МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ»

Факультет Управление и информатика в технологических системах

Кафедра Информационная безопасность

Специальность 10.05.03 «Информационная безопасность автоматизированных систем»

Отчет по практической работе №3

по дисциплине Безопасность Баз Данных

Тема: Средства определения, запроса и модификации данных языка SQL.

Выполнила: студентка 3 курса

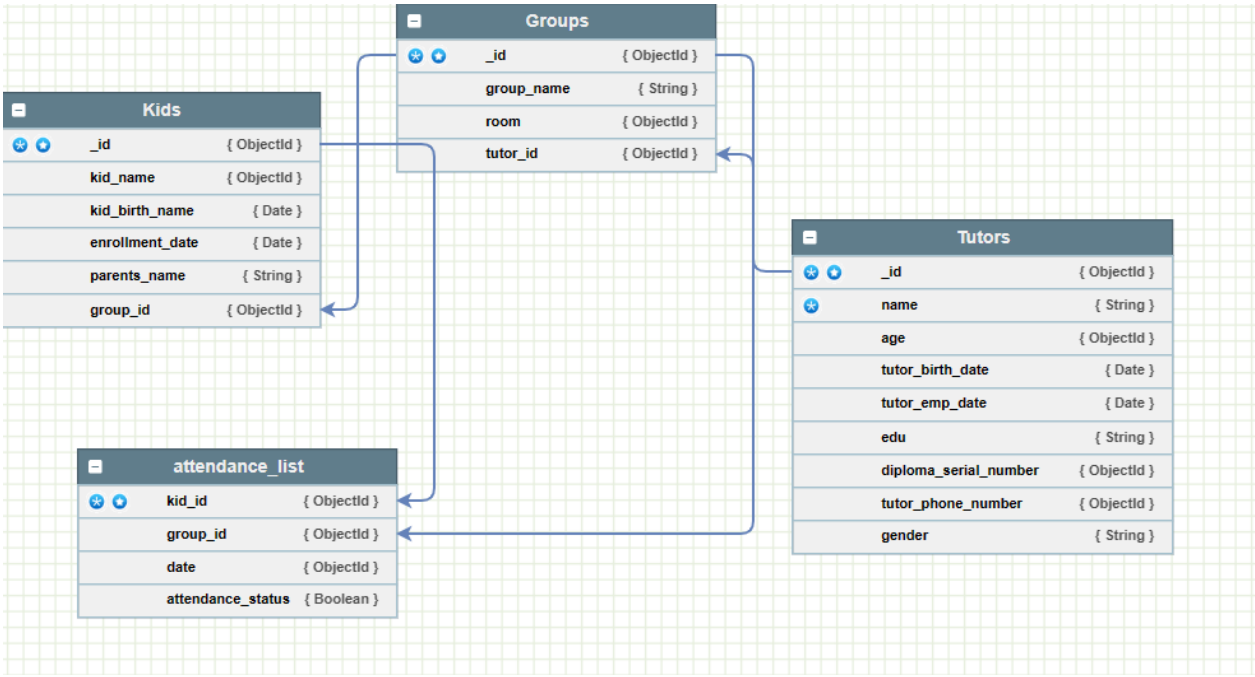
группы УБ-01

Лазарева Маргарита Вячеславовна

Воронеж – 2023

Цель работы: изучить операторы sql для работы с базой данных.

Модель данных для БД «детский сад»:



Для изучения операторов сначала подключимся к нашей базе данных.

```
postgres=# \c kindergarten
Вы подключены к базе данных "kindergarten" как пользователь "postgres".
kindergarten=#
```

Заполним таблицы данными:

```
kindergarten=# select * from tutors;
 tutor_id |                tutor_name                | tutor_age | birth_date | employment_date |
          diploma                          | diploma_serial | tutor_phone_number | gender
----------+-----------------------------------------+-----------+------------+-----------------+-----------------------------------------------------------------------------------
----------+----------------+--------------------+--------
       21 | Christen Stell                          |        27 | 1995-09-20 | 2020-01-20      | Bachelor in International Teacher Education for Prima
ry Schools (ITEPS) |         136789 | 89513467890        | woman
       23 | Dove Eyre                               |        27 | 1995-12-19 | 2018-08-01      | Bachelor in Intercultural Teacher Education
                   |         789061 | 89004562879        | woman
       24 | Olivia Rodgers                          |        31 | 1991-01-01 | 2019-11-30      | Bachelor in International Teacher Education for Prima
ry Schools (ITEPS) |         834567 | 89003458976        | woman
       25 | Candy Watt                              |        30 | 1992-05-24 | 2018-12-02      | Bachelor's Degree in Early Childhood Education Teache
r (Lugo Campus)    |         786123 | 89564873021        | woman
       26 | Ima Christen                            |        33 | 1989-02-12 | 2020-04-30      | Bachelor of Science in Science Education
                   |         675490 | 89456782345        | woman
       27 | Lame Starck                             |        28 | 1994-01-28 | 2022-05-26      | Bachelor's Degree in Early Childhood Education Teache
r (Lugo Campus)    |         458767 | 89503457891        | man
       22 | Daniel Michell                          |        30 | 1992-03-04 | 2019-09-30      | Bachelor's Degree in Early Childhood Education Teache
r (Lugo Campus)    |         786520 | 89203456712        | man
(7 строк)
```

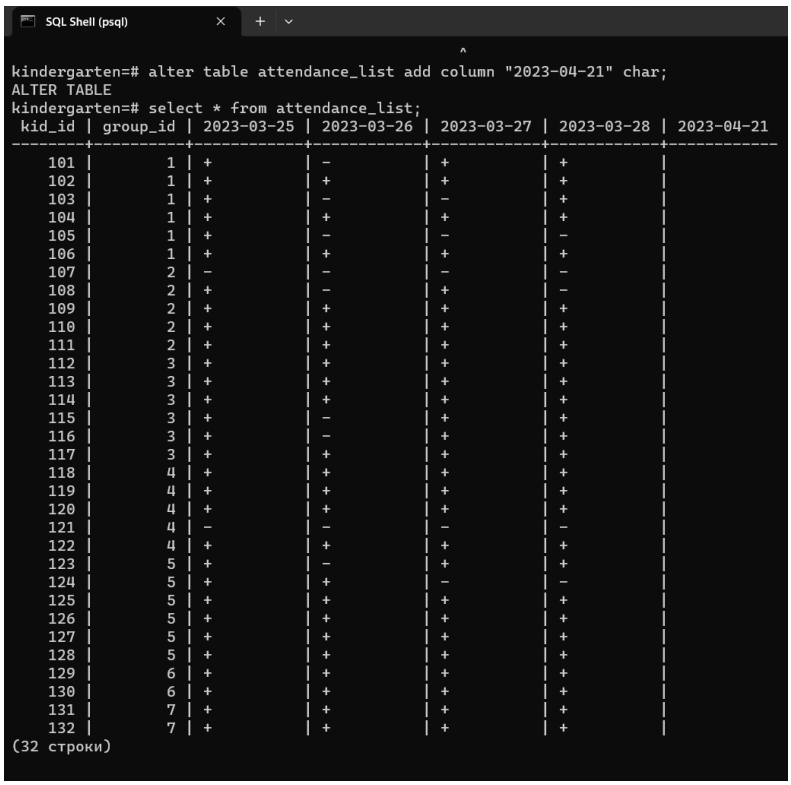
```
kindergarten=# select * from groups;
 group_id |                group_name                | room | tutor_id --
        1 | Flower                                  |  201 |       21
        2 | Sun                                     |  220 |       22
        3 | FlowerBloom                             |  247 |       23
        4 | Roses                                   |  289 |       24
        5 | SunFlower                               |  213 |       25
        6 | Friends                                 |  255 |       26
        7 | Fishers                                 |  291 |       27
(7 строк)
```

```
kindergarten=# select * from kids;
 kid_id |                kid_name                | group_id | kid_birth_date | enrollment_date |    hometown    | full_age
--------+----------------------------------------+----------+----------------+-----------------+----------------+----------
    101 | Sasha Hilton                           |        1 | 2022-02-23     | 2023-03-25      | Alexander City |        1
    102 | Nate Johnson                           |        1 | 2022-03-07     | 2023-03-12      | Andalusia      |        1
    103 | Eva Dwon                               |        1 | 2022-02-08     | 2023-03-10      | Anniston       |        1
    104 | Cristal Stewart                        |        1 | 2022-01-12     | 2023-02-12      | Alexander City |        1
    105 | Paris Brown                            |        1 | 2022-01-15     | 2023-01-16      | Alexander City |        1
    106 | Austin Kit                             |        1 | 2022-01-04     | 2023-01-16      | Bessemer       |        1
    107 | Danielle White                         |        2 | 2021-03-03     | 2022-02-01      | Auburn         |        2
    108 | Josh Dickenson                         |        2 | 2021-09-09     | 2022-02-02      | Montgomery     |        1
    109 | Rita Evans                             |        2 | 2021-05-21     | 2023-03-05      | Montgomery     |        1
    110 | Wendy Star                             |        2 | 2021-09-30     | 2022-06-04      | Bessemer       |        1
    111 | Justin Smith                           |        2 | 2021-02-28     | 2022-03-01      | Montgomery     |        2
    112 | Tanya Osten                            |        3 | 2020-03-10     | 2021-02-01      | Bessemer       |        3
    113 | Yeri Pinnet                            |        3 | 2020-09-04     | 2021-02-15      | Auburn         |        2
    114 | Will Suppet                            |        3 | 2020-05-13     | 2021-03-06      | Jasper         |        2
    115 | Jonathon Eyre                          |        3 | 2020-09-30     | 2022-08-01      | Jasper         |        2
    116 | Dana Willton                           |        3 | 2020-02-22     | 2022-08-01      | Clanton        |        3
    117 | Rod Wine                               |        3 | 2020-01-30     | 2021-02-02      | Auburn         |        3
    118 | Sam Osten                              |        4 | 2019-03-12     | 2020-04-12      | Montgomery     |        4
    119 | Kate Blake                             |        4 | 2019-01-13     | 2020-02-12      | Clanton        |        4
    120 | Roxanna Dastin                         |        4 | 2019-08-30     | 2020-09-12      | Montgomery     |        3
    122 | Nina Flores                            |        4 | 2019-02-27     | 2020-02-28      | Montgomery     |        4
    123 | Nina Adams                             |        5 | 2018-03-23     | 2022-03-24      | Huntsville     |        5
    124 | Linda Scott                            |        5 | 2018-01-25     | 2019-02-20      | Montgomery     |        5
    125 | Sarah Green                            |        5 | 2018-02-24     | 2019-09-29      | Montgomery     |        5
    126 | Scott Nelson                           |        5 | 2018-01-30     | 2021-08-25      | Huntsville     |        5
    127 | Jane Rivera                            |        5 | 2018-04-26     | 2020-04-05      | Montgomery     |        4
    128 | Mike Mitchell                          |        5 | 2018-03-09     | 2019-04-06      | Montgomery     |        5
    129 | Bob King                               |        6 | 2017-05-14     | 2019-10-15      | Montgomery     |        5
    130 | Janett Torres                          |        6 | 2016-12-16     | 2019-10-16      | Montgomery     |        5
    131 | Peter Hall                             |        7 | 2016-08-18     | 2017-08-31      | Huntsville     |        6
    132 | Nelly Wright                           |        7 | 2016-09-17     | 2018-02-26      | Auburn         |        6
    121 | Lily Valley                            |        4 | 2019-09-20     | 2020-09-21      | Montgomery     |        3
(32 строки)
```

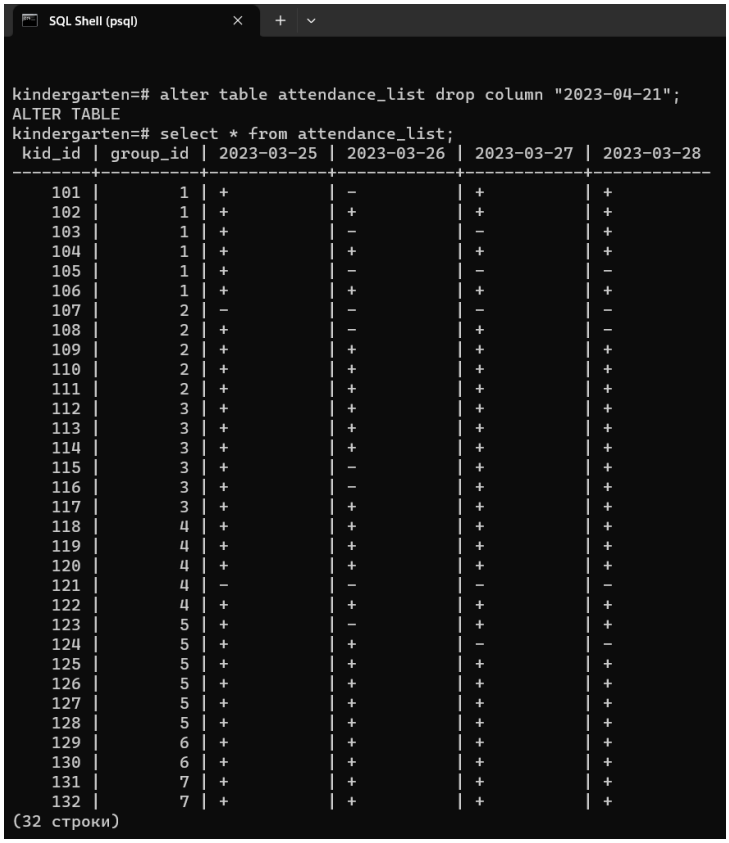
```
kindergarten=# INSERT INTO attendance_list VALUES (132, 7, '+', '+', '+', '
INSERT 0 1
kindergarten=# select * from attendance_list;
 kid_id | group_id | 2023-03-25 | 2023-03-26 | 2023-03-27 | 2023-03-28
--------+----------+------------+------------+------------+------------
    101 |        1 | +          | -          | +          | +
    102 |        1 | +          | +          | +          | +
    103 |        1 | +          | -          | -          | +
    104 |        1 | +          | +          | +          | +
    105 |        1 | +          | -          | -          | -
    106 |        1 | +          | +          | +          | +
    107 |        2 | -          | -          | -          | -
    108 |        2 | +          | -          | +          | -
    109 |        2 | +          | +          | +          | +
    110 |        2 | +          | +          | +          | +
    111 |        2 | +          | +          | +          | +
    112 |        3 | +          | +          | +          | +
    113 |        3 | +          | +          | +          | +
    114 |        3 | +          | +          | +          | +
    115 |        3 | +          | -          | +          | +
    116 |        3 | +          | -          | +          | +
    117 |        3 | +          | +          | +          | +
    118 |        4 | +          | +          | +          | +
    119 |        4 | +          | +          | +          | +
    120 |        4 | +          | +          | +          | +
    121 |        4 | -          | -          | -          | -
    122 |        4 | +          | +          | +          | +
    123 |        5 | +          | -          | +          | +
    124 |        5 | +          | +          | -          | -
    125 |        5 | +          | +          | +          | +
    126 |        5 | +          | +          | +          | +
    127 |        5 | +          | +          | +          | +
    128 |        5 | +          | +          | +          | +
    129 |        6 | +          | +          | +          | +
    130 |        6 | +          | +          | +          | +
    131 |        7 | +          | +          | +          | +
    132 |        7 | +          | +          | +          | +
(32 строки)
```

1. Оператор ALTER TABLE

Он используется для изменения таблицы. Добавим столбец в любую таблицу :

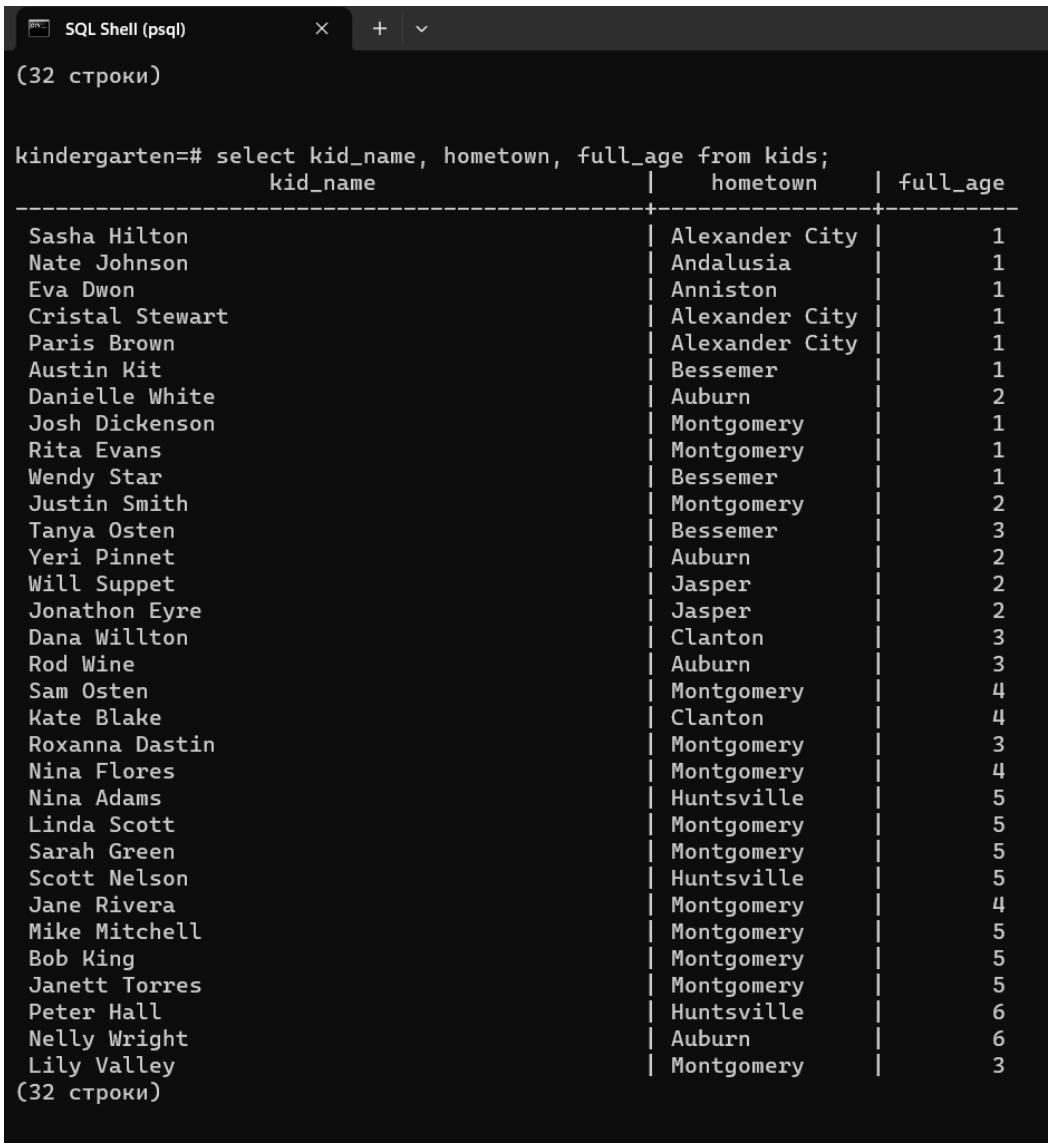


И удалим его:



2. Чтение заданных столбцов из одиночной таблицы

Следующий запрос произведет выборку (чтение) трех из семи столбцов таблицы kids: select **kid_name**, **hometown**, **full_age** from kids;



Порядок столбцов в результирующей таблице определяется порядком следования их имен после ключевого слова SELECT. Изменим порядок:

select **hometown**, **full_age, kid_name** from kids;

```
SQL Shell (psql)
(32 строки)


kindergarten=# select hometown, full_age, kid_name from kids;
    hometown    | full_age |                kid_name
----------------+----------+---------------------------------
 Alexander City |        1 | Sasha Hilton
 Andalusia      |        1 | Nate Johnson
 Anniston       |        1 | Eva Dwon
 Alexander City |        1 | Cristal Stewart
 Alexander City |        1 | Paris Brown
 Bessemer       |        1 | Austin Kit
 Auburn         |        2 | Danielle White
 Montgomery     |        1 | Josh Dickenson
 Montgomery     |        1 | Rita Evans
 Bessemer       |        1 | Wendy Star
 Montgomery     |        2 | Justin Smith
 Bessemer       |        3 | Tanya Osten
 Auburn         |        2 | Yeri Pinnet
 Jasper         |        2 | Will Suppet
 Jasper         |        2 | Jonathon Eyre
 Clanton        |        3 | Dana Willton
 Auburn         |        3 | Rod Wine
 Montgomery     |        4 | Sam Osten
 Clanton        |        4 | Kate Blake
 Montgomery     |        3 | Roxanna Dastin
 Montgomery     |        4 | Nina Flores
 Huntsville     |        5 | Nina Adams
 Montgomery     |        5 | Linda Scott
 Montgomery     |        5 | Sarah Green
 Huntsville     |        5 | Scott Nelson
 Montgomery     |        4 | Jane Rivera
 Montgomery     |        5 | Mike Mitchell
 Montgomery     |        5 | Bob King
 Montgomery     |        5 | Janett Torres
 Huntsville     |        6 | Peter Hall
 Auburn         |        6 | Nelly Wright
 Montgomery     |        3 | Lily Valley
(32 строки)
```

Следующий запрос извлекает из таблицы kids только столбец full_age:

select **full_age** from kids;

```
SQL Shell (psql)

kindergarten=# select full_age from kids;
 full_age
----------
        1
        1
        1
        1
        1
        1
        2
        1
        1
        1
        2
        3
        2
        2
        2
        3
        3
        4
        4
        3
        4
        5
        5
        5
        5
        5
        4
        5
        5
        5
        6
        6
        3
(32 строки)


kindergarten=#
```

Следует обратить внимание, что в таблице есть одинаковые строки. Согласно определению отношения, повторения строк в отношении недопустимо. Однако процесс поиска и удаления таких повторений отнимает много времени. Таким образом, на практике все же приходится сталкиваться с одинаковыми строками.

Если нужно, чтобы СУБД нашла и удалила повторяющиеся строки, при запросе необходимо использовать ключевое слово DISTINCT:

select distinct full_age from kids;

```
kindergarten=# select distinct full_age from kids;
 full_age
----------
         3
         5
         4
         6
         2
         1
(6 строк)
```

3. Чтение заданных строк из одиночной таблицы

Ранее рассмотренные SQL-запросы выбирали определенные столбцы всех строк таблицы. Теперь рассмотрим запросы, позволяющие выбирать столбцы определенных строк.

Будем работать с двумя таблицами (kids, tutors).

Следующий запрос получает все столбцы из тех строк таблицы kids, которые содержат сведения о детях, родившихся в Монтгомери, столице Алабамы:

select kid_id, kid_name, kid_birth_date, enrollment_date, full_age from kids where hometown = ‘Montgomery’;

```
kindergarten=# select kid_id, kid_name, kid_birth_date, enrollment_date, full_age from kids where hometown = 'Montgomery';
 kid_id |                  kid_name                  | kid_birth_date | enrollment_date | full_age
--------+--------------------------------------------+----------------+-----------------+----------
    108 | Josh Dickenson                             | 2021-09-09     | 2022-02-02      |        1
    109 | Rita Evans                                 | 2021-05-21     | 2023-03-05      |        1
    111 | Justin Smith                               | 2021-02-28     | 2022-03-01      |        2
    118 | Sam Osten                                  | 2019-03-12     | 2020-04-12      |        4
    120 | Roxanna Dastin                             | 2019-08-30     | 2020-09-12      |        3
    122 | Nina Flores                                | 2019-02-27     | 2020-02-28      |        4
    124 | Linda Scott                                | 2018-01-25     | 2019-02-20      |        5
    125 | Sarah Green                                | 2018-02-24     | 2019-09-29      |        5
    127 | Jane Rivera                                | 2018-04-26     | 2020-04-05      |        4
    128 | Mike Mitchell                              | 2018-03-09     | 2019-04-06      |        5
    129 | Bob King                                   | 2017-05-14     | 2019-10-15      |        5
    130 | Janett Torres                              | 2016-12-16     | 2019-10-16      |        5
    121 | Lily Valley                                | 2019-09-20     | 2020-09-21      |        3
(13 строк)


kindergarten=#
```

select hometown, kid_id, kid_name, kid_birth_date, enrollment_date, full_age from kids where hometown = 'Montgomery'; - данный запрос выводит все столбцы таблицы в указанном нами порядке:

```
kindergarten=# select hometown, kid_id, kid_name, kid_birth_date, enrollment_date, full_age from kids where hometown = 'Montgomery';
  hometown  | kid_id |                 kid_name                 | kid_birth_date | enrollment_date | full_age
------------+--------+------------------------------------------+----------------+-----------------+----------
 Montgomery |    108 | Josh Dickenson                           | 2021-09-09     | 2022-02-02      |        1
 Montgomery |    109 | Rita Evans                               | 2021-05-21     | 2023-03-05      |        1
 Montgomery |    111 | Justin Smith                             | 2021-02-28     | 2022-03-01      |        2
 Montgomery |    118 | Sam Osten                                | 2019-03-12     | 2020-04-12      |        4
 Montgomery |    120 | Roxanna Dastin                           | 2019-08-30     | 2020-09-12      |        3
 Montgomery |    122 | Nina Flores                              | 2019-02-27     | 2020-02-28      |        4
 Montgomery |    124 | Linda Scott                              | 2018-01-25     | 2019-02-20      |        5
 Montgomery |    125 | Sarah Green                              | 2018-02-24     | 2019-09-29      |        5
 Montgomery |    127 | Jane Rivera                              | 2018-04-26     | 2020-04-05      |        4
 Montgomery |    128 | Mike Mitchell                            | 2018-03-09     | 2019-04-06      |        5
 Montgomery |    129 | Bob King                                 | 2017-05-14     | 2019-10-15      |        5
 Montgomery |    130 | Janett Torres                            | 2016-12-16     | 2019-10-16      |        5
 Montgomery |    121 | Lily Valley                              | 2019-09-20     | 2020-09-21      |        3
(13 строк)


kindergarten=#
```

Второй способ запросить все столбцы таблицы – использовать специальный символ ‘*’ после ключевого слова SELECT:

select * from kids where hometown = ‘Montgomery’;

```
kindergarten=# select * from kids where hometown = 'Montgomery';
 kid_id |                 kid_name                 | group_id | kid_birth_date | enrollment_date |  hometown  | full_age
--------+------------------------------------------+----------+----------------+-----------------+------------+----------
    108 | Josh Dickenson                           |        2 | 2021-09-09     | 2022-02-02      | Montgomery |        1
    109 | Rita Evans                               |        2 | 2021-05-21     | 2023-03-05      | Montgomery |        1
    111 | Justin Smith                             |        2 | 2021-02-28     | 2022-03-01      | Montgomery |        2
    118 | Sam Osten                                |        4 | 2019-03-12     | 2020-04-12      | Montgomery |        4
    120 | Roxanna Dastin                           |        4 | 2019-08-30     | 2020-09-12      | Montgomery |        3
    122 | Nina Flores                              |        4 | 2019-02-27     | 2020-02-28      | Montgomery |        4
    124 | Linda Scott                              |        5 | 2018-01-25     | 2019-02-20      | Montgomery |        5
    125 | Sarah Green                              |        5 | 2018-02-24     | 2019-09-29      | Montgomery |        5
    127 | Jane Rivera                              |        5 | 2018-04-26     | 2020-04-05      | Montgomery |        4
    128 | Mike Mitchell                            |        5 | 2018-03-09     | 2019-04-06      | Montgomery |        5
    129 | Bob King                                 |        6 | 2017-05-14     | 2019-10-15      | Montgomery |        5
    130 | Janett Torres                            |        6 | 2016-12-16     | 2019-10-16      | Montgomery |        5
    121 | Lily Valley                              |        4 | 2019-09-20     | 2020-09-21      | Montgomery |        3
(13 строк)


kindergarten=#
```

Этот запрос получает все столбцы из тех строк таблицы tutors, которые содержат сведения о тьюторках (воспитательницах):

select tutor_id, tutor_name, tutor_age from tutors where gender = ‘woman’;

```
kindergarten=# select tutor_id, tutor_name, tutor_age from tutors where gender = 'woman';
 tutor_id |                tutor_name                | tutor_age
----------+------------------------------------------+-----------
       21 | Christen Stell                           |        27
       23 | Dove Eyre                                |        27
       24 | Olivia Rodgers                           |        31
       25 | Candy Watt                               |        30
       26 | Ima Christen                             |        33
(5 строк)


kindergarten=#
```

Следующий запрос извлекает столбцы с именем, датой рождения, датой начала работы в детском саду (приёма на работу), дипломом и возрастом тьюторов из тех строк таблицы tutors, где значение столбца tutor_age больше 30:

select tutor_name, birth_date, tutor_age, diploma, employment_date from tutors where tutor_age > 30;

```
kindergarten=# select tutor_name, birth_date, tutor_age, diploma, employment_date from tutors where tutor_age > 30;
              tutor_name                | birth_date | tutor_age |                                 diploma                                 | employme
nt_date
----------------------------------------+------------+-----------+-------------------------------------------------------------------------+---------
--------
 Olivia Rodgers                         | 1991-01-01 |        31 | Bachelor in International Teacher Education for Primary Schools (ITEPS) | 2019-11-
30
 Ima Christen                           | 1989-02-12 |        33 | Bachelor of Science in Science Education                                | 2020-04-
30
(2 строки)


kindergarten=#
```

В предложении where можно указать более одного условия, если использовать ключевое слово and:

select * from tutors where gender = 'woman' and tutor_age >= 30;

```
kindergarten=# select * from tutors where gender = 'woman' and tutor_age >= 30;
 tutor_id |              tutor_name                | tutor_age | birth_date | employment_date |                                 diploma
                | diploma_serial | tutor_phone_number | gender
----------+----------------------------------------+-----------+------------+-----------------+-------------------------------------------------------------------------
----------------+----------------+--------------------+--------
       24 | Olivia Rodgers                         |        31 | 1991-01-01 | 2019-11-30      | Bachelor in International Teacher Education for Prima
ry Schools (ITEPS) |         834567 | 89003458976        | woman
       25 | Candy Watt                             |        30 | 1992-05-24 | 2018-12-02      | Bachelor's Degree in Early Childhood Education Teache
r (Lugo Campus)    |         786123 | 89564873021        | woman
       26 | Ima Christen                           |        33 | 1989-02-12 | 2020-04-30      | Bachelor of Science in Science Education
                   |         675490 | 89456782345        | woman
(3 строки)
```

4. Чтение заданных строк и столбцов из одиночной таблицы

Объединив описанные выше методы, можно выбирать из таблицы определенные столбцы и определенные строки. Следующий запрос извлекает из таблицы kids столбцы kid_name и full_age имена детей возраста 1-3 года:

select kid_name, full_age from kids where full_age < 4;

```
kindergarten=# select kid_name, full_age from kids where full_age < 4;
                kid_name                | full_age
----------------------------------------+----------
 Sasha Hilton                           |        1
 Nate Johnson                           |        1
 Eva Dwon                               |        1
 Cristal Stewart                        |        1
 Paris Brown                            |        1
 Austin Kit                             |        1
 Danielle White                         |        2
 Josh Dickenson                         |        1
 Rita Evans                             |        1
 Wendy Star                             |        1
 Justin Smith                           |        2
 Tanya Osten                            |        3
 Yeri Pinnet                            |        2
 Will Suppet                            |        2
 Jonathon Eyre                          |        2
 Dana Willton                           |        3
 Rod Wine                               |        3
 Roxanna Dastin                         |        3
 Lily Valley                            |        3
(19 строк)


kindergarten=#
```

Еще одна форма предложения where предполагает задания списка значений, которые может иметь столбец. Это делается с помощью ключевого слова in:

select kid_name, full_age from kids where full_age in (‘1’, ’2’, ‘3’);

```
kindergarten=# select kid_name, full_age from kids where full_age in ('1', '2', '3');
                kid_name                  | full_age
------------------------------------------+----------
 Sasha Hilton                             |        1
 Nate Johnson                             |        1
 Eva Dwon                                 |        1
 Cristal Stewart                          |        1
 Paris Brown                              |        1
 Austin Kit                               |        1
 Danielle White                           |        2
 Josh Dickenson                           |        1
 Rita Evans                               |        1
 Wendy Star                               |        1
 Justin Smith                             |        2
 Tanya Osten                              |        3
 Yeri Pinnet                              |        2
 Will Suppet                              |        2
 Jonathon Eyre                            |        2
 Dana Willton                             |        3
 Rod Wine                                 |        3
 Roxanna Dastin                           |        3
 Lily Valley                              |        3
(19 строк)


kindergarten=#
```

Чтобы выбрать строки, у которых столбец full_age не равен ни одному из этих значений, воспользуемся ключевым словом not in:

select kid_name, full_age from kids where full_age not in (’1’, ’2’, ‘3’);

```
kindergarten=# select kid_name, full_age from kids where full_age not in ('1', '2', '3');
                kid_name                  | full_age
------------------------------------------+----------
 Sam Osten                                |        4
 Kate Blake                               |        4
 Nina Flores                              |        4
 Nina Adams                               |        5
 Linda Scott                              |        5
 Sarah Green                              |        5
 Scott Nelson                             |        5
 Jane Rivera                              |        4
 Mike Mitchell                            |        5
 Bob King                                 |        5
 Janett Torres                            |        5
 Peter Hall                               |        6
 Nelly Wright                             |        6
(13 строк)


kindergarten=#
```

5. Диапазоны, специальные символы и пустые значения в предложениях WHERE

В предложениях where могут также указываться диапазоны и шаблоны поиска. Для задания диапазонов используется ключевое слово between.

Например, запрос select group_name from groups where room between 240 and 300;

```
kindergarten=# select group_name from groups where room between 240 and 300;
                  group_name
-----------------------------------------------
 FlowerBloom
 Roses
 Friends
 Fishers
(4 строки)


kindergarten=#
```

Рассмотренный запрос эквивалентен следующему:

select group_name from groups where room >= 240 and room <= 300;

```
kindergarten=# select group_name from groups where room >= 240 and room <= 300;
                  group_name
-----------------------------------------------
 FlowerBloom
 Roses
 Friends
 Fishers
(4 строки)


kindergarten=#
```

Таким образом, диапазон, задаваемый ключевым словом between, включает в себя граничные значения (в данном случае 240 и 300).

Для задания шаблонов поиска в SQL используется ключевое слово LIKE.

Запросим две таблицы, в первой из которых будет отображаться информация о детях, родившихся в городах, названия которых начинаются на букву «A», а во второй – о группах, имена которых начинаются с буквы «F»:

select * from kids where hometown like ’A%’;

select * from groups where group_name like ‘F%’;

```
kindergarten=# select * from kids where hometown like 'A%';
 kid_id |                kid_name                | group_id | kid_birth_date | enrollment_date |    hometown    | full_age
--------+----------------------------------------+----------+----------------+-----------------+----------------+----------
    101 | Sasha Hilton                           |        1 | 2022-02-23     | 2023-03-25      | Alexander City |        1
    102 | Nate Johnson                           |        1 | 2022-03-07     | 2023-03-12      | Andalusia      |        1
    103 | Eva Dwon                               |        1 | 2022-02-08     | 2023-03-10      | Anniston       |        1
    104 | Cristal Stewart                        |        1 | 2022-01-12     | 2023-02-12      | Alexander City |        1
    105 | Paris Brown                            |        1 | 2022-01-15     | 2023-01-16      | Alexander City |        1
    107 | Danielle White                         |        2 | 2021-03-03     | 2022-02-01      | Auburn         |        2
    113 | Yeri Pinnet                            |        3 | 2020-09-04     | 2021-02-15      | Auburn         |        2
    117 | Rod Wine                               |        3 | 2020-01-30     | 2021-02-02      | Auburn         |        3
    132 | Nelly Wright                           |        7 | 2016-09-17     | 2018-02-26      | Auburn         |        6
(9 строк)


kindergarten=# select * from groups where group_name like 'F%';
 group_id |               group_name               | room | tutor_id
----------+----------------------------------------+------+----------
        1 | Flower                                 |  201 |       21
        3 | FlowerBloom                            |  247 |       23
        6 | Friends                                |  255 |       26
        7 | Fishers                                |  291 |       27
(4 строки)


kindergarten=#
```

Если требуется найти всех детей, названия родных городов которых заканчиваются на букву ’n’, можно использовать символ процента следующим образом: select * from kids where hometown like ’%n’;

```
kindergarten=# select * from kids where hometown like '%n';
 kid_id |                kid_name                | group_id | kid_birth_date | enrollment_date | hometown | full_age
--------+----------------------------------------+----------+----------------+-----------------+----------+----------
    103 | Eva Dwon                               |        1 | 2022-02-08     | 2023-03-10      | Anniston |        1
    107 | Danielle White                         |        2 | 2021-03-03     | 2022-02-01      | Auburn   |        2
    113 | Yeri Pinnet                            |        3 | 2020-09-04     | 2021-02-15      | Auburn   |        2
    116 | Dana Willton                           |        3 | 2020-02-22     | 2022-08-01      | Clanton  |        3
    117 | Rod Wine                               |        3 | 2020-01-30     | 2021-02-02      | Auburn   |        3
    119 | Kate Blake                             |        4 | 2019-01-13     | 2020-02-12      | Clanton  |        4
    132 | Nelly Wright                           |        7 | 2016-09-17     | 2018-02-26      | Auburn   |        6
(7 строк)


kindergarten=#
```

6. Сортировка результатов

Порядок строк в таблице, возвращаемой оператором select, является произвольным. Если нужно отсортировать строки результата, это можно сделать с помощью конструкции ORDER BY. Например, следующий запрос возвращает имена и родные города детей, отсортированных в алфавитном порядке по городам:

select kid_name, hometown from kids order by hometown;

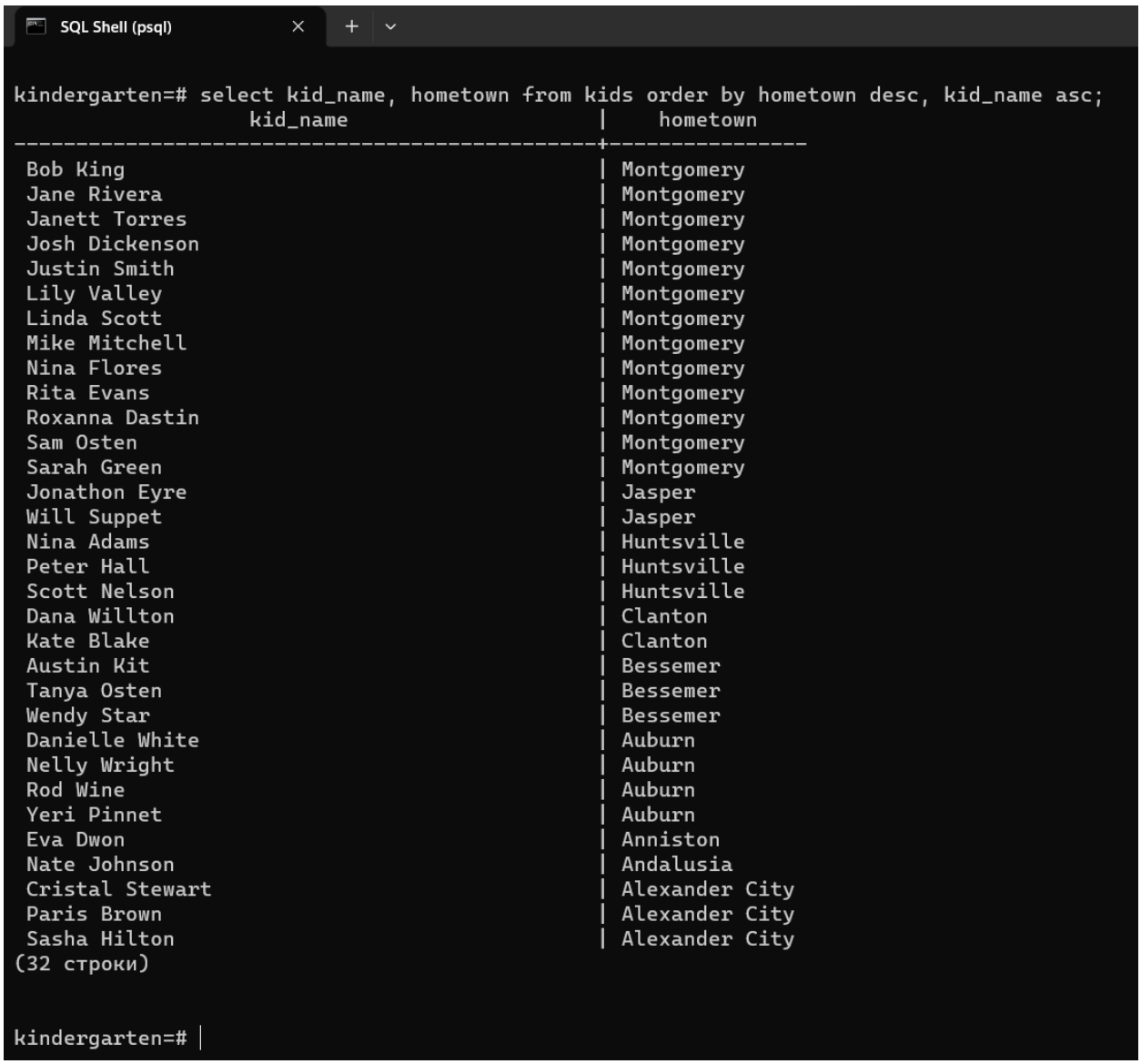
```
SQL Shell (psql)
                                        ^
kindergarten=# select kid_name, hometown from kids order by hometown;
                kid_name                |    hometown
----------------------------------------+----------------
 Sasha Hilton                           | Alexander City
 Cristal Stewart                        | Alexander City
 Paris Brown                            | Alexander City
 Nate Johnson                           | Andalusia
 Eva Dwon                               | Anniston
 Rod Wine                               | Auburn
 Danielle White                         | Auburn
 Yeri Pinnet                            | Auburn
 Nelly Wright                           | Auburn
 Tanya Osten                            | Bessemer
 Austin Kit                             | Bessemer
 Wendy Star                             | Bessemer
 Dana Willton                           | Clanton
 Kate Blake                             | Clanton
 Peter Hall                             | Huntsville
 Nina Adams                             | Huntsville
 Scott Nelson                           | Huntsville
 Will Suppet                            | Jasper
 Jonathon Eyre                          | Jasper
 Bob King                               | Montgomery
 Janett Torres                          | Montgomery
 Josh Dickenson                         | Montgomery
 Rita Evans                             | Montgomery
 Justin Smith                           | Montgomery
 Lily Valley                            | Montgomery
 Sam Osten                              | Montgomery
 Roxanna Dastin                         | Montgomery
 Nina Flores                            | Montgomery
 Linda Scott                            | Montgomery
 Sarah Green                            | Montgomery
 Jane Rivera                            | Montgomery
 Mike Mitchell                          | Montgomery
(32 строки)


kindergarten=#
```

По умолчанию сортировка в SQL производится в порядке возрастания. Для явного указания порядка сортировки можно использовать ключевые слова ASC (по возрастанию) и DESC (по убыванию). Например, следующий запрос отсортирует города по их названию в обратном порядке:

select kid_name, hometown from kids order by hometown desc;

Сортировать можно и более чем по одному столбцу. Чтобы отсортировать список имен художников и их национальностей сначала по национальностям в обратном порядке, а потом внутри каждой национальности по именам в прямом порядке, можно было бы использовать такой запрос:

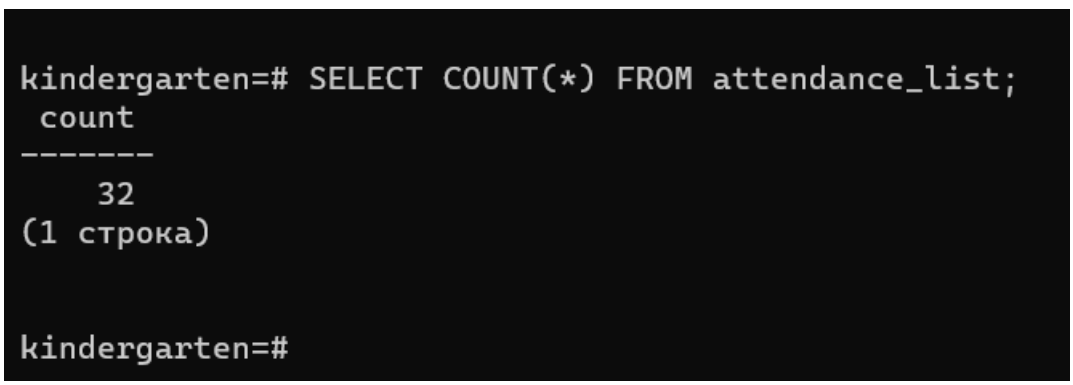
select kid_name, hometown from kids order by hometown desc, kid_name asc;



7. Агрегатные функции SQL

В SQL имеется пять агрегатных функций: COUNT, SUM, AVG, MAX и MIN. Они выполняют различные действия над результатами оператора SELECT. Функция COUNT работает вне зависимости от типа данных столбца, а функции SUM, AVG, MAX и MIN оперируют только числовыми столбцами (integer, numeric и т. д.).

Функция COUNT подсчитывает количество строк в результате, а функция SUM вычисляет сумму значений числового столбца. Например, следующий SQL-запрос подсчитывает количество строк в таблице attendance_list: SELECT COUNT(*) FROM attendance_list;



Как уже говорилось ранее, результатом SQL-оператора SELECT всегда является отношение. Если, как в данном случае, результат представляет собой одиночное число, это число все равно считается отношением, имеющим одну строку и один столбец.

Рассмотрим следующие два запроса: SELECT COUNT(2023-03-25) FROM attendance_list; и SELECT COUNT(DISTINCT 2023-03-25) FROM attendance_list;

```
kindergarten=# SELECT COUNT(2023-03-25) FROM attendance_list;
 count
-------
    32
(1 строка)


kindergarten=# SELECT COUNT(DISTINCT 2023-03-25) FROM attendance_list;
 count
-------
     1
(1 строка)


kindergarten=#
```

Разница в результатах возникает потому, что второй оператор SELECT не учитывает повторяющиеся строки.

Вот еще один пример использования агрегатных функций:

SELECT MIN(tutor_age), MAX(tutor_age), SUM(tutor_age) FROM tutors WHERE tutor_id < 24;

```
kindergarten=# SELECT MIN(tutor_age), MAX(tutor_age), SUM(tutor_age) FROM tutors WHERE tutor_id < 24;
 min | max | sum
-----+-----+-----
  27 |  30 |  84
(1 строка)


kindergarten=#
```

8. Агрегатные функции и группировка

Полезность агрегатных функций увеличивает тот факт, что их можно применять к группам строк данных. Например, следующий оператор подсчитывает количество детей, родившихся в каждом городе Алабамы: SELECT hometown, COUNT(*) FROM kids GROUP BY hometown;

```
kindergarten=# SELECT hometown, COUNT(*) FROM kids GROUP BY hometown;
    hometown    | count
----------------+-------
 Auburn         |     4
 Bessemer       |     3
 Andalusia      |     1
 Jasper         |     2
 Clanton        |     2
 Huntsville     |     3
 Anniston       |     1
 Alexander City |     3
 Montgomery     |    13
(9 строк)
```

Этот же подсчитывает количество + и -, то есть сколько 26.03.2023 присутствовало и отсутствовало детей:

SELECT 2023-03-26, COUNT(*) FROM attendance_list GROUP BY 2023-03-26;

```
kindergarten=# SELECT "2023-03-26", COUNT(*) FROM attendance_list GROUP BY "2023-03-26";
 2023-03-26 | count
------------+-------
 +          |    23
 -          |     9
(2 строки)


kindergarten=#
```

Далее ограничить множество выдаваемых результатов можно, применяя к формируемым группам различные условия. Например, если нас интересуют только те группы, в которых имеется более четырёх записей, мы могли бы написать следующее: SELECT hometown, COUNT(*) FROM kids GROUP BY hometown HAVING COUNT(*) > 4;

```
kindergarten=# SELECT hometown, COUNT(*) FROM kids GROUP BY hometown HAVING COUNT(*) > 4;
  hometown  | count
------------+-------
 Montgomery |    13
(1 строка)


kindergarten=#
```

Вместе с ключевым словом GROUP BY можно использовать и предложение WHERE. Однако здесь имеет место неоднозначность. Если условие в предложении WHERE применяется до формирования групп, результат будет иным, чем когда это условие применяется к уже сформированным группам. Для устранения этой неоднозначности стандарт SQL устанавливает, что в случаях, когда предложения WHERE и GROUP BY используются одновременно, первым должно применяться условие, записанное в предложении WHERE. Рассмотрим, например, следующий оператор: SELECT hometown, COUNT(*) FROM kids WHERE kid_id >106 GROUP BY hometown HAVING COUNT(*) >= 1;

```
kindergarten=# SELECT hometown, COUNT(*) FROM kids WHERE kid_id >106 GROUP BY hometown HAVING COUNT(*) >= 1;
  hometown  | count
------------+-------
 Auburn     |     4
 Bessemer   |     2
 Montgomery |    13
 Jasper     |     2
 Clanton    |     2
 Huntsville |     3
(6 строк)


kindergarten=#
```

При выполнении данного оператора сначала применяется условие из предложения WHERE, которое отбирает детей, чей идентификатор больше 106 (т.е. старше года). Затем, после формирования групп, применяется условие из предложения HAVING.

9. Оконные функции

Использование оконных функций – второй из двух случаев, когда имя столбца может появляться вместе с агрегатными функциями.

При использовании группировки результирующая таблица содержит по одной строке для каждого группируемого значения, а оконная функция не свертывает результаты, принадлежащие одной группе.

Следующий запрос использует оператор GROUP BY: select group_name, room, count(*) from groups group by group_name, room order by room;

```
kindergarten=# select group_name, room, count(*) from groups group by group_name, room order by room;
                group_name                | room | count
------------------------------------------+------+-------
 Flower                                   |  201 |     1
 SunFlower                                |  213 |     1
 Sun                                      |  220 |     1
 FlowerBloom                              |  247 |     1
 Friends                                  |  255 |     1
 Roses                                    |  289 |     1
 Fishers                                  |  291 |     1
(7 строк)


kindergarten=#
```

10. Чтение данных из нескольких таблиц с применением вложенных запросов

Все запросы, рассмотренные ранее, считывали данные из одиночной таблицы. Бывает, однако, что для получения требуемой информации необходимо обработать более одной таблицы. Предположим, например, что мы хотим знать имена детей, которые пришли в детский сад 28.03.23. Имена детей хранятся в таблице kids, а статус посещаемости в таблице attendance_list.

select kid_id from attendance_list where “2023-03-28” = ‘+’;

```
kindergarten=# select kid_id from attendance_list where "2023-03-28" = '+';
 kid_id
--------
    101
    102
    103
    104
    106
    109
    110
    111
    112
    113
    114
    115
    116
    117
    118
    119
    120
    122
    123
    125
    126
    127
    128
    129
    130
    131
    132
(27 строк)
```

Теперь мы можем объединить эти два SQL-запроса при помощи так называемого вложенного запроса (subquery):

select kid_name from kids where kid_id in (select kid_id from attendance_list where “2023-03-28” = ‘+’);

```
kindergarten=# select kid_name from kids where kid_id
                kid_name
-----------------------------------------------
 Sasha Hilton
 Nate Johnson
 Eva Dwon
 Cristal Stewart
 Austin Kit
 Rita Evans
 Wendy Star
 Justin Smith
 Tanya Osten
 Yeri Pinnet
 Will Suppet
 Jonathon Eyre
 Dana Willton
 Rod Wine
 Sam Osten
 Kate Blake
 Roxanna Dastin
 Nina Flores
 Nina Adams
 Sarah Green
 Scott Nelson
 Jane Rivera
 Mike Mitchell
 Bob King
 Janett Torres
 Peter Hall
 Nelly Wright
(27 строк)


kindergarten=#
```

11. Чтение данных из нескольких таблиц с помощью операции соединения

Вложенные запросы подходят для обработки нескольких таблиц до тех пор, пока результаты (столбцы в предложении SELECT) относятся к одной и той же таблице. Если же нам нужно извлечь данные из двух или более таблиц, при помощи вложенного запроса это сделать не удастся. Вместо этого необходимо использовать операцию соединения (JOIN). Основная идея здесь – создать новое отношение, связав между собой содержимое двух или более исходных отношений.

Рассмотрим следующий пример: SELECT kid_name, “2023-03-28” FROM kids, attendance_list WHERE kids.kid_id = attendance_list.kid_id;

```
kindergarten=# SELECT kid_name, "2023-03-28" FROM kids, attendance_list
                kid_name                 | 2023-03-28
-----------------------------------------+------------
 Sasha Hilton                            | +
 Nate Johnson                            | +
 Eva Dwon                                | +
 Cristal Stewart                         | +
 Paris Brown                             | -
 Austin Kit                              | +
 Danielle White                          | -
 Josh Dickenson                          | -
 Rita Evans                              | +
 Wendy Star                              | +
 Justin Smith                            | +
 Tanya Osten                             | +
 Yeri Pinnet                             | +
 Will Suppet                             | +
 Jonathon Eyre                           | +
 Dana Willton                            | +
 Rod Wine                                | +
 Sam Osten                               | +
 Kate Blake                              | +
 Roxanna Dastin                          | +
 Lily Valley                             | -
 Nina Flores                             | +
 Nina Adams                              | +
 Linda Scott                             | -
 Sarah Green                             | +
 Scott Nelson                            | +
 Jane Rivera                             | +
 Mike Mitchell                           | +
 Bob King                                | +
 Janett Torres                           | +
 Peter Hall                              | +
 Nelly Wright                            | +
(32 строки)
```

Смысл этого оператора заключается в том, что создается новая таблица с двумя столбцами kid_name и 2023-03-28. Эти столбцы берутся соответственно из таблиц kids и attendance_list при условии, что столбец kid_id в таблице kids равен одноименному столбцу в таблице attendance_list. Обозначения kids.kid_id и attendance_list.kid_id необходимы для устранения конфликта имен столбцов.

Соответственно, мы получили отношение, в котором указаны имена детей и статус посещаемости на 28.03.2023 (мини-журнал на одну дату).

Предложение WHERE мы также можем применить в процессе создания соединения:

SELECT kid_name, “2023-03-28” FROM kids, attendance_list WHERE kids.kid_id = attendance_list.kid_id and “2023-03-28” = ‘-‘;

```
kindergarten=# SELECT kid_name, "2023-03-28" FROM kids, attendance_list WHERE kids.kid_id = attendance_list.kid_id and "2023-03-28" = '-';
                kid_name                 | 2023-03-28
-----------------------------------------+------------
 Paris Brown                             | -
 Danielle White                          | -
 Josh Dickenson                          | -
 Linda Scott                             | -
 Lily Valley                             | -
(5 строк)


kindergarten=#
```

Нам удалось узнать имена детей, не пришедших в детский сад 28.03.2023.

12. Объединения

Для объединения отношений используется оператор UNION. Рассмотрим следующий пример: SELECT "2023-03-25", "2023-03-26", "2023-03-27", "2023-03-28" FROM attendance_list WHERE group_id = 5 UNION SELECT "2023-03-25", "2023-03-26", "2023-03-27", "2023-03-28" FROM attendance_list WHERE group_id = 6; Данный запрос получает значения столбцов датами для произведения с идентификатором 5 и аналогичные данные,

но в отдельном запросе, для клиента с идентификатором 6. Оператор UNION удаляет дублирующиеся строки, поэтому воспользуемся оператором UNION ALL:

```
kindergarten=# SELECT "2023-03-25", "2023-03-26", "2023-03-27", "2023-03-28" FROM attendance_list WHERE group_id = 5 UNION SELECT "2023-03-25", "2023-03-26"
, "2023-03-27", "2023-03-28" FROM attendance_list WHERE group_id = 6;
 2023-03-25 | 2023-03-26 | 2023-03-27 | 2023-03-28
------------+------------+------------+------------
 +          | -          | +          | +
 +          | +          | -          | -
 +          | +          | +          | +
(3 строки)


kindergarten=# SELECT "2023-03-25", "2023-03-26", "2023-03-27", "2023-03-28" FROM attendance_list WHERE group_id = 5 UNION ALL SELECT "2023-03-25", "2023-03
-26", "2023-03-27", "2023-03-28" FROM attendance_list WHERE group_id = 6;
 2023-03-25 | 2023-03-26 | 2023-03-27 | 2023-03-28
------------+------------+------------+------------
 +          | -          | +          | +
 +          | +          | -          | -
 +          | +          | +          | +
 +          | +          | +          | +
 +          | +          | +          | +
 +          | +          | +          | +
 +          | +          | +          | +
 +          | +          | +          | +
(8 строк)


kindergarten=#
```

Мы получили список посещаемости для детей из 5 и 6 группы.

13. Изменение данных, оператор UPDATE

Изменим номер телефона у одного из тьюторов:

update tutors set tutor_phone_number = ’89009325673’ where tutor_name = ‘Ima Christen’;

Проверим таблицу:

```
kindergarten=# update tutors set tutor_phone_number = '89009325673' where tutor_name = 'Ima Christen';
UPDATE 1
kindergarten=# select * from tutors;
 tutor_id |                tutor_name                 | tutor_age | birth_date | employment_date |                                diploma
                | diploma_serial | tutor_phone_number | gender
----------+-------------------------------------------+-----------+------------+-----------------+-------------------------------------------------------
------------------+----------------+--------------------+--------
       21 | Christen Stell                            |        27 | 1995-09-20 | 2020-01-20      | Bachelor in International Teacher Education for Prima
ry Schools (ITEPS) |         136789 | 89513467890        | woman
       23 | Dove Eyre                                 |        27 | 1995-12-19 | 2018-08-01      | Bachelor in Intercultural Teacher Education
                   |         789061 | 89004562879        | woman
       24 | Olivia Rodgers                            |        31 | 1991-01-01 | 2019-11-30      | Bachelor in International Teacher Education for Prima
ry Schools (ITEPS) |         834567 | 89003458976        | woman
       25 | Candy Watt                                |        30 | 1992-05-24 | 2018-12-02      | Bachelor's Degree in Early Childhood Education Teache
r (Lugo Campus)    |         786123 | 89564873021        | woman
       27 | Lame Starck                               |        28 | 1994-01-28 | 2022-05-26      | Bachelor's Degree in Early Childhood Education Teache
r (Lugo Campus)    |         458767 | 89503457891        | man
       22 | Daniel Michell                            |        30 | 1992-03-04 | 2019-09-30      | Bachelor's Degree in Early Childhood Education Teache
r (Lugo Campus)    |         786520 | 89203456712        | man
       26 | Ima Christen                              |        33 | 1989-02-12 | 2020-04-30      | Bachelor of Science in Science Education
                   |         675490 | 89009325673        | woman
(7 строк)
```

Выведем информацию только о Ima Christen:

```
kindergarten=# select * from tutors where tutor_name = 'Ima Christen';
 tutor_id |                tutor_name                 | tutor_age | birth_date | employment_date |               diploma                | diploma_se
rial | tutor_phone_number | gender
----------+-------------------------------------------+-----------+------------+-----------------+--------------------------------------+-----------
-----+--------------------+--------
       26 | Ima Christen                              |        33 | 1989-02-12 | 2020-04-30      | Bachelor of Science in Science Education |        67
5490 | 89009325673        | woman
(1 строка)


kindergarten=#
```

Вывод: в данной практической работе мы изучили различные операторы языка sql, увидели, как они работают, вывели несколько таблиц с конкретными запросами.