание 2.....	32
Упражнение 2.3.....	32
Упражнение 2.4.....	32
Задание 3.....	33
Упражнение 3.2.....	33
Вариант 28.....	33
Задание 2.....	33
Упражнение 2.3.....	33
Упражнение 2.4.....	33
Задание 3.....	33
Упражнение 3.2.....	33
Вариант 29.....	34
Задание 2.....	34
Упражнение 2.3.....	34
Упражнение 2.4.....	34
Задание 3.....	34
Упражнение 3.2.....	34
Вариант 30.....	34
Задание 2.....	34
Упражнение 2.3.....	34

Упражнение 2.4.....	34
Задание 3.....	35
Упражнение 3.2.....	35
Вариант 31.....	35
Задание 2.....	35
Упражнение 2.3.....	35
Упражнение 2.4.....	35
Задание 3.....	36
Упражнение 3.2.....	36
Вариант 32.....	36
Задание 2.....	36
Упражнение 2.3.....	36
Упражнение 2.4.....	36
Задание 3.....	36
Упражнение 3.2.....	36
Вариант 33.....	37
Задание 2.....	37
Упражнение 2.3.....	37
Упражнение 2.4.....	37
Задание 3.....	37
Упражнение 3.2.....	37
Вариант 34.....	37
Задание 2.....	37
Упражнение 2.3.....	37
Упражнение 2.4.....	37
Задание 3.....	38
Упражнение 3.2.....	38
Вариант 35.....	38
Задание 2.....	38
Упражнение 2.3.....	38
Упражнение 2.4.....	38
Задание 3.....	38
Упражнение 3.2.....	38
Вариант 36.....	39

Задание 2.....	39
Упражнение 2.3.....	39
Упражнение 2.4.....	39
Задание 3.....	39
Упражнение 3.2.....	39
Вариант 37.....	39
Задание 2.....	39
Упражнение 2.3.....	39
Упражнение 2.4.....	39
Задание 3.....	40
Упражнение 3.2.....	40
Вариант 38.....	40
Задание 2.....	40
Упражнение 2.3.....	40
Упражнение 2.4.....	40
Задание 3.....	40
Упражнение 3.2.....	40
Вариант 39.....	41
Задание 2.....	41
Упражнение 2.3.....	41
Упражнение 2.4.....	41
Задание 3.....	41
Упражнение 3.2.....	41
Вариант 40.....	41
Задание 2.....	41
Упражнение 2.3.....	41
Упражнение 2.4.....	41
Задание 3.....	42
Упражнение 3.2.....	42
Контрольные вопросы.....	42

Требования к отчёту

В файле Readme проекта на github должна быть ссылка на отчёт. Отчет опубликовать в репозитории в папке docs или во внешнем хранилище, добавив ссылку на него в файл Readme репозитория.

Отчет по лабораторной работе содержит тексты задач, примеры запросов к базе данных для заданий 2-3 или скриншоты, протокол тестирования для задачи 5.

К защите необходимо представить исходные тексты программ, оформленных в соответствии с требованиями для оформления кода и согласно модели КИС, и протоколы тестирования каждой программы, подтверждающие правильность ее работы.

Протокол тестирования включает в себя тест (описание входных данных и соответствующих им выходных данных), описание выходных данных, полученных при запуске программы на данном тесте, и отметку о прохождении теста. Тест считается пройденным, если действительные результаты работы программы совпали с ожидаемыми.

Пример оформления протокола тестирования программы на определение количества вхождений слов в строке представлен в таблице 1.

Программы, не прошедшие тестирование, к защите не принимаются. В случае неверной работы программы хотя бы на одном тесте студент обязан выполнить отладку программы для поиска и устранения ошибки.

Таблица 1. Пример протокола тестирования задачи на определение количества вхождений слова в строке.

№ п/п	Входные данные	Ожидаемые выходные данные	Действительные выходные данные	Тест пройден
1	foo bar foo bar	bar: 2 foo: 2	bar: 2 foo: 2	Да
2	test record created	test: 1 record: 1 created: 1	test: 1 record: 1 created: 1	Да
3	1 2 3 4 5 1 2 3 4 5 6 7 8 9	1: 2 2: 2 3: 2 4: 2 5: 2	1: 2 2: 2 3: 2 4: 2 5: 2	Да
4	Hello World	Hello: 1 World: 1	Hello: 1 World: 1	Да

Критерии оценивания

Оценка 4

Выполнены **Задание 1-2, 4**. Отчёт содержит описание задачи, результаты выполнения запросов для задания 2, протоколы тестирования для задания 4. Отчёт должен быть опубликован в репозитории или должна быть ссылка на него в файле

Readme. Отчёт и исходный код проекта может содержать ошибки. Максимальная задержка на предоставление отчёта, исходного кода проекта, включая публикацию в репозитории и задании курса, а также его защиту — 2 недели.

Оценка 5-6

Выполнены **Задание 1-4**. Отчёт содержит описание задачи, результаты выполнения запросов для задания 2-3 и скриншоты, протоколы тестирования для задания 4. Отчёт должен быть опубликован в репозитории или должна быть ссылка на него в файле README. Отчёт и исходный код проекта может содержать ошибки. Максимальная задержка на предоставление отчёта, исходного кода проекта, включая публикацию в репозитории и задании курса, а также его защиту — 1 неделя. Ответы на контрольные вопросы 1-10.

Оценка 7-8

Выполнено **Задание 1-5**. Отчёт содержит описание задачи, результаты выполнения запросов для задания 2-3 и скриншоты, протоколы тестирования для задания 5. Отчёт должен быть опубликован в репозитории или должна быть ссылка на него в файле README. Отчёт и исходный код могут содержать незначительные ошибки. Максимальная задержка на предоставление отчёта, исходного кода проекта, включая публикацию в репозитории и задании курса, а также его защиту — 1 неделя. Ответы на контрольные вопросы 1-16.

Оценка 9

Выполнено **Задание 1-5**. Отчёт содержит описание задачи, результаты выполнения запросов для задания 2-3 и скриншоты, протоколы тестирования для задания 5. Отчёт должен быть опубликован в репозитории или должна быть ссылка на него в файле README. Лабораторная работа опубликована в репозитории и в задании курса в срок, а так же защищена в срок. Не содержит ошибок. Ответы на контрольные вопросы 1-16.

Задание 1. Установка sqlite в macOS

Последняя версия macOS идет с предустановленной SQLite, но если база данных недоступна, то установить SQLite можно выполнив следующие шаги.

Проверить, что SQLite установлена можно с помощью команды:

```
$ whereis sqlite3
```

```
/usr/bin/sqlite3
```

или с помощью команды:

```
$ sqlite3
```

Если sqlite3 установлен, то выполнять установку **НЕ ТРЕБУЕТСЯ**.

Способ 1. Установить sqlite из исходников

1. Скачать SQLite со страницы <https://www.sqlite.org/download.html>

2. Распаковать полученный архив tar.gz

3. Перейти в папку с исходниками sqlite и выполнить команды:

```
$ ./configure --prefix=/usr/local
```

```
$ make
```

```
$ make install
```

Способ 2. Установить с помощью утилиты Homebrew

```
$ brew install sqlite
```

Способ 3. Установить из порта

```
$ sudo port install sqlite
```

Задание 2. Управление базой данных из консоли

Упражнение 2.1. Изучить видео

Ознакомиться с видео https://youtu.be/QjICgmk31js?list=PLGLfVvz_LVvTsslWD1HBQeJBbmAaAF9Xy.

Упражнение 2.2. Изучить примеры

Изучить примеры работы с базой sqlite из документа [Базы данных.pdf](#) и из книги [Grant Allen, Mike Owens. The Definitive Guide to SQLite \(Second Edition\) — 2010.pdf](#) (Chapter 3: SQL for SQLite, стр. 47-87).

Упражнение 2.3. Создать базу данных и выполнить запросы

Создать базу данных согласно варианту, продемонстрировать следующие навыки работы с консолью sqlite:

- создание таблицы (create);
- вставка данных в таблицу (insert);
- выборка данных (select) с выводом всех данных по столбцам и строкам, с сортировкой по id и по имени и с выводом последних 5 строк (инструкция limit);
- выборка данных с фильтрацией (условие where), если id=5;
- выборка данных с фильтрацией (условие where) и с совпадением по маске, например все записи, где имя объекта (согласно варианту) начинается на первую букву вашей фамилии (инструкция like);
- переименование таблицы (alter);

- обновление данных с использованием update;
- удаление строк по id и по названию объекта;
- экспорт базы данных в файлы .sql, .csv.;
- удаление таблицы;

Упражнение 2.4. Выполнить запросы по вариантам

Выполнить дополнительные задания согласно варианту.

Задание 3. Управление базой данных в SQLite Database Manager

Упражнение 3.1. Изучить примеры

Изучить примеры работы с базой sqlite из документа [Печеночкин Г. SQL для непрограммистов](#).

Упражнение 3.2. Выполнить учебные запросы по вариантам

Скачать базу данных [Учет расходов](#).

Выполнить задания в SQLite Database Manager, например **DB Browser for SQLite** (<http://sqlitebrowser.org/>) или **Valentina Studio** (скачать из AppStore <https://itunes.apple.com/us/app/valentina-studio/id604825918?mt=12>) согласно варианту.

Упражнение 3.3. Создать БД и выполнить запросы по вариантам

Создать БД согласно варианту для задания 2 (упражнение 2.3).

Продемонстрировать выполнение всех операций и запросов согласно упражнениям 2.3 и 2.4.

Задание 4. Изучение примеров приложений на C подключения и запросов к базе данных

Познакомиться с руководством http://www.tutorialspoint.com/sqlite/sqlite_c_cpp.htm, разобрать и реализовать примеры из руководства с компиляцией в консоли.

Изучить примеры выполнения параметризованных запросов, вставку изображений в базу данных, вставку данных в режиме **autocommit** и в виде **транзакций**, вывод метаданных базы данных на примере материала из <http://zetcode.com/db/sqlitec/>.

Примеры опубликовать в репозитории для лабораторной работы 4 edufmi.bsu.by в ветке examples. Под примеры создать папку examples и именовать файлы согласно шаблону example1_grN_LastnameFirstname.c, где **N** — номер группы, **LastnameFirstname** — Ваши фамилия и имя. В каталоге examples создать файл EXAMPLES.md, в котором перечислить список рассмотренных примеров. Использовать синтаксис markdown для формирования файла EXAMPLES.md.

В репозитории должен быть создан файл README.md согласно примеру

README файла из лабораторной работы 3, в котором указать список веток репозитория и какой код в каждой ветке размещен.

Дополнительные инструкции

1. Как создать приложение и подключиться к базе данных - <https://www.sqlite.org/quickstart.html>
2. How To Compile SQLite - <https://www.sqlite.org/howtocompile.html>
3. An Introduction To The SQLite C/C++ Interface - <http://www.sqlite.org/cintro.html>
4. SQLite C tutorial - <http://zetcode.com/db/sqlitec/>
5. Using SQLite in C programs - <http://www.wassen.net/sqlite-c.html>
6. How to Use SQLite to Manage Data in iOS Apps - <http://www.appcoda.com/sqlite-database-ios-app-tutorial/>

Задание 5. Создание приложения на C для подключения к базе данных и запросы к базе данных

В репозитории для лабораторной работы кроме веток `main` и `examples` создать ветку `dev`. Структура и код проекта реализовать в папке `bdproject5`. В файле `README.md` предоставить полное описание проекта. Реализовать весь функционал в ветке `dev` и опубликовать используя `pull request` в ветку `main`.

Используя **DB Browser for SQLite** или **Valentina Studio**, создайте и заполните БД согласно варианту для задания 2. База данных должна состоять из нескольких таблиц, одна из таблиц должна содержать поле `фотография`, в которое при выполнении приложений добавлять фотографию объекта согласно варианту.

Используя консольные редакторы `nano/vi` или `Xcode` ([Обзор возможностей Xcode](#)), напишите консольное приложение на языке C, которое устанавливает соединение с БД SQLite и выполняет запросы типа **SELECT**, **INSERT**, **DELETE**, **параметризованные запросы**, фотографию объекта выводит в файл. Структура проекта приложения должна соответствовать модели КИС и содержать `Makefile` для сборки с помощью утилиты `make`.

Иллюстрирует вставку данных в режиме **autocommit** (исполнение инструкций `sqlite` построчно) и **транзакцией**.

Требования к приложению:

- Структура проекта приложения размещается в каталоге `bdproject5`, должна соответствовать модели КИС и содержать `Makefile` для сборки с помощью утилиты `make`.
- наличие меню для выбора операций с приложением;
- возможность ввода параметра для выборки;
 - по `id`,
 - по шаблону для разных полей, например фрагменту названия продукта или

- фрагменту фамилии,
- полю, общему для нескольких строк, например в зависимости от варианта это может быть название категории товара, или специальности, или города и т.д.
- запросы должны быть описаны в коде приложения (без выполнения в консоли);
- содержит функционал для демонстрации вставки данных в режиме **autocommit** (исполнение инструкций sqlite построчно) и **транзакцией**.

Варианты

Вариант 1

Задание 2

Упражнение 2.3

«Человек»:

id, фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира).

Упражнение 2.4

Выполнить запросы:

- Вывести сведения о самом молодом человеке.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `position_id` (тип `integer` и содержит идентификаторы должностей).
- Создать таблицу `category` (`id`, `position_name`, `position_description`).
- Вывести данные обо всех людях в форме идентификатор человека, фамилия, имя, дата рождения, название должности.
- Подсчет количества человек с помощью `count`, если `рост > 164` см.
- Суммарный вес всех человек с помощью `sum`, если `вес < 76` кг.
- максимальный и минимальный рост с помощью `max` и `min`.
- Используя инструкцию `inner join` вывести полные сведения о людях и должностях для должности с `id=4`.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий все поля всех записей таблицы `Categories`.
2. Составьте запрос, возвращающий название отслеживаемого товара, название магазина, в котором он был куплен и потраченную на этот товар сумму для тех покупок, в которых эта сумма превышает 200 руб, отсортированные в следующем порядке: название товара по алфавиту, название магазина по алфавиту, потраченная сумма в порядке уменьшения.

Вариант 2

Задание 2

Упражнение 2.3

«Автомобиль»: id; марка; цвет; серийный номер; регистрационный номер; год выпуска; год техосмотра; цена.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про автомобили, которым больше 2 лет.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых vendor_id (тип integer и содержит идентификаторы заводов).
- Создать таблицу vendor (id, ven_name, ven_description).
- Вывести данные обо всех автомобилях в форме идентификатор автомобиля, наименование, год выпуска, название завода.
- подсчет количества машин с помощью count, если стоимость > 16000 руб
- суммарная стоимость машин с помощью sum, если год выпуска = 2016
- максимальная и минимальная стоимость с помощью max и min
- Используя инструкцию inner join вывести полные сведения о машинах и заводе для завода с id=3.

Задание 3.

Упражнение 3.2

1. Составьте запрос, возвращающий перечень названий всех товаров, по которым ведётся детализированный учёт расходов.
2. Составьте запрос, возвращающий названия отслеживаемых товаров и все потраченные на них суммы в следующем порядке: названия отсортированы по алфавиту, а суммы по каждому товару отсортированы в порядке уменьшения.

Вариант 3

Задание 1

Упражнение 2.3

«Государство»: id; название страны; столица; государственный язык; население; площадь территории; денежная единица; государственный строй; глава государства.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про государства, население которых больше 20 млн жителей.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых area_id (тип integer и содержит идентификаторы частей света).
- Создать таблицу area (id, area_name, area_description).
- Вывести данные обо всех государствах в форме идентификатор государства, столица, площадь территории, название части света.
- подсчет количества стран с помощью count, если население > 10 млн

- суммарное количество населения с помощью sum, если идентификатор части света=2
- максимальное и минимальное количество населения с помощью max и min
- Используя инструкцию inner join вывести полные сведения о государствах и части света для части света с id=1.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий названия и описание всех категорий расходов.
2. Составьте запрос к таблице Spendings, возвращающий названия магазинов, даты совершенных покупок и суммы покупок, отсортированные в порядке убывания сумм.

Вариант 4

Задание 1

Упражнение 2.3

«Абитуриент»: id; фамилия; имя; отчество; пол; национальность; дата рождения (год, месяц, число); домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); оценки по ЦТ; проходной балл.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про абитуриентов, проходной балл которых равен больше 225.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых vuz_id (тип integer и содержит идентификаторы вузов).
- Создать таблицу vuz (id, vuz_name, vuz_description).
- Вывести данные обо всех абитуриентах в форме идентификатор абитуриента, фамилия, имя, отчество, дата рождения, название вуза, куда поступает.
- подсчет количества абитуриентов с помощью count, если проходной балл >250
- суммарный балл абитуриентов с помощью sum, если пол=муж
- максимальный и минимальный балл с помощью max и min
- Используя инструкцию inner join вывести полные сведения об абитуриентах и вузе для вуза с id=3.

Задание 3

Упражнение 3.2

1. Составьте запрос, выбирающий из таблиц Goods и Goods_Spending и возвращающий перечень названий отслеживаемых товаров и всех потраченных на них сумм.
2. Составьте запрос, возвращающий названия всех категорий покупок, отсортированных по алфавиту.

Вариант 5

Задание 2

Упражнение 2.3

«Рабочий»: id; фамилия; имя; отчество; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); национальность; дата рождения (год, месяц число); табельный номер; образование; год поступления на работу; заработная плата.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про рабочих, поступивших на работу в 2010 году.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых department_id (тип integer и содержит идентификаторы цеха).
- Создать таблицу department (id, dep_name, dep_description).
- Вывести данные обо всех рабочих в форме идентификатор рабочего, фамилия, имя, табельный номер, № цеха, название цеха.
- подсчет количества рабочих с помощью count, если год поступления на работу >2014
- суммарная заработная плата с помощью sum, если год поступления на работу >2016
- максимальная и минимальная заработная плата с помощью max и min
- Используя инструкцию inner join вывести полные сведения о рабочих и цехе для цеха с id=5.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий описание категории с номером 1.
2. Составьте запрос, обращающийся к таблицам Goods и Goods_Spendings и возвращающий названия отслеживаемых товаров и потраченные на них суммы, превышающие 500 руб.

Вариант 6

Задание 2

Упражнение 2.3

«Военнослужащий»: id; фамилия; имя; отчество; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); национальность; дата рождения (год, месяц число); должность; звание; зарплата.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про военнослужащих в звании “лейтенант”.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых department_id (тип integer и содержит идентификаторы частей).
- Создать таблицу department (id, dep_name, dep_description).
- Вывести данные обо всех военнослужащих в форме идентификатор военнослужащего, фамилия, имя, дата рождения, должность, название части.
- подсчет количества военнослужащих с помощью count, если звание=сержант

- суммарная заработная плата с помощью `sum`, если звание<>лейтенант
- максимальная и минимальная заработная плата с помощью `max` и `min`
- Используя инструкцию `inner join` вывести полные сведения о военнослужащих и части для части с `id=2`.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий название и описание категории с номером 2.
2. Составьте запрос к таблице `Spending`s, возвращающий название магазина, дату покупки и сумму для покупок, сумма которых находится в диапазоне от 1000 до 2000 руб (с использованием ключевого слова `between`).

Вариант 7

Упражнение 2.3

«Владелец телефона»: `id`; фамилия; имя; отчество; дата рождения; город, домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); № телефона; год заключения договора; название тарифа; стоимость тарифа.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про владельцев телефона номер, которого начинается на 621.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `operator_id` (тип `integer` и содержит идентификаторы мобильных операторов).
- Создать таблицу `operator` (`id`, `oper_name`, `oper_description`).
- Вывести данные о владельцах телефонов в форме идентификатор владельца, фамилия, имя, дата рождения, № телефона; название оператора.
- подсчет количества владельцев телефонов с помощью `count`, если год заключения договора>2017
- суммарная стоимость тарифов с помощью `sum`, если год заключения договора <2015
- максимальная и минимальная стоимость тарифа с помощью `max` и `min`
- Используя инструкцию `inner join` вывести полные сведения о владельцах телефонов и операторе для оператора с `id=1`.

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам `Categories` и `Spending`s, который возвращает название категории покупки, название магазина и потраченную сумму для всех покупок, относящихся к категории с номером 4.
2. Составьте запрос, возвращающий названия категорий с номерами 1, 3 и 4 (с использованием ключевого слова `IN`)

Вариант 8

Задание 2

Упражнение 2.3

«Владелец автомобиля»: id; фамилия; имя; отчество; номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира) марка автомобиля; номер автомобиля; номер техпаспорта; стоимость; год выпуска.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про автомобили марки «Audi».
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых vendor_id (тип integer и содержит идентификаторы производителей автомобилей).
- Создать таблицу vendor (id, ven_name, ven_description).
- Вывести данные о владельцах автомобилей в форме идентификатор владельца, фамилия, имя, № телефона; марка автомобиля, номер автомобиля, название производителя.
- подсчет количества владельцев автомобилей с помощью count, если год выпуска автомобиля=2017
- суммарная стоимость автомобилей с помощью sum, если год выпуска >2015
- максимальная и минимальная стоимость автомобиля с помощью max и min
- Используя инструкцию inner join вывести полные сведения о владельцах автомобилей и производителях для производителей с id = 1 и 2.

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам Goods и Goods_Spendings, возвращающий наименование товара и количество купленных единиц по всем покупкам для товара «Пиво».

2. Составьте запрос к таблице Spendings, возвращающий название магазина и потраченную сумму для покупок, совершённых не позже 3 июня 2015 года.

Подсказка. Эту дату в запросе нужно указать в виде такой текстовой строки: '2015-06-03'

Вариант 9

Задание 2

Упражнение 2.3

«Пациент»: id; фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); номер медицинской карты; диагноз; группа крови; отделение; .

Упражнение 2.4

Выполнить запросы:

- Вывести данные про пациентов с 18 отделения.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых hospital_id (тип integer и содержит идентификаторы больниц).

- Создать таблицу hospital (id, hos_name, hos_description).
- Вывести данные о пациентах в форме идентификатор пациента, фамилия, имя, дата рождения; диагноз, номер отделения, название больницы.
- подсчет количества пациентов с помощью count, если отделение =17
- суммарный вес пациентов с помощью sum, если рост >164 см
- максимальный и минимальный рост пациента с помощью max и min
- Используя инструкцию inner join вывести полные сведения о пациентах и больнице для больниц с id = 3 и 2.

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам Goods, Goods_Spendings, Spendings, возвращающий наименование товара и количество купленных единиц по всем покупкам для товара «Пиво», дату покупки. Таким образом, ваш запрос должен обращаться к трём таблицам и возвращать наименование товара, количество купленных единиц и дату каждой покупки для товара «Пиво».

2. Составьте запрос, возвращающий значения всех полей таблицы Spendings для покупок на сумму более 1000 руб.

Вариант 10

Задание 2

Упражнение 2.3

«Покупатель»: id; фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); номер кредитной карточки; номер банковского счета.

Упражнение 2.4

Выполнить запросы:

- Вывести данные о покупателях города Брест.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых shop_id (тип integer и содержит идентификаторы магазинов).
- Создать таблицу shop (id, shop_name, shop_description).
- Вывести данные обо всех покупателях в форме идентификатор покупателя, фамилия, имя, номер телефона, номер кредитной карточки, название магазина.
- подсчет количества покупателей с помощью count, если рост=165 см
- суммарный вес покупателей с помощью sum, если пол=жен
- максимальный и минимальный вес с помощью max и min
- Используя инструкцию inner join вывести сведения о покупателе и магазинах для магазинов с id= 2 и 4.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий описание категории с номером 2.
2. Составьте запрос, обращающийся ко всем четырём таблицам (Categories,

Spending, Goods, Goods_Spending), возвращающий следующую информацию обо всех покупках отслеживаемых товаров: название категории, название товара, название магазина и дату покупки.

Вариант 11

Задание 2

Упражнение 2.3

«Студент»: id; фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); курс; группа; средний бал; специальность.

Упражнение 2.4

Выполнить запросы:

- Вывести сведения про всех студентов у которых средний балл ниже 7.0 баллов.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых vuz_id (тип integer и содержит идентификаторы вузов).
- Создать таблицу vuz (id, vuz_name, vuz_description).
- Вывести данные обо всех студентах в форме идентификатор студента, фамилия, имя, отчество, дата рождения, курс, название вуза.
- подсчет количества студентов с помощью count, если средний балл ≥ 8
- суммарный средни балл студентов с помощью sum, если пол=муж
- максимальный и минимальный балл с помощью max и min
- Используя инструкцию inner join вывести полные сведения о студента и вузах для вузов с id= 1 и3.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий название и описание категории с номером 1.
2. Составьте запрос, возвращающий название отслеживаемого товара, название магазина, в котором он был куплен и потраченную на этот товар сумму для тех покупок, в которых эта сумма превышает 150 руб, отсортированные в следующем порядке: название товара по алфавиту, название магазина по алфавиту, потраченная сумма в порядке увеличения.

Вариант 12

Задание 2

Упражнение 2.3

«Школьник»: id; фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); класс.

Упражнение 2.4

Выполнить запросы:

- Вывести сведения про всех учеников пятых классов.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `school_id` (тип `integer` и содержит идентификаторы школ).
- Создать таблицу `school` (`id`, `sch_name`, `sch_description`).
- Вывести данные обо всех школьниках в форме идентификатор школьника, фамилия, имя, отчество, дата рождения, класс, название школы.
- подсчет количества школьников с помощью `count`, если рост >128 см
- суммарный вес школьников с помощью `sum`, если рост = 120 см
- максимальный и минимальный вес с помощью `max` и `min`
- Используя инструкцию `inner join` вывести полные сведения об школьника и школах для школ с `id= 1` и `5`.

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам `Goods` и `Goods_Spendings`, возвращающий наименование товара и количество купленных единиц по всем покупкам для товара «Траты в WoT».
2. Составьте запрос, возвращающий названия отслеживаемых товаров и все потраченные на них суммы в следующем порядке: названия отсортированы по алфавиту, а суммы по каждому товару отсортированы в порядке увеличения.

Вариант 13

Задание 2

Упражнение 2.3

«Человек»: `id`; фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира).

Упражнение 2.4

Выполнить запросы:

- Вывести сведения о самом молодом человеке.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `city_id` (тип `integer` и содержит идентификаторы городов).
- Создать таблицу `category` (`id`, `city_name`, `city_description`).
- Вывести данные обо всех людях в форме идентификатор человека, фамилия, имя, дата рождения, название города
- подсчет количества человек с помощью `count`, если рост >170 см
- суммарный вес людей с помощью `sum`, если рост < 176 см
- максимальный и минимальный вес с помощью `max` и `min`
- Используя инструкцию `inner join` вывести сведения о людях и городе для города с `id=5`.

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам `Categories` и `Spendings`, который возвращает

название категории покупки, название магазина и потраченную сумму для всех покупок, относящихся к категории с номером 5.

2. Составьте запрос, обращающийся к таблицам Goods и Goods_Spendings и возвращающий названия отслеживаемых товаров и потраченные на них суммы, превышающие 400 руб.

Вариант 14

Задание 2

Упражнение 2.3

«Товар»: id; наименование; стоимость; срок хранения; сорт; дата выпуска; срок годности.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про товары срок годности которых истекает в этом году.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых category_id (тип integer и содержит идентификаторы категорий товаров).
- Создать таблицу category (id, cat_name, cat_description).
- Вывести данные обо всех товарах в форме идентификатор товара, наименование, дата выпуска, название категории товара.
- подсчет количества товаров с помощью count, если стоимость=49 руб
- суммарная стоимость товаров с помощью sum, если год выпуска>2016
- максимальная и минимальная стоимость с помощью max и min
- Используя инструкцию inner join вывести полные сведения о товарах и категории для категории с id=2.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий перечень названий всех товаров, по которым ведётся детализированный учёт расходов.

2. Составьте запрос, возвращающий названия отслеживаемых товаров и все потраченные на них суммы в следующем порядке: названия отсортированы по алфавиту, а суммы по каждому товару отсортированы в порядке уменьшения.

Вариант 15

Задание 2

Упражнение 2.3

«Рейс»: id, марка автомобиля; номер автомобиля; пункт назначения; грузоподъемность (в тоннах); стоимость единицы груза; общая стоимость груза.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про автомобили, грузоподъемность которых больше 2 тонн.
- Используя инструкцию alter, добавить дополнительные столбцы, один из

которых category_id (тип integer и содержит идентификаторы категорий транспортных средств для перевозки грузов (см. <https://auto-tahograf.ru/vopros-otvet/o-tahografah/kategorii-transportnykh-sredstv/>)).

- Создать таблицу category (id, cat_name, cat_description).
- Вывести данные обо всех автомобилях в форме идентификатор автомобиля, марка, номер автомобиля, название категории автомобиля.
- Подсчет количества автомобилей с помощью count, если стоимость единицы единицы груза=2 руб.
- Суммарная стоимость товаров с помощью sum, если пункт назначения=«Виталюр».
- Максимальная и минимальная общая стоимость груза с помощью max и min.
- Средняя стоимость всех грузов для магазинов «5 элемент».
- Используя инструкцию inner join вывести полные сведения об автомобилях и категории для категории=N1.

Задание 3

Упражнение 3.2

1. Составьте запрос, выбирающий из таблиц Goods и Goods_Spending и возвращающий перечень названий отслеживаемых товаров и всех потраченных на них сумм.
2. Составьте запрос, возвращающий названия всех категорий покупок, отсортированных по убыванию.

Вариант 16

Задание 2

Упражнение 2.3

«Книга»: id, название; автор (фамилия; имя); год выхода; издательство; себестоимость; цена; прибыль.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про книги авторов, фамилия которых начинается с буквы “К”.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых type_id (тип integer и содержит идентификаторы категорий жанров книг).
- Создать таблицу category (id, cat_name, cat_description).
- Вывести данные обо всех книгах в форме идентификатор книги, название, год выхода, название категории книги.
- подсчет количества товаров с помощью count, если цена=28,32 руб
- суммарная прибыль книг с помощью sum, если год выхода >2017
- максимальная и минимальная стоимость книги с помощью max и min.
- Используя инструкцию inner join вывести полные сведения о книгах и жанрах для жанра= «Биографии и мемуары».

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий описание категории с номером 2.
2. Составьте запрос, обращающийся к таблицам Goods и Goods_Spendings и возвращающий названия отслеживаемых товаров и потраченные на них суммы, превышающие 450 руб.

Вариант 17

Задание 2

Упражнение 2.3

«Здание»: id, адрес; тип здания; количество этажей; количество квартир; срок эксплуатации; срок до капитального ремонта (70 лет - срок эксплуатации).

Упражнение 2.4

Выполнить запросы:

- Вывести сведения о зданиях, срок эксплуатации которых > 50 лет.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых district_id (тип integer и содержит идентификаторы районы расположения зданий).
- Создать таблицу district (id, dist_name, dist_description).
- Вывести данные обо всех зданиях в форме идентификатор здания, адрес, срок эксплуатации, район расположения здания.
- подсчёт количества зданий с помощью count, если срок эксплуатации=14 лет
- суммарная количество квартир с помощью sum, если срок эксплуатации>49 лет.
- максимальный и минимальный срок эксплуатации с помощью max и min.
- Используя инструкцию inner join вывести полные сведения о зданиях и районах расположения для района=«Московский».

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам Categories и Spendings, который возвращает название категории покупки, название магазина и потраченную сумму для всех покупок, относящихся к категории с номером 5.
2. Составьте запрос, возвращающий названия категорий с номерами 1, 3 и 4 (с использованием ключевого слова IN)

Вариант 18

Задание 2

Упражнение 2.3

«Учёный»: id, фамилия; имя; отчество; пол; национальность; дата рождения (год, месяц, число); возраст; учёная степень; должность; номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира).

Упражнение 2.4

Выполнить запросы:

- Вывести сведения об учёных кандидатах технических наук.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `organization_id` (тип `integer` и содержит идентификаторы научных и образовательных организаций).
- Создать таблицу `organization` (`id`, `org_name`, `org_description`).
- Вывести данные обо всех ученых в форме идентификатор ученого, фамилия, имя, степень, должность, название организации.
- Подсчет количества ученых с помощью `count`, если национальность=белорус.
- суммарный возраст ученых с помощью `sum`, если должность<>доцент.
- максимальный и минимальный возраст с помощью `max` и `min`.
- Используя инструкцию `inner join` вывести полные сведения об ученых и организациях для организации= «Белорусский государственный университет».

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам `Goods` и `Goods_Spendings`, возвращающий наименование товара и количество купленных единиц по всем покупкам для товара «Пиво».
2. Составьте запрос к таблице `Spendings`, возвращающий название магазина и потраченную сумму для покупок, совершённых не позже 31 мая 2015 года.

Подсказка. Эту дату в запросе нужно указать в виде такой текстовой строки: '2015-05-31'.

Вариант 19

Задание 2

Упражнение 2.3

«Международная компания»: *id, название; интернет сайт; адрес главного офиса (почтовый индекс, страна, область, район, город, улица, дом, квартира); год основания; продолжительность пребывания на мировом рынке; количество сотрудников; количество филиалов в Европе.*

Упражнение 2.4

Выполнить запросы:

- Вывести международные компании, количество сотрудников у которых больше 10000.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `category_id` (тип `integer` и содержит идентификаторы категорий компаний).
- Создать таблицу `category` (`id`, `cat_name`, `cat_description`), где `category` содержит классификацию компаний по уставным различиям:
 - Международный картель — представляет собой объединение, в котором его участники сохраняют производственную и коммерческую самостоятельность, но договариваются между собой (заключают соглашение) об установлении единых цен на продаваемые товары, о распределении рынков сбыта, о квотах, (доле каждого участника в

- производстве продукции).
- Международный синдикат — союз корпораций различных стран, связанных соглашением о создании единой сбытовой организации. При такой форме союза его участники не имеют права продавать свои товары на внешнем рынке и сдают их специальному акционерному обществу, которое реализует всю продукцию синдиката на мировом рынке.
- Международный трест — объединяет собственность на все вошедшие в его состав предприятия, а владельцы последних получают прибыль как пайщики по сумме вложенных капиталов.
- Международный концерн — объединяют предприятия различных отраслей промышленности, транспорта, торговли, банков, страховых компаний, причем многочисленные предприятия и компании различных стран подчинены одному финансовому центру, сосредоточивающему в своих руках контрольные пакеты их акций.
- Вывести данные обо всех компаниях в форме идентификатор компании, название, год основания, название категории компании.
- подсчет количества компаний с помощью count, если год основания < 2008.
- суммарное количество сотрудников с помощью sum, если год основания=2012
- максимальное и минимальное количество сотрудников с помощью max и min.
- Используя инструкцию inner join вывести полные сведения о компаниях и их категории для категории «Международный трест».

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам Goods, Goods_Spendings, Spendings, возвращающий наименование товара и количество купленных единиц по всем покупкам для товара «Пиво», дату покупки. Таким образом, ваш запрос должен обращаться к трём таблицам и возвращать наименование товара, количество купленных единиц и дату каждой покупки для товара «Пиво».
2. Составьте запрос, возвращающий значения всех полей таблицы Spendings для покупок на сумму более 800 руб.

Вариант 20

Задание 2

Упражнение 2.3

«Пенсионер»: *id, фамилия; имя; отчество; пол; национальность; год рождения; номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); пенсионный стаж; пенсия.*

Упражнение 2.4

Выполнить запросы:

- Вывести сведения про всех пенсионеров, которые на пенсии больше 5 лет.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых job_id (тип integer и содержит идентификаторы должностей).
- Создать таблицу job (id, job_position, job_period), где job_position — название

- должности, job_period — количество лет работы на этой должности.
- Вывести данные обо всех пенсионерах в форме идентификатор пенсионера, фамилия, имя, пенсионный стаж, название должности, с которой ушел на пенсию.
- подсчет количества пенсионеров с помощью count, если пенсионный стаж=2 года.
- суммарная пенсия с помощью sum, если пенсионный стаж>10 лет.
- максимальная и минимальная пенсия с помощью max и min.
- Используя инструкцию inner join вывести полные сведения о пенсионерах и должностях для должности «Учитель с высшей категорией».

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий описание категории с номером 4.
2. Составьте запрос, обращающийся ко всем четырём таблицам (Categories, Spendings, Goods, Goods_Spendings), возвращающий следующую информацию обо всех покупках отслеживаемых товаров: название категории, название товара, название магазина и дату покупки.

Вариант 21

Задание 2

Упражнение 2.3

«Зоопарк»: id, Название животного; количество вида; адрес зоопарка (почтовый индекс, страна, область, район, город, улица, дом, квартира); общее количество животных в зоопарке; количество работников.

Упражнение 2.4

Выполнить запросы:

- Вывести сведения про зоопарки, в которых есть уссурийские тигры.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых category_id (тип integer и содержит идентификаторы видов).
- Создать таблицу category (id, cat_name, cat_description) для описания видов животных.
- Вывести данные обо всех животных в форме идентификатор животного, название, количество животных, название вида животного.
- подсчет количества животных с помощью count, если количество вида=3.
- среднее количества вида животных с помощью avg, если общее количество животных в зоопарке=64.
- максимальное и минимальное количество вида животных с помощью max и min.
- Используя инструкцию inner join вывести полные сведения о животных и их вида для вида=«Млекопитающие».

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий название и описание категории с номером 3.
2. Составьте запрос, возвращающий название отслеживаемого товара, название магазина, в котором он был куплен и потраченную на этот товар сумму для тех покупок, в которых эта сумма превышает 250 руб, отсортированные в следующем порядке: название товара по алфавиту, название магазина по возрастанию, потраченная сумма в порядке увеличения.

Вариант 22

Задание 2

Упражнение 2.3

«Программное обеспечение»: *id, название; название компании производителя; год выхода; версия; период поддержки; цена.*

Упражнение 2.4

Выполнить запросы:

- Вывести данные о программном обеспечении, которое дороже 60 рублей и период поддержки которого более 3 лет.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых category_id (тип integer и содержит идентификаторы категорий программного обеспечения).
- Создать таблицу category (id, cat_name, cat_description) для категорий «Прикладное ПО» и «Системное ПО».
- Вывести данные обо всех ПО в форме идентификатор программного обеспечения, названия, год выхода, название категории ПО.
- подсчет количества ПО с помощью count, если период поддержки <=5 лет.
- суммарная стоимость ПО с помощью sum, если год выхода>2011
- максимальная и минимальная стоимость с помощью max и min.
- Используя инструкцию inner join вывести полные сведения о ПО и категории для категории «Прикладное программное обеспечение».

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам Goods и Goods_Spendings, возвращающий наименование товара и количество купленных единиц по всем покупкам для товара «Траты в WoT».
2. Составьте запрос, возвращающий названия отслеживаемых товаров и все потраченные на них суммы в следующем порядке: названия отсортированы по алфавиту по убыванию, а суммы по каждому товару отсортированы в порядке возрастания.

Вариант 23

Задание 2

Упражнение 2.3

«Область»: *id*, название области; областной центр; население (мужчины и женщины; мужчины; женщины); площадь территории; руководитель областной администрации.

Добавить в таблицу области Беларуси, Украины и России (не менее 5 для каждой страны).

Упражнение 2.4

Выполнить запросы:

- Вывести данные об областях, население которых меньше 1,2 млн. жителей.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `country_id` (тип `integer` и содержит идентификаторы стран).
- Создать таблицу `country` (`id`, `country_name`, `country_capital`) и добавить в нее страны и их столицы.
- Вывести данные обо всех областях в форме идентификатор области, название, количество жителей области, страна.
- подсчет количества жителей с помощью `count`, если площадь территории ≥ 39000 кв. км.
- суммарная количество жителей с помощью `sum`, если площадь территории < 38000 кв. км
- максимальная и минимальная площадь территории области с помощью `max` и `min`
- Используя инструкцию `inner join` вывести полные сведения об областях и стране для Беларуси.

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам `Categories` и `Spending`, который возвращает название категории покупки, название магазина и потраченную сумму для всех покупок, относящихся к категории с номером 3.

2. Составьте запрос, обращающийся к таблицам `Goods` и `Goods_Spendings` и возвращающий названия отслеживаемых товаров и потраченные на них суммы, превышающие 450 руб.

Вариант 24

Задание 2

Упражнение 2.3

«Автомобиль»: *id*, марка; цвет; серийный номер; регистрационный номер; год выпуска; год техосмотра; цена.

Добавить в таблицу сведения как минимум от трех производителей автомобилей и трех моделей с разными годами выпуска и стоимости.

Упражнение 2.4

Выполнить запросы:

- Вывести данные об автомобилях, стоимость которых меньше 18 000 BYN.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых insurance_id (тип integer и содержит идентификаторы страховок).
- Создать таблицу insurance (id, insurance_name, date_start, date_end, cost, money_volume) и добавить в нее страховки, стоимость и период действия.
- Вывести данные обо всех автомобилях в форме идентификатор автомобиля, модель, регистрационный номер, стоимость автомобиля.
- Подсчет количества автомобилей с помощью count, если стоимость ≥ 20000 BYN.
- Суммарную стоимость автомобилей с помощью sum, если цвет автомобиля «красный».
- Максимальная и минимальная стоимость автомобиля с помощью max и min
- Используя инструкцию inner join вывести полные сведения обо всех моделях и страховках для General Motors.

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам Goods и Goods_Spendings, возвращающий наименование товара и количество купленных единиц по всем покупкам для товара «Пиво».

2. Составьте запрос к таблице Spendings, возвращающий название магазина и потраченную сумму для покупок, совершённых не позже 28 мая 2015 года.

Подсказка. Эту дату в запросе нужно указать в виде такой текстовой строки: '2015-05-28'.

Вариант 25

Задание 2

Упражнение 2.3

«Государство»: id, название страны; столица; материк; государственный язык; население; площадь территории; денежная единица; государственный строй; глава государства.

Добавить сведения не менее 5 стран с разным количеством жителей Евразии и Южной Америки.

Упражнение 2.4

Выполнить запросы:

- Вывести данные обо всех странах, население которых меньше 20 млн. жителей.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых world_part_id (тип integer и содержит идентификаторы частей света).
- Создать таблицу world_parts (id, world_part_name_ru, world_part_name_be, world_part_name_en) и добавить в нее части света.
- Вывести данные обо всех странах в форме идентификатор страны, название, количество жителей страны, материк.

- подсчет количества жителей с помощью count, если площадь территории >=200000 кв. км.
- суммарная количество жителей с помощью sum, если площадь территории <700000 кв. км
- максимальная и минимальная площадь территории страны с помощью max и min
- Используя инструкцию inner join вывести полные сведения о странах и части света для Европы.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий описание категории с номером 3.
2. Составьте запрос, обращающийся ко всем четырём таблицам (Categories, Spendings, Goods, Goods_Spendings), возвращающий следующую информацию обо всех покупках отслеживаемых товаров: название категории, название товара, название магазина и дату покупки.

Вариант 26

Задание 2

Упражнение 2.3

«Владелец телефона»: id, фамилия; имя; отчество; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); № телефона, модель телефона, стоимость телефона, дата покупки.

Добавить сведения о разных моделях телефонов и разных производителей (не менее 5 для каждого производителя).

Упражнение 2.4

Выполнить запросы:

- Вывести данные о номерах телефонов, имя владельцев которых «Иван» или «Никита».
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых company_id (тип integer и содержит идентификаторы производителей мобильных телефонов).
- Создать таблицу companies (id, company_name, country) и добавить в нее производителей и страны их регистрации.
- Вывести данные обо всех владельцах телефонов в форме идентификатор владельца, фамилия, имя, номер телефона, модель.
- подсчет количества владельцев с помощью count, если стоимость телефона <=1200 BYN.
- Суммарную стоимость всех телефонов с помощью sum, стоимость которых >1350 BYN
- максимальная и минимальная стоимость модели телефона с помощью max и min
- Используя инструкцию inner join вывести полные сведения о владельцах, модели телефона, номере телефона и производителе Apple.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий название и описание категории с номером 2.
2. Составьте запрос, возвращающий название отслеживаемого товара, название магазина, в котором он был куплен и потраченную на этот товар сумму для тех покупок, в которых эта сумма превышает 140 руб, отсортированные в следующем порядке: название товара по алфавиту, название магазина по алфавиту, потраченная сумма в порядке увеличения.

Вариант 27

Задание 2

Упражнение 2.3

Рабочий»: id, фамилия; имя; отчество; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); национальность; дата рождения (год, месяц число); № цеха; табельный номер; образование; год поступления на работу; стаж, оклад.

Добавить сведения о сотрудниках с разным стажем работы и образованием (не менее 10).

Упражнение 2.4

Выполнить запросы:

- Вывести данные о рабочих, стаж работы которых > 2 лет.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых school_id (тип integer и содержит идентификаторы школ или училищ).
- Создать таблицу schools (id, school_name, school_city) и добавить в нее учебные заведения и города, где находятся.
- Вывести данные обо всех рабочих в форме идентификатор рабочего, фамилия, образование, стаж.
- подсчет количества рабочих с помощью count, если оклад >=390 BYN.
- суммарный оклад с помощью sum, если стаж работы <4 лет.
- максимальный и минимальный оклад рабочих с помощью max и min
- Используя инструкцию inner join вывести полные сведения о рабочих, образовании и учебном заведении для города Минск.

Задание 3

Упражнение 3.2

1. Составьте запрос, выбирающий из таблиц Goods и Goods_Spending и возвращающий перечень названий отслеживаемых товаров и всех потраченных на них сумм.
2. Составьте запрос, возвращающий названия всех категорий покупок, отсортированных по алфавиту по убыванию.

Вариант 28

Задание 2

Упражнение 2.3

«Товар»: *id, наименование; количество; цена; дата изготовления (год, месяц число); изготовитель (название), номер телефона изготовителя; адрес (почтовый индекс, страна, область, район, город, улица, дом, офис).*

Добавить сведения не менее чем о 10 товарах разных производителей и разных категорий.

Упражнение 2.4

Выполнить запросы:

- Вывести данные об товарах, цена которых меньше 10 BYN.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `shop_id` (тип `integer` и содержит идентификаторы магазинов).
- Создать таблицу `shops` (`id, shop_name, shop_city`) и добавить в нее магазины и города, где они находятся.
- Вывести данные обо всех товарах в форме идентификатор товар, наименование, цена.
- подсчет количества товаров с помощью `count`, если `цена >= 5 BYN`.
- суммарная стоимость всех товаров, произведенных в 2020 году.
- максимальная и минимальная цена товаров с помощью `max` и `min`.
- Используя инструкцию `inner join` вывести полные сведения об товарах и магазинах для сети Корона.

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам `Categories` и `Spending`s, который возвращает название категории покупки, название магазина и потраченную сумму для всех покупок, относящихся к категории с номером 3.
2. Составьте запрос, возвращающий названия категорий с номерами 1, 2 и 4 (с использованием ключевого слова `IN`)

Вариант 29

Задание 2

Упражнение 2.3

«Владелец телефона»: *id; фамилия; имя; отчество; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); № телефона, тариф, цена тарифа.*

Добавить сведения не менее чем о 10 пользователях с разными тарифами и стоимостью.

Упражнение 2.4

Выполнить запросы:

- Вывести данные об тарифах, цена которых меньше 10 BYN.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из

- которых branch_id (тип integer и содержит идентификаторы офисов продаж).
- Создать таблицу branches (id, branch_name, branch_city) и добавить в нее филиалы компании и города, где они находятся.
- Вывести данные обо всех тарифах в форме идентификатор абонента, фамилия, цена тарифа.
- подсчет количества тарифов с помощью count, если цена ≥ 5 BYN.
- суммарная стоимость всех тарифов из филиалов г. Минска.
- максимальная и минимальная цена тарифов с помощью max и min.
- Используя инструкцию inner join вывести полные сведения о владельцах номеров и филиалах компании, их обслуживавших.

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам Goods и Goods_Spendings, возвращающий наименование товара и количество купленных единиц по всем покупкам для товара «Траты в WoT».
2. Составьте запрос, возвращающий названия отслеживаемых товаров и все потраченные на них суммы в следующем порядке: названия отсортированы по алфавиту, а суммы по каждому товару отсортированы в порядке увеличения.

Вариант 30

Задание 2

Упражнение 2.3

Школьник: id, фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц, число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); средний балл, класс.

Добавить сведения не менее чем о 10 учениках разных классов, роста и веса.

Упражнение 2.4

Выполнить запросы:

- Вывести данные о школьниках, вес которых меньше 40 кг.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых school_id (тип integer и содержит идентификаторы школ).
- Создать таблицу schools (id, school_name, school_city) и добавить в неё школы и города, где они находятся.
- Вывести данные обо всех школьниках в форме идентификатор школьника, фамилия, класс.
- подсчет количества школьников с помощью count, если рост ≥ 130 см.
- средний балл всех школьников школ г. Минска.
- максимальный и минимальный средний балл с помощью max и min.
- Используя инструкцию inner join вывести полные сведения о школьниках всех школ.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий название и описание категории с номером 1.
2. Составьте запрос, возвращающий название отслеживаемого товара, название магазина, в котором он был куплен и потраченную на этот товар сумму для тех покупок, в которых эта сумма превышает 200 руб, отсортированные в следующем порядке: название товара по алфавиту, название магазина по алфавиту, потраченная сумма в порядке увеличения.

Вариант 31

Задание 2

Упражнение 2.3

«Пациент»: id; фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); номер медицинской карты; диагноз; группа крови; отделение.

Добавить записи не менее чем о 10 пациентов разных отделений.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про пациентов с 14 отделения.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых hospital_id (тип integer и содержит идентификаторы больниц).
- Создать таблицу hospital (id, hos_name, hos_description).
- Вывести данные о пациентах в форме идентификатор пациента, фамилия, имя, дата рождения; диагноз, номер отделения, название больницы.
- подсчет количества пациентов с помощью count, если отделение = 3
- средний вес пациентов с помощью average, если рост > 164 см.
- максимальный и минимальный вес пациента с помощью max и min.
- Используя инструкцию inner join вывести полные сведения о пациентах и больницы для больниц с id = 1 и 2.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий описание категории с номером 4.
2. Составьте запрос, обращающийся ко всем четырём таблицам (Categories, Spendings, Goods, Goods_Spendings), возвращающий следующую информацию обо всех покупках отслеживаемых товаров: название категории, название товара, название магазина и дату покупки.

Вариант 32

Задание 2

Упражнение 2.3

«Кинолента»: *id, название; режиссер (фамилия; имя); год выхода; страна; стоимость; доход; прибыль, цена билета, жанр кино.*

Добавить сведения не менее чем о 10 кинолентах разных держателей.

Упражнение 2.4

Выполнить запросы:

- Вывести данные об кинофильмах, доход которых меньше 100 000 BYN.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `cinematheatre_id` (тип `integer` и содержит идентификаторы кинотеатров).
- Создать таблицу `cinematheatres` (`id`, `ct_name`, `ct_city`) и добавить в неё кинотеатры и города, где они находятся.
- Вывести данные обо всех кинолентах в форме идентификатор киноленты, название, цена билета.
- подсчет количества кинолент с помощью `count`, если цена билета ≥ 3 BYN.
- суммарный доход всех кинолент, которых были показаны в кинотеатрах г. Минска.
- максимальная и минимальная прибыль с помощью `max` и `min`.
- Используя инструкцию `inner join` вывести полные сведения о кинолентах и кинотеатрах, где они проходили.

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам `Goods`, `Goods_Spendings`, `Spendings`, возвращающий наименование товара и количество купленных единиц по всем покупкам для товара «Пиво», дату покупки. Таким образом, ваш запрос должен обращаться к трём таблицам и возвращать наименование товара, количество купленных единиц и дату каждой покупки для товара «Пиво».

2. Составьте запрос, возвращающий значения всех полей таблицы `Spendings` для покупок на сумму более 1200 руб.

Вариант 33

Задание 2

Упражнение 2.3

«Покупатель»: *id; фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); номер кредитной карточки; номер банковского счета; сумма на счёте.*

Упражнение 2.4

Выполнить запросы:

- Вывести данные о покупателях города Пинск.

- Используя инструкцию alter, добавить дополнительные столбцы, один из которых shop_id (тип integer и содержит идентификаторы магазинов).
- Создать таблицу shop (id, shop_name, shop_description).
- Вывести данные обо всех покупателях в форме идентификатор покупателя, фамилия, имя, номер телефона, номер кредитной карточки, название магазина.
- подсчет количества покупателей с помощью count, если пол=женский
- суммарный вес покупателей с помощью sum, если пол=муж
- максимальную и минимальную сумму ea счете с помощью max и min
- Используя инструкцию inner join вывести сведения о покупателе и магазинах для магазинов с id= 1 и 3.

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам Goods и Goods_Spendings, возвращающий наименование товара и количество купленных единиц по всем покупкам для товара «Пиво».
2. Составьте запрос к таблице Spendings, возвращающий название магазина и потраченную сумму для покупок, совершённых не позже 3 июня 2015 года.
Подсказка. Эту дату в запросе нужно указать в виде такой текстовой строки: '2015-06-03'

Вариант 34

Задание 2

Упражнение 2.3

«Товар»: id; наименование; стоимость; срок хранения; сорт; дата выпуска; срок годности.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про товары срок годности которых истекает в этом году.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых category_id (тип integer и содержит идентификаторы категорий товаров).
- Создать таблицу category (id, cat_name, cat_description).
- Вывести данные обо всех товарах в форме идентификатор товара, наименование, дата выпуска, название категории товара.
- подсчет количества товаров с помощью count, если стоимость=49 руб
- суммарная стоимость товаров с помощью sum, если год выпуска>2016
- максимальная и минимальная стоимость с помощью max и min
- Используя инструкцию inner join вывести полные сведения о товарах и категории для категории с id=2.

Задание 3

Упражнение 3.2

1. Составьте запрос к таблицам Categories и Spendings, который возвращает

название категории покупки, название магазина и потраченную сумму для всех покупок, относящихся к категории с номером 2.

2. Составьте запрос, возвращающий названия категорий с номерами 1, 3 и 4 (с использованием ключевого слова IN)

Вариант 35

Задание 2

Упражнение 2.3

«Рейс»: id, марка автомобиля; номер автомобиля; пункт назначения; грузоподъёмность (в тоннах); стоимость единицы груза; общая стоимость груза. Добавить не менее 10 записей.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про автомобили, грузоподъёмность которых больше 1,5 тонн.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых category_id (тип integer и содержит идентификаторы категорий транспортных средств для перевозки грузов (см. <https://auto-tahograf.ru/vopros-otvet/o-tahografah/kategorii-transportnykh-sredstv/>)).
- Создать таблицу category (id, cat_name, cat_description).
- Вывести данные обо всех автомобилях в форме идентификатор автомобиля, марка, номер автомобиля, название категории автомобиля.
- Подсчет количества автомобилей с помощью count, если стоимость единицы единицы груза=2 руб.
- Суммарная стоимость товаров с помощью sum, если пункт назначения=«Корона-Сити».
- Максимальная и минимальная общая стоимость груза с помощью max и min.
- Средняя стоимость всех грузов для магазинов «Электросила».
- Используя инструкцию inner join вывести полные сведения об автомобилях и категории для категории=2.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий название и описание категории с номером 3.
2. Составьте запрос к таблице Spendings, возвращающий название магазина, дату покупки и сумму для покупок, сумма которых находится в диапазоне от 1000 до 2000 руб (с использованием ключевого слова between).

Вариант 36

Задание 2

Упражнение 2.3

«Военнослужащий»: id; фамилия; имя; отчество; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); национальность; дата рождения (год, месяц число); должность; звание; зарплата.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про военнослужащих в звании “сержант”.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `department_id` (тип `integer` и содержит идентификаторы частей).
- Создать таблицу `department` (`id`, `dep_name`, `dep_type`, `dep_city`).
- Вывести данные обо всех военнослужащих в форме идентификатор военнослужащего, фамилия, имя, дата рождения, должность, название части.
- подсчет количества военнослужащих с помощью `count`, если звание=лейтенант
- суммарная заработная плата с помощью `sum`, если звание<>лейтенант
- максимальная и минимальная заработная плата с помощью `max` и `min`
- Используя инструкцию `inner join` вывести полные сведения о военнослужащих и части для части с `id=3`.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий описание категории с номером 1.
2. Составьте запрос, обращающийся к таблицам `Goods` и `Goods_Spendings` и возвращающий названия отслеживаемых товаров и потраченные на них суммы, превышающие 400 руб.

Вариант 37

Задание 2

Упражнение 2.3

«Государство»: `id`; название страны; столица; государственный язык; население; площадь территории; денежная единица; государственный строй; глава государства.

Добавить записи стран, которые находятся в Европе, Азии и Африке.

Упражнение 2.4

Выполнить запросы:

- Вывести данные про государства, население которых больше 10 млн жителей.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `area_id` (тип `integer` и содержит идентификаторы частей света).
- Создать таблицу `area` (`id`, `area_name`, `area_description`, `mainland`).
- Вывести данные обо всех государствах в форме идентификатор государства, столица, площадь территории, название части света.
- подсчет количества стран с помощью `count`, если `mainland` = Евразия
- суммарное количество населения с помощью `sum`, если идентификатор части света=3
- максимальное и минимальное количество населения с помощью `max` и `min`.
- Используя инструкцию `inner join` вывести полные сведения о государствах и части света для материка Африка.

Задание 3

Упражнение 3.2

1. Составьте запрос, выбирающий из таблиц Goods и Goods_Spending и возвращающий перечень названий отслеживаемых товаров и всех потраченных на них сумм.
2. Составьте запрос, возвращающий названия всех категорий покупок, отсортированных по алфавиту.

Вариант 38

Задание 2

Упражнение 2.3

«Человек»: *id, фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира).*

Упражнение 2.4

Выполнить запросы:

- Вывести сведения о самом пожилом человеке.
- Используя инструкцию alter, добавить дополнительные столбцы, один из которых position_id (тип integer и содержит идентификаторы должностей).
- Создать таблицу category (id, position_name, position_description, position_salary).
- Вывести данные обо всех людях в форме идентификатор человека, фамилия, имя, дата рождения, название должности.
- Подсчет количества человек с помощью count, если рост>164 см.
- Суммарный вес всех человек с помощью sum, если вес< 76 кг.
- максимальный и минимальный рост с помощью max и min.
- Используя инструкцию inner join вывести полные сведения о людях и должностях для должности с id=2.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий названия и описание всех категорий расходов.
2. Составьте запрос к таблице Spendings, возвращающий названия магазинов, даты совершенных покупок и суммы покупок, отсортированные в порядке возрастания сумм.

Вариант 39

Задание 2

Упражнение 2.3

«Автомобиль»: *id; марка; цвет; серийный номер; регистрационный номер; год выпуска; год техосмотра; цена.*

Упражнение 2.4

Выполнить запросы:

- Вывести данные про автомобили, которым больше 2 лет.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `vendor_id` (тип `integer` и содержит идентификаторы заводов).
- Создать таблицу `vendor` (`id`, `ven_name`, `ven_description`).
- Вывести данные обо всех автомобилях в форме идентификатор автомобиля, наименование, год выпуска, название завода.
- подсчет количества машин с помощью `count`, если стоимость > 18000 руб
- средняя стоимость машин с помощью `average`, если год выпуска = 2020
- максимальная и минимальная стоимость с помощью `max` и `min`
- Используя инструкцию `inner join` вывести полные сведения о машинах и заводе для завода с `id=2`.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий перечень названий всех товаров, по которым ведётся детализированный учёт расходов.
2. Составьте запрос, возвращающий названия отслеживаемых товаров и все потраченные на них суммы в следующем порядке: названия отсортированы по алфавиту, а суммы по каждому товару отсортированы в порядке уменьшения.

Вариант 40

Задание 2

Упражнение 2.3

«Товар»: *id, наименование; количество; цена нетто; цена брутто; дата изготовления (год, месяц число); изготовитель (название), номер телефона изготовителя; адрес (почтовый индекс, страна, область, район, город, улица, дом, офис).*

Добавить сведения не менее чем о 10 товарах разных производителей и разных категорий.

Упражнение 2.4

Выполнить запросы:

- Вывести данные об товарах, цена нетто которых больше 10 BYN.
- Используя инструкцию `alter`, добавить дополнительные столбцы, один из которых `shop_id` (тип `integer` и содержит идентификаторы магазинов).
- Создать таблицу `shops` (`id`, `shop_name`, `shop_city`) и добавить в нее магазины и города, где они находятся.
- Вывести данные обо всех товарах в форме идентификатор товар, наименование, цена.
- подсчет количества товаров с помощью `count`, если пцена >= 5 BYN.
- суммарная стоимость всех товаров, произведенных в 2022 году.
- максимальная и минимальная цена товаров с помощью `max` и `min`.
- Используя инструкцию `inner join` вывести полные сведения об товарах и магазинах для сети Санта.

Задание 3

Упражнение 3.2

1. Составьте запрос, возвращающий все поля всех записей таблицы Categories.
2. Составьте запрос, возвращающий название отслеживаемого товара, название магазина, в котором он был куплен и потраченную на этот товар сумму для тех покупок, в которых эта сумма превышает 250 руб, отсортированные в следующем порядке: название товара по алфавиту, название магазина по алфавиту, потраченная сумма в порядке уменьшения.

Контрольные вопросы

1. Перечислите способы создания базы данных sqlite.
2. С помощью какой команды в консоли sqlite можем просмотреть список баз данных и подключённых файлов баз данных?
3. Приведите перечень команд для экспорта данных из таблицы базы данных в файл с расширением .csv.
4. Приведите перечень команд для экспорта отдельной таблицы и всей базы данных в файл с расширением .sql и сжатый файл, например в файл с расширением .sql.tgz.
5. Как вывести из таблицы данных по строкам и по столбцам?
6. Для чего используется команда .headers в консоли sqlite?
7. Какая команды используется для вывода настроек окружения в sqlite?
8. С помощью какой команды выводится список таблиц базы данных в консоли sqlite?
9. Приведите пример запроса выборки из 2-х таблиц.
10. Приведите пример запроса для обновления данных в строках таблицы в зависимости от значения определённого поля.
11. Приведите пример функции, которая открывает соединение с файлом базы данных SQLite и возвращает объект соединения с базой данных, который будет использоваться другими функциями SQLite?
12. Приведите пример синтаксиса функции sqlite3_exec и объясните результат выполнения.
13. Какая функция закрывает соединение с базой данных, ранее открытое вызовом sqlite3_open()? Приведите пример синтаксиса.
14. Приведите пример фрагмента кода на языке C для создания таблицы в базе данных sqlite и объясните его.
15. Приведите пример фрагмента кода на языке C для вставки данных в таблицу в базе данных sqlite и объясните его.
16. Приведите пример фрагмента кода на языке C выполнением AUTOCOMMIT и TRANSACTION и объясните в чем особенности использования их.