**Лабораторная работа 12: Основы выборки SQL**

1. Для выполнения работы воспользуйтесь схемой БД из лабораторной работы №5

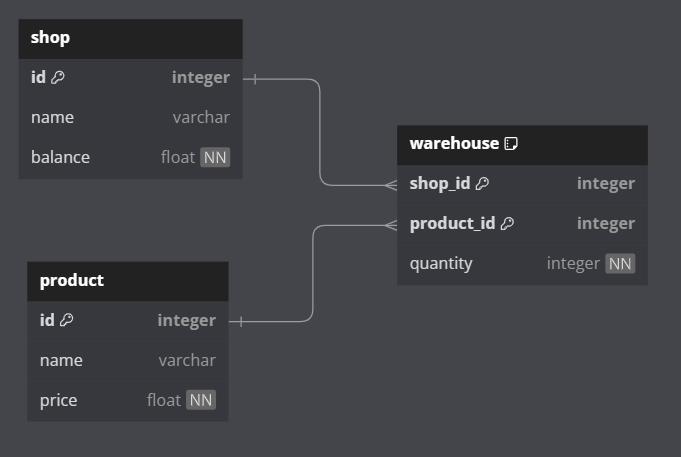


Рисунок 1 – Схемы базы данных

**Сортировка, оператор ORDER BY**

При выполнении SELECT запроса, строки по умолчанию возвращаются в неопределённом порядке. Фактический порядок строк в этом случае зависит от плана соединения и сканирования, а также от порядка расположения данных на диске, поэтому полагаться на него нельзя. Для упорядочивания записей используется конструкция ORDER BY

| SELECT поля\_таблиц FROM наименование\_таблицы WHERE ... ORDER BY столбец\_1 [ASC | DESC][, столбец\_n [ASC | DESC]] |
| --- |

Где ASC и DESC - направление сортировки:

* ASC - сортировка по возрастанию (по умолчанию)
* DESC - сортировка по убыванию

Для сортировки результатов по двум или более столбцам их следует указывать через запятую.

| ...ORDER BY столбец\_1 [ASC | DESC], столбец\_2 [ASC | DESC]; |
| --- |

|  |
| --- |

**Группировка, оператор GROUP BY**

Оператор GROUP BY в SQL используется для группировки строк из таблицы в более крупные наборы результатов на основании одного или нескольких столбцов. Эти группы включают разные записи в таблице и, соответственно, обладают разными характеристиками, которые нам могут быть весьма полезны.

Общая структура запроса с GROUP BY выглядит следующим образом

| SELECT [литералы, агрегатные\_функции, поля\_группировки] FROM имя\_таблицы GROUP BY поля\_группировки; |
| --- |

При использовании оператора GROUP BY мы перешли от работы с отдельными записями на работу с образовавшимися группами. В связи с этим мы не можем просто вывести любое поле из записи , как мы это могли делать раньше. Так как в каждой группе может быть несколько записей и в каждой из них в этом поле может быть разное значение.

При использовании GROUP BY мы можем выводить только:

* литералы, т.е. указанное явным образом фиксированные значения
* результаты агрегатных функций, т.е. вычисленные значения на основании набора значений.

**Агрегатные функции**

Агрегатная функция – это функция, которая выполняет вычисление на наборе значений и возвращает одиночное значение.

Общая структура запроса с агрегатной функцией выглядит следующим образом:

| SELECT [литералы, агрегатные\_функции, поля\_группировки] FROM имя\_таблицы GROUP BY поля\_группировки; |
| --- |

**Виды агрегатных функций**

| SUM(поле\_таблицы) | Возвращает сумму значений |
| --- | --- |
| AVG(поле\_таблицы) | Возвращает среднее значение |
| COUNT(поле\_таблицы) | Возвращает количество записей |
| MIN(поле\_таблицы) | Возвращает минимальное значение |
| MAX(поле\_таблицы) | Возвращает максимальное значение |

Ниже представлены несколько примеров использования агрегатных функций:

* Найдём количество видов магазинов и отсортируем полученный список по убыванию:

| SELECT name, COUNT(\*) as amount FROM Shop GROUP BY name ORDER BY amount DESC |
| --- |

* Найдем средний остаток на балансе магазинов отсортированный по возрастанию

| SELECT name, avg(balance) as avg\_balance from shop group by name order by avg\_balance |
| --- |

**Практическое задание**

1. Создайте дополнительную таблицу “Сотрудник” выполнив следующую команду

| CREATE TABLE worker ( worker\_id INTEGER PRIMARY KEY, shop\_id INTEGER REFERENCES product (id), name VARCHAR(255), salary INTEGER NOT NULL, position VARCHAR(255)); |
| --- |

1. Внесите в таблицу данные.
2. К таблицам напишите запросы операциями группировки и сортировки. Также реализуйте запросы с использованием агрегатных функций для таблицы “Сотрудник”