ЛАБОРАТОРНАЯ РАБОТА №2

**ТИПЫ ДАННЫХ.   
ФУНКЦИИ РАБОТЫ СО СТРОКАМИ И ДАТАМИ.**

**ТЕОРЕТИЧЕСКАЯ ЧАСТЬ**

# **Сортировка записей**

При выполнении SELECT запроса, строки по умолчанию возвращаются в неопределённом порядке. Для упорядочивания записей используется конструкция **ORDER BY**.

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

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

*SELECT name FROM sal ORDER BY name;*

# **Оператор LIKE**

## *... WHERE поле\_таблицы [NOT] LIKE шаблон\_строки*

Шаблон может включать следующие специальные символы:

% - последовательность любых символов (0 и более символов)

\_ – любой единичный символ

## [Примеры](https://sql-academy.org/ru/guide/operator-like" \l "primery)

## *... WHERE поле\_таблицы LIKE 'text%'* – сопоставляется любым строкам, начинающимся на «text»

* *... WHERE поле\_таблицы LIKE '%text'* – сопоставляется любым строкам, заканчивающимся на «text»
* *... WHERE поле\_таблицы LIKE '\_ext' –* сопоставляется строкам, имеющим длину 4 символа, при этом 3 последних обязательно должны быть «ext».
* *... WHERE поле\_таблицы LIKE 'begin%end'* - сопоставляется строкам, начинающихся на «begin» и заканчивающихся на «end»

В PostgreSQL шаблоны чувствительны к регистру.

Для поиска без учета регистра используется оператор**ILIKE.**

Если нужно найти строки, содержащие символы % и \_ как обычный текст, их необходимо экранировать с помощью **ESCAPE-символа.**

* *...WHERE поле\_таблицы LIKE 'шаблон\_строки'   
  ESCAPE 'символ\_экранирования'*

Например, вы хотите получить идентификаторы задач, прогресс которых равен 3%:

*SELECT job\_id FROM Jobs WHERE progress LIKE '3!%' ESCAPE '!';*

# **Функции работы со строками:**

**CHR** – возвращает символ с указанным ASCII-кодом.

*SELECT CHR(72)*

**CONCAT** – объединяет переменное количество аргументов в одну строку. Null значения игнорируются.

*SELECT CONCAT('Post', 'greSQL')*

**SUBSTRING** – извлекает подстроку из строки начиная с заданной позиции   
на заданную длину.

*SELECT SUBSTRING(‘'string' FROM start FOR length ), string* - исходная строка, *start* - начальная позиция (с 1), *length* - количество символов для извлечения.

*SELECT SUBSTRING('string' FROM 1 FOR 2)*

**LENGTH** – возвращает количество символов в строке.

*SELECT LENGTH('PostgreSQL')*

**POSITION** – возвращает начальную позицию подстроки в строке или ноль,   
если не найдено.

*SELECT POSITION('om' IN 'Thomas')*

**REPEAT** – повторяет строку указанное количество раз.

*SELECT REPEAT('111', 3)*

**REPLACE** – заменяет все вхождения подстроки другой подстрокой в строке.

*SELECT REPLACE('abcdef', 'cd', 'XX')*

**REVERSE** – переворачивает строку.

*SELECT REVERSE('drawer')*

**LOWER** – преобразует строку в нижний регистр.

*SELECT LOWER('POSTGRESQL')*

**UPPER** – преобразует строку в верхний регистр.

*SELECT UPPER('postgresql')*

**LPAD** – дополняет строку слева до указанной длины с помощью строки заполнителя.

*SELECT LPAD('9', 5, '0')*

**RPAD** – дополняет строку справа до указанной длины с помощью строки заполнителя.

*SELECT RPAD('9', 5, '0')*

**LTRIM** – удаляет ведущие пробелы из строки.

*SELECT LTRIM(' Hello ')*

**RTRIM** – удаляет завершающие пробелы из строки.

*SELECT RTRIM(' Hello ')*

**TRIM** – удаляет самую длинную строку, содержащую только указанные символы, из начала, конца или обоих концов строки.

*TRIM([LEADING | TRAILING | BOTH] [characters] FROM string)  
LEADING | TRAILING | BOTH* – указывает, откуда удалять (BOTH по умолчанию), *characters* –символы для удаления (пробел по умолчанию), *string* – входная строка.

*SELECT TRIM( BOTH 'x' FROM 'xTomx')*

# **Функции работы с датами:**

**AGE** – возвращает разницу между двумя метками времени как интервал.

*AGE(timestamp1, timestamp2),* *timestamp1* - более поздняя метка времени, *timestamp2* - Более ранняя метка времени

*SELECT AGE('2022-12-05', '2012-06-12')*

**CURRENT\_DATE** – возвращает текущую дату.

*SELECT CURRENT\_DATE*

**CURRENT\_TIME** – возвращает текущее время.

*SELECT CURRENT\_TIME*

**DATE** – извлекает часть даты из даты/времени.

*SELECT DATE('2022-12-05 10:37:22')*

**DATE\_PART** – извлекает подполя из значения даты/времени.

*DATE\_PART(field, source), field* - поле для извлечения (year, month, day и т.д.), *source* - значение даты/времени (timestamp).

*SELECT DATE\_PART('year', TIMESTAMP '2023-01-01')*

**DATE\_TRUNC** – обрезает дату/время до указанной точности.

*DATE\_TRUNC(field, source),* *field* -точность (year, month, day и т.д.), *source* - значение даты/времени.

*SELECT DATE\_TRUNC('month', TIMESTAMP '2023-02-15')*

**EXTRACT** – извлекает подполя из значения даты/времени (аналогично DATE\_PART)

*EXTRACT(field FROM source),* *field* - поле для извлечения, *source* - значение даты/времени.

*SELECT EXTRACT( YEAR FROM TIMESTAMP '2023-01-01')*

**MAKE\_DATE** – создает дату из года, месяца и дня.

*MAKE\_DATE(year, month, day)*

*SELECT MAKE\_DATE(2025, 6, 5)*

**MAKE\_INTERVAL** – создает интервал из указанных компонентов.

*MAKE\_INTERVAL(years, months, weeks, days, hours, mins, secs)*

*SELECT**MAKE\_INTERVAL* ***(****2012, 3, 2, 1, 12, 13, 14)*

**MAKE\_TIME** – создает время из часа, минуты и секунды.

*MAKE\_TIME(hour, minute, second)*

*SELECT MAKE\_TIME(12, 30, 45)*

**NOW** – возвращает текущую дату и время с временной зоной.

*SELECT NOW()*

**TO\_CHAR** – преобразует дату/время или число в строку по формату.

*TO\_CHAR(timestamp, format), timestamp* - значение для преобразования, *format* - строка формата.

*SELECT TO\_CHAR(NOW(), 'YYYY-MM-DD')*

**TO\_DATE** – преобразует строку в дату с использованием указанного формата.

*TO\_DATE(string, format), string* - строка для преобразования*, format* - формат

*SELECT TO\_DATE('November 13, 1998', 'Month DD, YYYY')*

**TO\_TIMESTAMP** – преобразует Unix-метку времени в метку времени.

*TO\_TIMESTAMP(unix\_timestamp)*

S*ELECT TO\_TIMESTAMP(1717584000)*

## **[Использование регулярных выражений](https://sql-academy.org/ru/guide/operator-regexp" \l "sintaksis-regulyarnyh-vyrazhenij)**

*... WHERE table\_field ~ 'pattern'; – с учетом регистра*

*... WHERE table\_field ~\* 'pattern'; – без учета регистра*

Для экранирования специальных символов (например, ., \*, +, ?, [, ], (, ), {, }, |, \) используется одинарный обратный слеш — \.

[**Специальные символы и структуры**](https://sql-academy.org/ru/guide/operator-regexp#specialnye-simvoly-i-struktury)

|  |  |
| --- | --- |
| *Символы и структуры* | *Чему соответствует* |
| \* | 0 или более экземпляров предшествующей строки |
| + | 1 или более экземпляров предшествующих строк |
| . | Любой одиночный символ |
| ? | 0 или 1 экземпляр предшествующей строки |
| ^ | Соответствует началу строки |
| $ | Соответствует окончанию строки |
| [abc] | Любой символ, указанный в квадратных скобках |
| [^abc] | Любой символ, не указанный в квадратных скобках |
| [A-Z], [А-Я] | Соответствует любой заглавной букве латинского и кириллического алфавита соответственно |
| [a-z], [а-я] | Соответствует любой строчной букве латинского и кириллического алфавита соответственно |
| [0-9] | Соответствует любой цифре |
| p1|p2|p3 | Соответствует любому из паттернов p1 или p2 или p3 |
| {n} | n экземпляров предыдущей строки |
| {m,n} | от m до n экземпляров предыдущей строки |

*SELECT \* FROM Users WHERE email ~ '@(outlook\.com|icloud\.com)$'*

**ПРАКТИЧЕСКОЕ ЗАДАНИЕ**

*Вариант 1*

Выполнить задания 1-2 с использованием оператора LIKE

1. Вывести имена покупателей, которые начинаются и заканчиваются на букву v.
2. Вывести названия городов из таблицы продуктов, в которых вторая буква о, отсортированные в обратном порядке.

Выполнить задания 3-4 с использованием строковых функций

1. Вывести в верхнем регистре все города из таблицы продавцов, имеющие длину более 6 символов. Пробелы в названиях городов должны быть удалены, а названия выведены по алфавиту.
2. Вывести в одном столбце номера покупателей, соединенные с его именем и городом, в формате: ‘2001-IVANOV-PERM, только для тех покупателей, чье имя содержит подстроку ‘ov’.

Выполнить задания 5-6 с использованием функций работы с датами

1. Вывести временной интервал между сегодняшней датой и датой самого раннего заказа.
2. Вывести текущее время в формате 12тичасовых часов, с обозначением “am” для первой половины суток и “pm” – для второй.

Выполнить задания 7-8 с использованием регулярных выражений

1. Вывести имена покупателей, начинающиеся с букв A-I и заканчивающиеся на ‘ov’.
2. Вывести имена продавцов, в которых содержится две подряд идущих согласных, не стоящих в конце слова.

*Вариант 2*

Выполнить задания 1-2 с использованием оператора LIKE

1. Вывести имена продавцов, имя которых начинается и заканчивается на букву a.
2. Вывести названия городов из таблицы продуктов, в которых присутствует буква e, отсортированные в алфавитном порядке.

Выполнить задания 3-4 с использованием строковых функций

1. Вывести в верхнем регистре все города из таблицы покупателей, имеющие длину не менее 7 символов. Каждое название должно выведено в обратном порядке.
2. Вывести в одном столбце номера продавцов, соединенные с его именем и городом, в формате: ‘3001/DNS/NOVOSIBIRSK, только для тех продавцов, чье имя содержит подстроку ‘i’.

Выполнить задания 5-6 с использованием функций работы с датами

1. Вывести число минут в момент выполнения лабораторной работы.
2. Вывести для каждой уникальной даты заказа запись. состоящую из двух столбцов: день в формате «dd» и отдельно день недели заказа.

Выполнить задания 7-8 с использованием регулярных выражений

1. Вывести названия городов из таблицы покупателей, начинающиеся с гласной буквы и заканчивающиеся на согласную.
2. Вывести имена покупателей, которые содержат комбинацию букв ‘iv’ или ‘ir’  
   в любом регистре.

*Вариант 3*

Выполнить задания 1-2 с использованием оператора LIKE

1. Вывести названия продуктов, которые начинаются и заканчиваются на букву s.
2. Вывести названия городов из таблицы покупателей, в которых буква r стоит на предпоследнем месте, отсортированные в прямом порядке.

Выполнить задания 3-4 с использованием строковых функций

1. Вывести в нижнем регистре все имена из таблицы продавцов, имеющие длину более 5 символов. Первая буква в каждом названии должна быть удалена, а строки выведены в алфавитном порядке.
2. Вывести в одном столбце рейтинг покупателей, соединенный со словом «scores» и именем в нижнем регистре, в формате: ‘ivanov: 100 scores’, только для тех покупателей, чье имя содержит подстроку ‘ev’.

Выполнить задания 5-6 с использованием функций работы с датами

1. Вывести, сколько дней осталось от даты выполнения лабораторной до нового года.
2. Вывести для каждой уникальной даты заказа запись. состоящую из трех столбцов: день в формате «dd», месяц в формате «mm», год в формате «yy».

Выполнить задания 7-8 с использованием регулярных выражений

1. Вывести имена продавцов, начинающиеся с двух или более согласных букв.
2. Вывести названия продуктов, в которых содержится буква k в любом регистре и на любом месте, кроме первого.

*Вариант 4*

Выполнить задания 1-2 с использованием оператора LIKE

1. Вывести названия городов покупателей, которые начинаются на букву y или заканчиваются на букву k.
2. Вывести названия продуктов, в которых присутствует буква p, отсортированные в алфавитном порядке.

Выполнить задания 3-4 с использованием строковых функций

1. Вывести в нижнем регистре все имена из таблицы покупателей, имеющие длину не более 6 символов. В каждом имени удалить последние две буквы, имена должны быть выведены по обратном алфавитному порядке.
2. Вывести в одном столбце комиссионные продавцов, соединенные с символом «%» и именем в нижнем регистре, в формате: ‘dns: 0.14%’, только для тех продавцов, чье имя содержит подстроку ‘in’.

Выполнить задания 5-6 с использованием функций работы с датами

1. Вывести часовой пояс, в котором выполняется лабораторная (смещение часов относительно Гринвича).
2. Вывести для каждой уникальной даты заказа запись. состоящую из трех столбцов: день в формате «dd», месяц в формате «mm», год в формате «yy».

Выполнить задания 7-8 с использованием регулярных выражений

1. Вывести города продуктов, начинающиеся с буквы o или содержащие пробелы.
2. Вывести названия продуктов, в которых содержится буква e в любом регистре и на любом месте, кроме последнего.

*Вариант 5*

Выполнить задания 1-2 с использованием оператора LIKE

1. Вывести названия городов продавцов, которые начинаются на букву m и заканчиваются на букву w.
2. Вывести имена покупателей, в которых буква o стоит на предпоследнем месте, отсортированные в обратном порядке.

Выполнить задания 3-4 с использованием строковых функций

1. Вывести в верхнем регистре все названия продуктов, имеющие длину не более 7 символов. Все названия менее 7 символов должны быть дополнены до 7 знаков символом точки.
2. Вывести в одном столбце вес продуктов, соединенный со словом «GRAMS» и названием в верхнем регистре, в формате: ‘MONITOR: 2000 GRAMS’, только для тех продуктов, название которых содержит подстроку ‘ri’.

Выполнить задания 5-6 с использованием функций работы с датами

1. Вывести число секунд в момент выполнения лабораторной работы.
2. Вывести для каждой уникальной даты заказа запись. состоящую из двух столбцов: дата в формате «dd.mm» и отдельно год заказа.

Выполнить задания 7-8 с использованием регулярных выражений

1. Вывести имена продавцов, начинающиеся с букв A-I и заканчивающиеся на букву a или на комбинацию ‘nk’.
2. Вывести имена продавцов, в которых содержится две или более подряд идущих согласных.

*Вариант 6*

Выполнить задания 1-2 с использованием оператора LIKE

1. Вывести названия городов из таблицы продуктов, которые начинаются на букву m или на букву n и заканчиваются на букву k.
2. Вывести имена продавцов, в которых присутствует буква e, отсортированные в прямом порядке.

Выполнить задания 3-4 с использованием строковых функций

1. Вывести в нижнем регистре все города из таблицы продуктов, имеющие длину более 6 символов. Все гласные буквы в названиях городов должны быть заменены   
   на букву o.
2. Вывести в одном столбце номера продуктов, соединенные с его названием и городом, в формате: ‘1001.MONITOR.OBNINSK, только для тех продуктов, название которых содержит подстроку ‘rd’.

Выполнить задания 5-6 с использованием функций работы с датами

1. Вывести временной интервал между сегодняшней датой и датой самого позднего заказа.
2. Вывести текущее время в формате «hh.mm.ss».

Выполнить задания 7-8 с использованием регулярных выражений

1. Вывести города продавцов, которые начинаются с согласной буквы, за которой следует гласная.
2. Вывести имена покупателей, которые начинаются с любой буквы кроме букв A-D и содержат букву s в любом регистре.