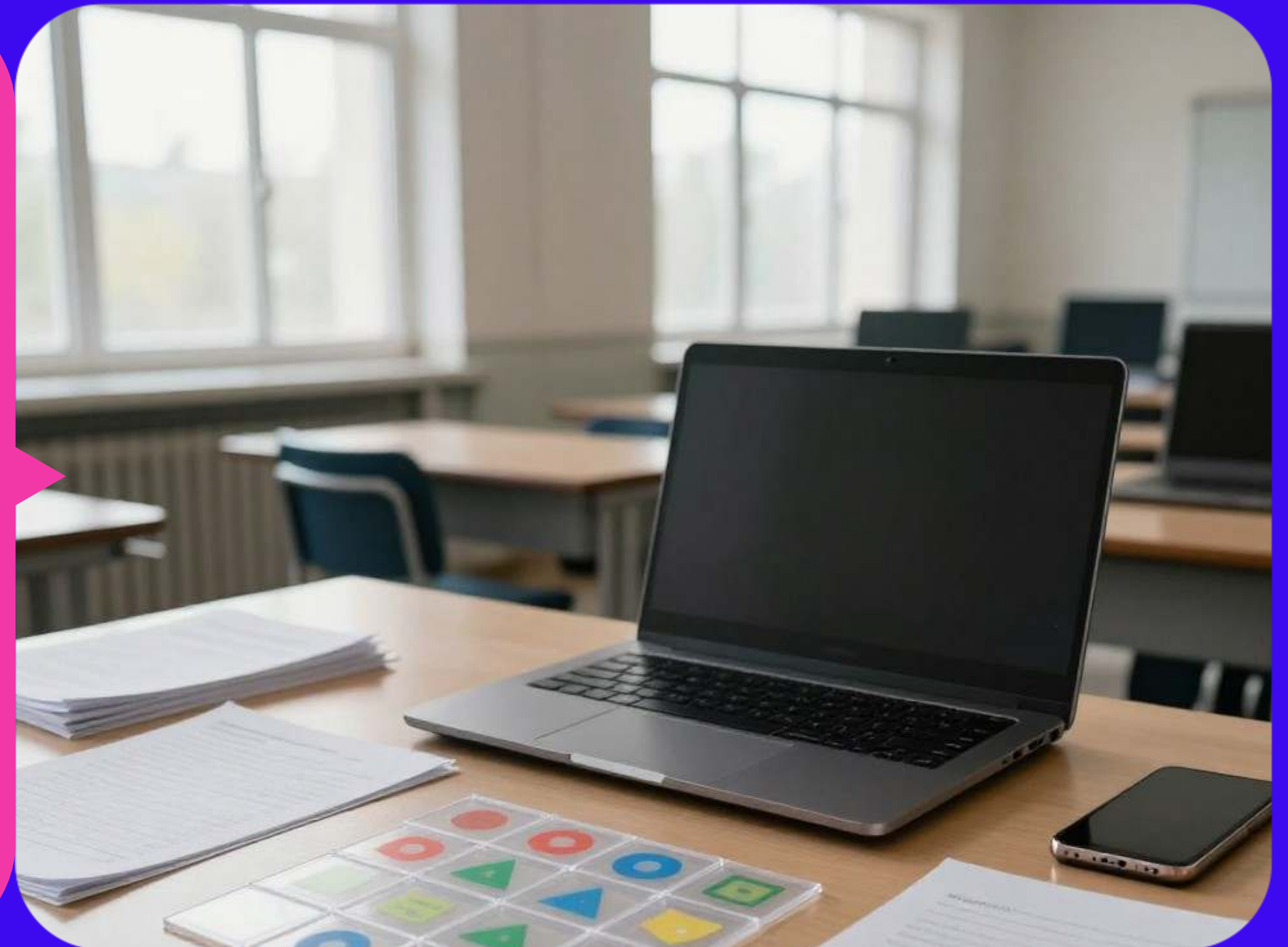


Запросы на выборку данных: как получить нужную информацию

SELECT извлекает нужные строки из таблиц без ручного поиска по всем данным.

Где нужны запросы выборки и зачем они помогают

База данных хранит сведения в таблицах: строка — одна запись, столбец — одно поле. SQL — стандартный язык для реляционных баз. Запросы выборки дают точный ответ: например, найти учеников по оценке, книги автора или товары по цене.



Структура таблицы базы данных

↗ Таблица состоит из строк и столбцов. Каждая строка — это одна запись, например сведения об одном ученике, а каждый столбец хранит отдельный признак: фамилию, класс, возраст, оценку или город.

↗ Отдельно важную роль играет ключевое поле — оно отличает одну запись от другой. В учебных примерах это номер ученика или код товара: так система понимает, где «какой именно» объект записан.

↗ Названия столбцов — это подсказки смысла данных. Они помогают понять, что именно лежит в каждом поле, и как использовать это при поиске: например, по какому столбцу сравнивать значения и отбирать строки.

Команда SELECT и ее части

01

Базовый запрос на выборку строится из SELECT, списка полей, FROM и имени таблицы. Например, `SELECT * FROM students;` возвращает все столбцы, а `SELECT name, class FROM students;` — только имя и класс.

02

Когда вы выбираете конкретные поля, результат получается короче и удобнее. Меньше лишней информации — проще читать и сравнивать: особенно если таблица содержит десятки столбцов и много данных.



Основные операторы выборки в SQL

Сравните роли ключевых частей запроса: они отвечают за источник данных, отбор, порядок и размер результата.

Понимание «что делает каждый оператор» помогает быстро собрать правильный запрос и не путать отбор строк с сортировкой и ограничением.

Базовый синтаксис SQL (общие правила).

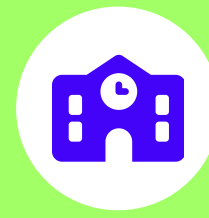
Оператор	Задаёт	Итоговое влияние
SELECT	какие поля показать	определяет состав столбцов
FROM	где искать	указывает таблицу-источник
WHERE / ORDER BY / LIMIT	как отбирать, упорядочивать и ограничивать	WHERE фильтрует, ORDER BY меняет порядок, LIMIT уменьшает число строк

Выбор нужных столбцов: когда звездочка не нужна



Показывайте только важное

Запрос может вернуть все поля через *, а может — только нужные столбцы. Если в таблице много данных, выбор нескольких полей делает ответ компактнее и снижает объем лишней информации.



Примеры из школы

Например, можно вывести имя и класс ученика, или фамилию и оценку. Такие маленькие выборки удобны для проверки успеваемости и подготовки коротких списков без лишних столбцов.



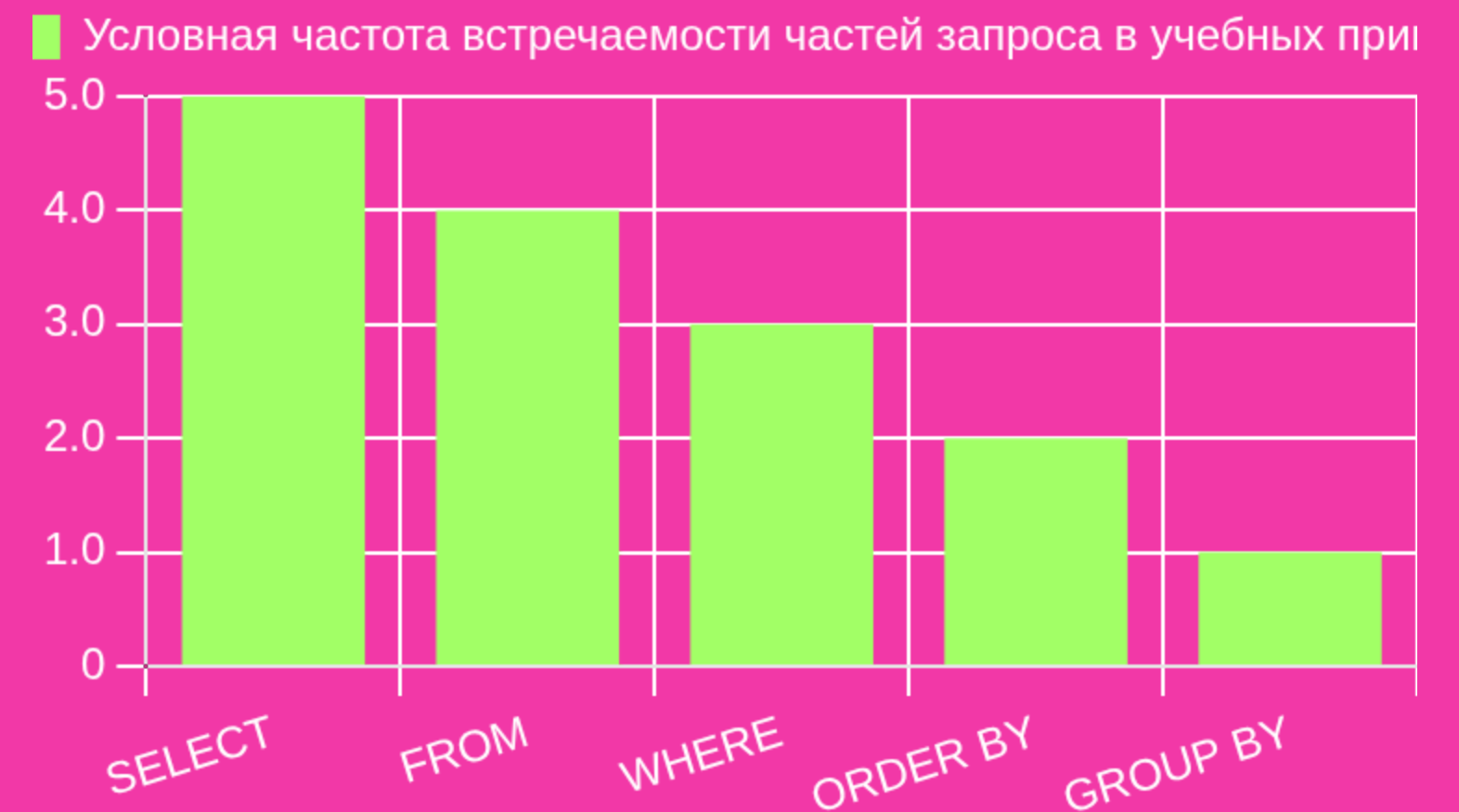
Примеры из товаров и справочников

Для магазинов логично выбирать название и цену товара, а для справочников — город и население. Такой подход особенно полезен, когда таблица содержит десятки столбцов и полей слишком много.

Какие части запроса чаще всего используют

➤ Почти каждый учебный пример начинается с SELECT и FROM. Остальные элементы подключают по задаче: WHERE для поиска, ORDER BY для порядка, GROUP BY — для группировки.

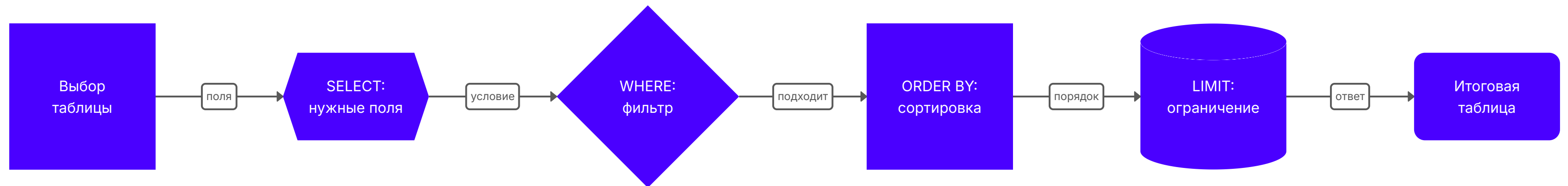
➤ Вывод простой: базовый запрос собирается как конструктор из нескольких частей, и вы добавляете нужные элементы только тогда, когда они реально требуются.



Сводный обзор типовых учебных задач; значения на диаграмме — иллюстративные, без привязки к точной статистике.

Как выполняется запрос на выборку

Шаги запроса: от выбора таблицы до получения итоговой таблицы.



Условия отбора: WHERE

01

Оператор WHERE оставляет только те строки, которые соответствуют условию. Он нужен для поиска по заданным правилам: например, выбрать учеников определенного возраста, класса или с конкретной оценкой.

02

В условиях используют знаки сравнения: =, <, >, <=, >=. Для объединения нескольких условий применяют логические связки AND и OR, чтобы составлять более точные запросы.

03

Примеры условий: возраст больше 12 лет, оценка равна 5, цена товара меньше 500 рублей, ученик учится в 9 классе. Такие формулировки превращаются в проверяемые правила для строк.

04

Важно помнить: WHERE не меняет данные в таблице. Он только выбирает подходящие записи и формирует результат. Поэтому одна и та же таблица может давать разные выборки под разные условия.

Сортировка результатов: ORDER BY

01

ORDER BY упорядочивает строки в ответе: по возрастанию или по убыванию. Для букв и названий сортировка работает по алфавиту, а для чисел — по величине (оценки, цены, даты).

02

Запись ORDER BY name ASC задает возрастание, а ORDER BY score DESC — убывание. Наглядный смысл такой: состав строк тот же, но их порядок становится удобным для чтения и сравнения.



WHERE, ORDER BY и LIMIT: что меняет каждый оператор

Оператор	Что делает	Как меняется результат
WHERE	Фильтрует по условию	Убирает неподходящие строки
ORDER BY	Сортирует	Оставляет те же строки, меняет порядок
LIMIT	Ограничивает	Возвращает только первые N строк



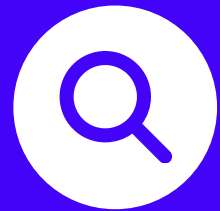
Эти три части по-разному влияют на результат одного и того же запроса: одна отбирает строки, другая меняет их порядок, а третья ограничивает количество возвращаемых записей. Источник: базовые правила SQL-запросов.



Чтобы понимать ответ запроса, важно мысленно «разложить» его на этапы: сначала отбор, затем порядок, потом ограничение. Так проще не путать смысл WHERE и ORDER BY.

Базовые правила синтаксиса SQL (учебный материал)

Поиск по образцу и списку значений



LIKE и шаблон

LIKE помогает искать по части слова или по образцу. Например, LIKE 'Ан%' находит строки, начинающиеся с «Ан», а оставшаяся часть может быть любой длины.



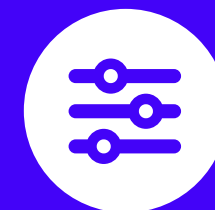
Символ % в LIKE

Знак % в шаблоне означает «любое количество символов». Поэтому запрос не требует точного совпадения всего слова и подходит для фамилий, городов и названий.



IN: проверка списка

Оператор IN проверяет, входит ли значение в заранее заданный набор. Например, класс IN ('7А','7Б','7В') вернет только записи из указанных вариантов.



Когда выбирать какой оператор

Используйте LIKE, когда варианты похожи по началу или формату. Используйте IN, когда перечень значений известен заранее и выбранных вариантов всего несколько.

Как уменьшается объем данных после фильтрации

➤ Если добавить сортировку, набор строк не становится меньше: ORDER BY готовит данные к просмотру. Главное сокращение происходит, когда включают WHERE и затем LIMIT.

➤ График показывает путь от «всего» к «нужному»: сначала фильтр убирает лишние записи, сортировка делает ответ удобным, а LIMIT дает короткий список для дальнейшей работы.



Иллюстративные значения для понимания: условная таблица из 100 строк

avg_score

Псевдоним упрощает чтение результата,
особенно когда в исходных названиях много
символов или используются вычисления.

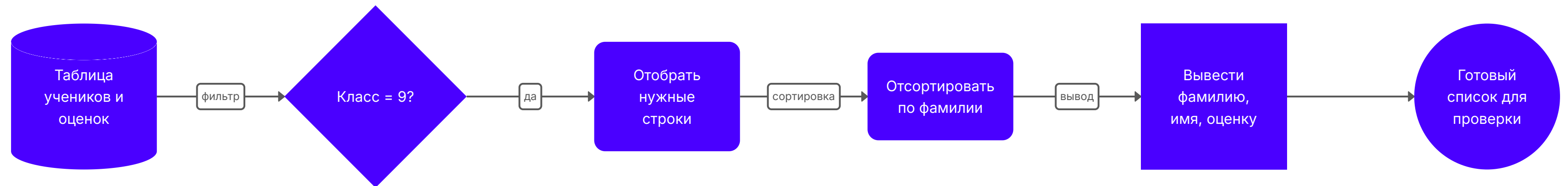
Учебный синтаксис SQL: ключевое слово AS

AS позволяет заменить длинные или технические названия на короткие подписи в выводе, например средняя_оценка AS avg_score, делая таблицу понятнее при анализе.



Пример решения учебной задачи запросом

Задача: выбрать записи только для 9 класса, отсортировать фамилии по алфавиту и вывести фамилию, имя и оценку.



Ошибки в запросах на выборку



Частая ошибка — пропустить имя таблицы после FROM. Тогда SQL не понимает, откуда брать данные. В результате вы получаете сообщение об ошибке синтаксиса или невозможность выполнить запрос.



Лишняя запятая в списке полей или неправильный порядок ключевых слов нарушают структуру запроса. Пример: SELECT перечисление должно идти в нужной последовательности, иначе система не сможет распознать запрос.



Неверное название столбца тоже приводит к сбою: запрос обращается к полю, которого нет. Ещё опаснее перепутать знаки сравнения, например «=» и «<>», из-за чего фильтр станет неверным.



Текстовые значения записываются в кавычках, а числа — без. Если поставить кавычки там, где они не нужны, или забыть кавычки для текста, запрос часто возвращает пустой результат или ошибку.

Запросы в реальной жизни



Электронный журнал: найти «нужных» учеников

Учитель открывает электронный журнал и быстро находит учеников с определенной оценкой или теми, у кого много пропусков. Это помогает подготовить план повторения и увидеть картину без ручного перебора строк.



Магазин: подобрать товары по бюджету

Покупатель задает условия: цена меньше заданной суммы и нужная категория. Система выдает короткий список подходящих товаров. Так запросы помогают не читать десятки страниц и сразу сравнить варианты.



Библиотека: найти автора и нужный год

В каталоге библиотеки можно искать книги по автору, а затем уточнить год издания. Запросы выборки позволяют получить точные результаты из большого массива карточек. Пользователь быстрее находит нужный материал для учебы или проекта.

Ключевые команды SQL для выборки

➤ Диаграмма показывает, как «собирается» ответ запроса: сначала выбирают поля и источник, затем отбирают строки, задают порядок и при необходимости ограничивают количество.

➤ Когда элементы работают вместе, запрос становится понятным по смыслу: WHERE отвечает за точность, ORDER BY — за удобный просмотр, а LIMIT — за краткость выдачи.

SELECT FROM WHERE
ORDER BY LIMIT GROUP BY



Иллюстративная схема (веса условные, отражают учебное распределение ролей команд)

Что нужно запомнить о выборке данных

01

SELECT получает данные из таблицы и задает, какие столбцы попадут в результат. Чем точнее выбор полей, тем меньше лишней информации и тем проще анализировать ответ.

02

WHERE отбирает строки по условию, а ORDER BY упорядочивает результат по возрастанию или убыванию. LIMIT дополнительно сокращает выдачу до нужного числа строк.

03

LIKE ищет по образцу с шаблонами, IN проверяет значение по списку, а AS задает псевдонимы. Вместе эти элементы экономят время при работе с большими таблицами.

Итог: запросы выборки как инструмент работы с данными

Запросы на выборку позволяют быстро находить нужные сведения, сравнивать записи и делать выводы по таблицам. Основа — SELECT, FROM, WHERE, ORDER BY, LIMIT, LIKE, IN и AS, нужные для дальнейшего обучения.

