Университет ИТМО

Лабораторная работа № 4

«Запросы на выборку и модификацию данных. Представления. Работа с индексами»

Выполнила: Анисимова Ксения Сергеевна

Группа: К3241

Преподаватель: Говорова Марина Михайловна

Цель работы: овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и индексов.

Оборудование: компьютерный класс.

Программное обеспечение: СУБД PostgreSQL, pgadmin 4.

Практическое задание:

- 1) Создать запросы:
 - 1. Вывести все номера групп и программы, где количество слушателей меньше 10.
 - 2. Вывести список преподавателей с указанием количества программ, где они преподавали за истекший год.
 - 3. Вывести список преподавателей, которые не проводят занятия на третьей паре ни в один из дней недели.
 - 4. Вывести список свободных лекционных аудиторий на ближайший понедельник.
 - 5. Вычислить общее количество обучающихся по каждой программе за последний год.
 - 6. Вычислить среднюю загруженность компьютерных классов в неделю за последний месяц (в часах).
 - 7. Найти самые популярные программы за последние 3 года.
- 2) Создать представление:
 - 1. для потенциальных слушателей, содержащее перечень специальностей, изучаемых на них дисциплин и количество часов;
 - 2. общих доход по каждой программе за последний год.
- **3)** Создать запросы на модификацию данных. Выполнить запросы на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов (составить самостоятельно). В отчете привести формулировку запроса, команду, скриншот до и после выполнения запроса. Учитывается сложность запроса;
- **4)** Создать индексы. Выполнить запросы без индекса и создать планы запросов. Выполнить создание индексов. Выполнить запросы с индексами и создать планы запросов. Сравнить время выполнения запросов. Удалить индексы.

Ход работы:

1) Создать запросы:

1. Вывести все номера групп и программы, где количество слушателей меньше 10. select g.group_name, p.program_name from courses.group g join courses.education e on e.group_name = g.group_name join courses.program p on g.program_id = p.program_id group by g.group_name, p.program_id having count (e.listener_number) < 10</p>

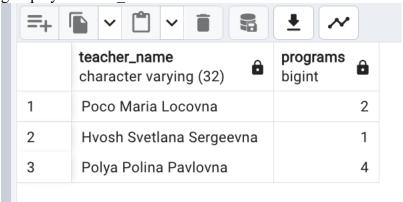
Select g.group_name, p.program_name from courses.group g join courses.education e on e.group_name = g.group_name join courses.program p on g.program_id = p.program_id group by g.group_name, p.program_id having count (e.listener_number) < 10

	group_name character varying	program_name character	ê
1	M12	Дополнительные главы высшей математики - 1	
2	M21	Линейная алгебра - 2	
3	M23	Линейная алгебра - 2	
4	K33	Теория вероятностей и статистика	

2. Вывести список преподавателей с указанием количества программ, где они преподавали за истекший год.

```
select t.teacher_name, count(d.program_id) as programs
from courses.teacher t
    join courses.discipline d on t.teacher_number = d.teacher_number
        join courses.program p on d.program_id = p.program_id
where p.start_date between '2023-01-01' and '2023-12-31' or p.end_date between '2023-01-01' and '2023-12-31'
group by t.teacher_name
```

select t.teacher_name, count(d.program_id) as programs from courses.teacher t join courses.discipline d on t.teacher_number = d.teacher_number join courses.program p on d.program_id = p.program_id where p.start_date between '2023-01-01' and '2023-12-31' or p.end_date between '2023-01-01' and '2023-12-31' group by t.teacher_name



3. Вывести список преподавателей, которые не проводят занятия на третьей паре ни в один из лней недели.

```
select t.teacher_name
from courses.teacher t
where t.teacher_number not in(
    select distinct d.teacher_number
    from courses.discipline d
        join courses.timetable tt on d.discipline_id = tt.discipline_id
    where tt.period = 3);
```

select t.teacher_name from courses.teacher t where t.teacher_number not in(select distinct d.teacher_number from courses.discipline d join courses.timetable tt on d.discipline id = tt.discipline id where tt.period = 3);

	teacher_name character varying (32)
1	Forge Katya Lolitovna
2	Forge Ivan Kolynkin
3	Jorjev Kolyan Kolynkin
4	Lokin Kolyan Sergeev
5	Hvosh Svetlana Sergeevna
6	Horhe Svetlana Golins
7	Shishkina Marina Gopons
8	Polya Polik Pavlovich
9	Solec Polina
10	Solec Pavel

4. Вывести список свободных лекционных аудиторий на ближайший понедельник.

```
select c.classroom_id
from courses.classroom c
where c.classroom_id not in (
    select distinct t.classroom_id
    from courses.timetable t
    where t.week_day = 'NH' and period is not null );
```

select c.classroom_id from courses.classroom c where c.classroom_id not in (select distinct t.classroom_id from courses.timetable t where t.week day = '\PiH' and period is not null);

	classroom_id [PK] integer
1	1
2	2
3	4
4	7
5	9
6	10
7	3
8	5

5. Вычислить общее количество обучающихся по каждой программе за последний год.

```
select p.program_name, count(e.listener_number) as number_of_listeners
from courses.program p
    join courses.group g on p.program_id = g.program_id
        join courses.education e on g.group_name = e.group_name
where p.start_date between '2023-01-01' and '2023-12-31'
    or p.end_date between '2023-01-01' and '2023-12-31'
group by p.program_name
```

select p.program_name, count(e.listener_number) as number_of_listeners from courses.program p join courses.group g on p.program_id = g.program_id join courses.education e on g.group_name = e.group_name where p.start_date between '2023-01-01' and '2023-12-31' or p.end_date between '2023-01-01' and '2023-12-31' group by p.program name

	program_name character	â	number_of_listeners bigint
1	Математический анализ		5
2	Линейная алгебра - 1		11

6. Вычислить среднюю загруженность компьютерных классов в неделю (в часах). select class, avg(periods*1.5) as avg_hours
from (select c.classroom_id as class, count(tt.period) as periods from courses.classroom c
join courses.timetable tt on c.classroom_id = tt.classroom_id where c.type = 'компьютерный класс'
group by c.classroom_id) group by class
select class, avg(periods*1.5) as avg_hours
from (select c.classroom_id as class, count(tt.period) as periods from courses.classroom c
join courses.timetable tt on c.classroom_id = tt.classroom_id where c.type = 'компьютерный класс'
group by c.classroom id) group by class

	class integer	â	avg_hours numeric
1		3	7.50000000000000000
2		5	7.50000000000000000

7. Найти самые популярные программы за последние 3 года.

```
with listener_counts as (
    select p.program_name as program,
    extract(year from p.start_date) as year,
    coalesce(count(distinct e.listener_number), 0) as listener_count
    from courses.program p
        join courses.group g on p.program_id = g.program_id
        join courses.education e on g.group_name = e.group_name
    group by p.program_name, extract(year from p.start_date)
select year, program
from listener_counts lc1
where listener_count = (
        select coalesce(max(lc2.listener_count), 0)
        from listener_counts lc2
        where lc2.year = lc1.year and
    lc1.year between (extract(year from current_date) - 2) and (extract(year from current_date))
);
```

```
with listener counts as (
  select p.program name as program,
  extract(year from p.start date) as year,
  coalesce(count(distinct e.listener number), 0) as listener count
  from courses.program p
    join courses.group g on p.program id = g.program id
    join courses.education e on g.group name = e.group name
  group by p.program name, extract(year from p.start date)
select year, program
from listener counts lc1
where listener count = (
    select coalesce(max(lc2.listener count), 0)
    from listener counts lc2
    where lc2.year = lc1.year and
       lc1.year between (extract(year from current date) - 2) and (extract(year from current date))
         year
                        program
                                                                         a
         numeric
                        character
 1
                2023
                        Линейная алгебра - 1
 2
                2022
                        Линейная алгебра - 2
```

Математический анализ

2) Создать представление:

2021

3

1. для потенциальных слушателей, содержащее перечень специальностей(программ), изучаемых на них дисциплин и количество часов;

	program character	â	discipline character varying (64)	hours integer
1	Линейная алгебра - 1		Алгебра. Начальный уровень	9
2	Линейная алгебра - 1		Линейная алгебра	9
3	Линейная алгебра - 1		Алгебра. Продвинутый уровень	4
4	Линейная алгебра - 1		Математический анализ	4
5	Линейная алгебра - 1		Геометрия. Начальный уровень	9
6	Линейная алгебра - 1		Геометрия	9
7	Линейная алгебра - 1		Геометрия. Продвинутый уровень	4
8	Линейная алгебра - 2		Алгебра. Начальный уровень	5
9	Линейная алгебра - 2		Линейная алгебра	5
10	Линейная алгебра - 2		Геометрия. Начальный уровень	5
11	Линейная алгебра - 2		Геометрия	5

2. общий доход по каждой программе за последний год.

```
create view program_revenue_last_year as
select p.program_name as program,
    sum(p.cost) as total_revenue_last_year
from courses.program p
where extract(year from p.end_date) = extract(year from current_date) - 1
group by p.program_name;

create view program_revenue_last_year as
select p.program_name as program,
    sum(p.cost) as total_revenue_last_year
from courses.program p
where extract(year from p.end_date) = extract(year from current_date) - 1
group by p.program_name;
```

select * from program_revenue_last_year

select * from program_revenue last year

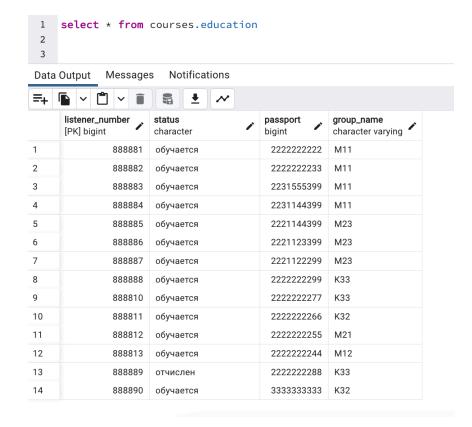
	program character	â	total_revenue_last_year numeric
1	Дополнительные главы высшей математи	1	200000
2	Математический анализ		100000
3	Линейная алгебра - 1		200000

3) Создать запросы на модификацию данных. Выполнить запросы на модификацию данных (INSERT, UPDATE, DELETE) <u>с использованием подзапросов</u> (составить самостоятельно). В отчете привести формулировку запроса, команду, скриншот до и после выполнения запроса. Учитывается сложность запроса;

INSERT - Добавить нового студента в группу с максимальным количеством студентов

```
1
      select* from courses.education
 2
 3
                           Notifications
Data Output
              Messages
                                <u>*</u>
     listener_number
                                                             group_name
                        status
                                               passport
                                                             character varying
      [PK] bigint
                        character
                                               bigint
1
                888881
                         обучается
                                                222222222
                                                             M11
2
                888882
                         обучается
                                                222222233
                                                             M11
3
                888883
                         обучается
                                                2231555399
                                                             M11
4
                888884
                                                2231144399
                                                             M11
                         обучается
5
                                                2221144399
                888885
                         обучается
                                                             M23
6
                888886
                         обучается
                                                2221123399
                                                             M23
7
                                                2221122299
                888887
                         обучается
                                                             M23
8
                888888
                         обучается
                                                222222299
                                                             K33
9
                888810
                         обучается
                                                222222277
                                                             K33
10
                888811
                         обучается
                                                222222266
                                                             K32
11
                888812
                         обучается
                                                222222255
                                                             M21
12
                888813
                                                222222244
                                                             M12
                        обучается
                                                222222288
13
                888889
                        отчислен
                                                             K33
```

```
insert into courses.education (listener_number, status, group_name, passport)
values (
    (select coalesce(max(listener_number) + 1, 1) from courses.education),
    'obywaetcs',
    (select group_name from courses.group where volume = (select max(volume) from courses.group)),
    '3333333333'
);
insert into courses.education (listener_number, status, group_name, passport)
    values (
        (select coalesce(max(listener_number) + 1, 1) from courses.education),
        'obywaetcs',
        (select group_name from courses.group where volume = (select max(volume) from courses.group)),
        '33333333333'
);
```

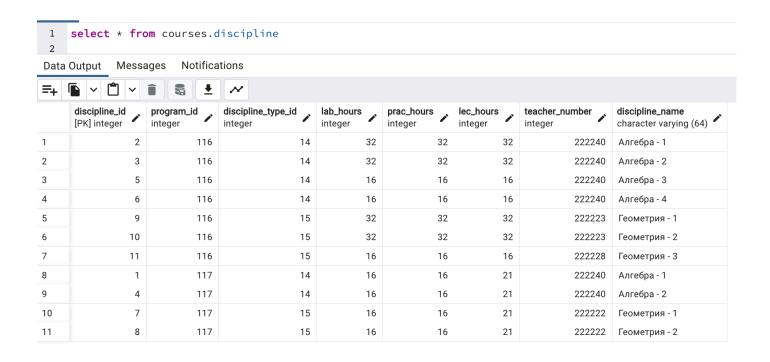


UPDATE - увеличить количество часов лекций для всех дисциплин в определенной программе.

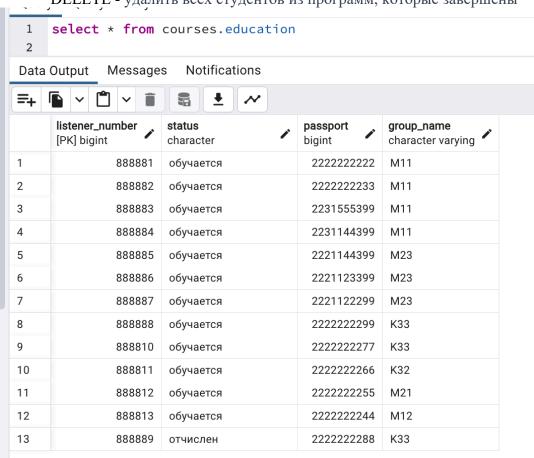
select* from courses.discipline Data Output Messages **Notifications** <u>*</u> =+ ~ discipline_id program_id discipline_type_id lab_hours prac_hours teacher_number discipline_name lec_hours character varying (64) [PK] integer integer integer integer integer integer integer Алгебра - 1 Алгебра - 1 Алгебра - 2 Алгебра - 3 Алгебра - 4 Алгебра - 2 Геометрия - 1 Геометрия - 2 Геометрия - 1 Геометрия - 2 Геометрия - 3

```
UPDATE courses.discipline
SET lec_hours = lec_hours + 5
WHERE program_id = (SELECT program_id FROM courses.program WHERE program_id = 117);
```

UPDATE courses.discipline
SET lec_hours = lec_hours + 5
WHERE program_id = (SELECT program_id FROM courses.program WHERE program_id = 117);



DELETE - удалить всех студентов из программ, которые завершены

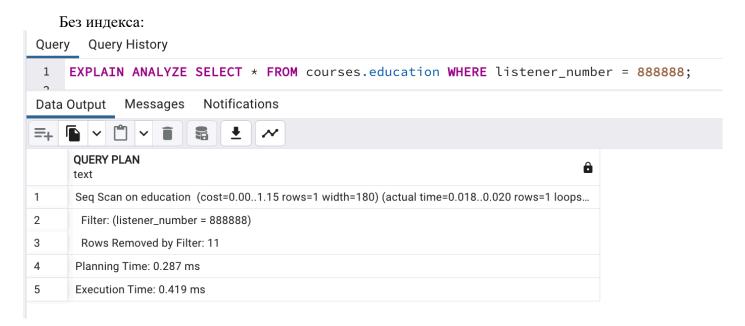


delete from courses.education
where group_name in (
 select g.group_name
 from courses.group g
 join courses.program p on g.program_id = p.program_id
 where p.end_date < current_date
);</pre>

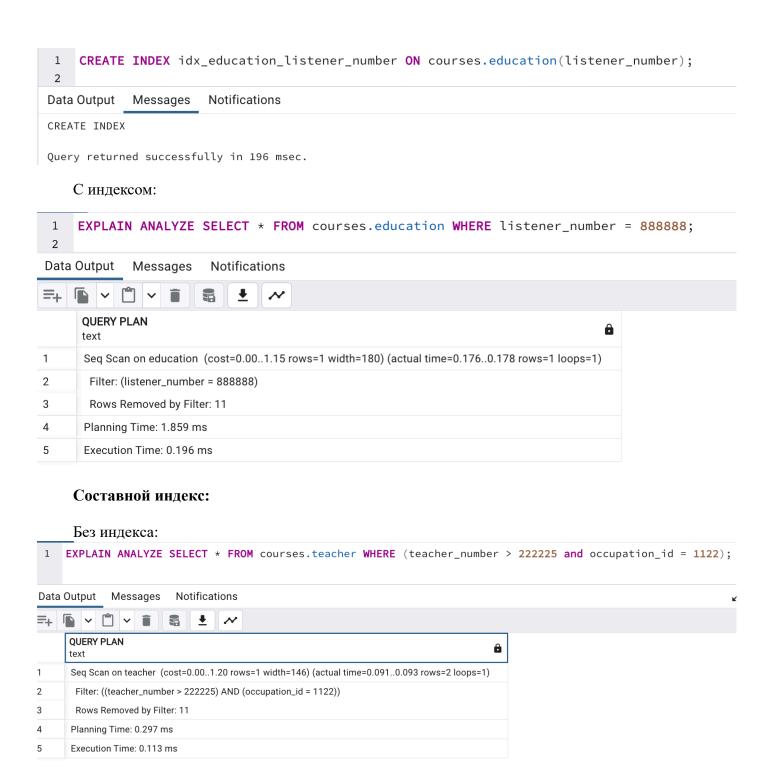
```
where group name in (
         select g.group name
         from courses.group g
         join courses.program p on g.program id = p.program id
         where p.end date < current date
      );
     select * from courses.education
 2
Data Output
              Messages
                           Notifications
                                <u>*</u>
      listener_number
                        status
                                                             group_name
                                               passport
                                                             character varying
      [PK] bigint
                        character
                                               bigint
1
                888881
                                                222222222
                         обучается
                                                             M11
2
                888882
                         обучается
                                                222222233
                                                             M11
3
                888883
                         обучается
                                                2231555399
                                                             M11
4
                888884
                         обучается
                                                2231144399
                                                             M11
5
                         обучается
                888885
                                                2221144399
                                                             M23
6
                888886
                         обучается
                                                2221123399
                                                             M23
7
                         обучается
                888887
                                                2221122299
                                                             M23
8
                888888
                                                222222299
                         обучается
                                                             K33
9
                888810
                         обучается
                                                222222277
                                                             K33
10
                888812
                         обучается
                                                222222255
                                                             M21
11
                888813
                         обучается
                                                222222244
                                                             M12
12
                888889
                         отчислен
                                                222222288
                                                             K33
```

4) Создать индексы. Выполнить запросы без индекса и создать планы запросов. Выполнить создание индексов. Выполнить запросы с индексами и создать планы запросов. Сравнить время выполнения запросов. Удалить индексы.

Простой индекс:



Создаем индекс:



Создаем индекс:

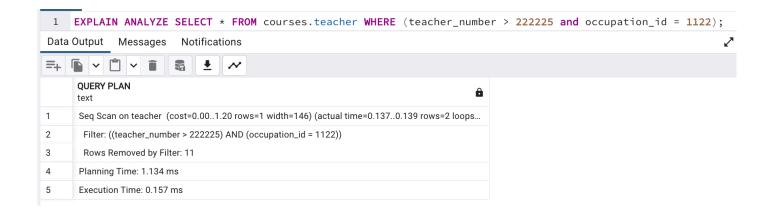
CREATE INDEX idx_teacher_number_occupation ON courses.teacher(teacher_number, occupation_id);

Data Output Messages Notifications

CREATE INDEX

Query returned successfully in 104 msec.

С индексом:



Вывод:

В ходе выполнения лабораторной работы были разработаны и выполнены запросы на выборку данных, а также были созданы представления для базы данных PostgreSQL в соответствии с поставленной индивидуальной задачей. Кроме того, мы успешно реализовали разнообразные запросы на модификацию данных. Провели создание как простых, так и составных индексов, а также проанализировали время выполнения запросов при их использовании.