

ФГБОУ ВО

«Заполярный государственный университет им. Н.М. Федоровского»

Кафедра __ИСиТ__

Специальность ИС-21

ОТЧЕТ

о выполнении лабораторной работы

Выполнил:

Костин Е.А.

Дата:

« » апреля 2023 г.

Норильск 2023

Лабораторная работа №5

Тема: SQL. Индексы и представления. Цель работы: Получение практических навыков работы с СУБД и языком SQL (операторы create index, create view, alter view, drop index, drop view). Задание:

1. Разработать представления к базе данных, созданной и заполненной на предыдущих лабораторных работах, следующих видов:
 - a. простое нематериализованное;
 - b. материализованное неизменяемое;
 - c. простое изменяемое (невозможно изменить неотображаемые в представлении строки);
 - d. простое изменяемое (можно изменить неотображаемые в представлении строки).
2. Выполнить изменение данных в базовых таблицах через изменяемые представления (три разных оператора модификации).
3. Обновить данные в материализованном представлении.
4. Разработать индексы к базе данных, созданной и заполненной на предыдущих лабораторных работах, следующих видов:
 - a. простой в целой таблице;
 - b. составной частичный к таблице;
 - c. уникальный к материализованному представлению;
 - d. с заданной сортировкой составной к таблице.
5. Переименовать одно из представлений.
6. Удалить один из индексов и одно представление. Отчет по лабораторной работе должен содержать:
 1. Фамилию и номер группы учащегося, задание.
 2. Коды операций.
 3. Принтскрины всех выполненных операторов.

1. A.

```
create view onefamily as
select secondname, count(*) from personal
group by secondname
order by count desc
```

Query Query History

1 create view onefamily as
2 select secondname, count(*) from personal
3 group by secondname
4 order by count desc

Data Output Messages Notifications

CREATE VIEW

Query returned successfully in 102 msec.

Query Query History

1 select * from onefamily

Data Output Messages Notifications

	secondname character varying (64)	count bigint
1	Долгонос	2
2	Маслеников	1
3	Богатырь	1

B.

```
create materialized view allpro as
select name from produser
union all select name from provider
```

Query Query History

1 create materialized view allpro as
2 select name from produser
3 union all select name from provider
4

Data Output Messages Notifications

SELECT 6

Query returned successfully in 67 msec.

The screenshot shows the DBeaver interface. At the top, the 'Query' tab is active, displaying the SQL query: `SELECT * FROM allpro`. Below the query editor, the 'Data Output' tab is selected, showing a table with 6 rows and 1 column named 'name'. The table has a lock icon in the top right corner. The data rows are as follows:

	name
1	Чих-Пых
2	Пух-Плюх
3	Жыж-Пыж
4	Баломут
5	Травы-Нави
6	Фармолаб.Inc

C.

```
create view provi as
```

```
SELECT name, inn FROM provider
```

```
Query History
```

1	create view provi as
2	SELECT name,inn FROM provider
3	

```
Data Output Messages Notifications
```

```
CREATE VIEW
```

Query returned successfully in 85 msec.

```
Query    Query History  
1 update provi set address = 'пупушка' where inn = 999111222  
  
Data Output Messages Notifications  
ERROR: ОШИБКА: столбец "address" в таблице "provi" не существует  
LINE 1: update provi set address = 'пупушка' where inn = 999111222  
                          ^  
  
SQL state: 42703  
Character: 18
```

The screenshot shows the 'Query' tab in the SQL editor. The query entered is: `1 select * from provi`. Below the editor, the 'Data Output' tab is active, displaying a table with 3 rows and 2 columns. The columns are 'name' (character varying (64)) and 'inn' (integer). The rows contain the following data:

	name	inn
1	Баломут	234078951
2	Травы-Нави	173329862
3	Фармолаб. Inc	999111222

D.

```
create view prodo as  
select * from produser
```

Query Query History

```
1 create view prodo as  
2 select * from produser
```

Data Output Messages Notifications

CREATE VIEW

Query returned successfully in 82 msec.

Query Query History

```
1 select * from prodo
```

Data Output Messages Notifications

	id_producer integer	name character varying (64)	inn integer
1	1	Чих-Пых	228133133
2	2	Пух-Плюх	173322912
3	3	Жыж-Пыж	992371265

Query Query History

```
1 update prodo set inn = 9110204 where inn = 228133133
```

Data Output Messages Notifications

UPDATE 1

Query returned successfully in 83 msec.

Query Query History

```
1 select * from prodo
```

Data Output Messages Notifications

	id_producer integer	name character varying (64)	inn integer
1	2	Пух-Плюх	173322912
2	3	Жыж-Пыж	992371265
3	1	Чих-Пых	9110204

2.

Query		Query History	
1		<code>select * from prodo</code>	

Data Output		Messages	Notifications
	<code>id_producer</code> integer	<code>name</code> character varying (64)	<code>inn</code> integer
1	2	Пух-Плюх	173322912
2	3	Жыж-Пыж	992371265
3	1	Чих-Пых	9110204

update prodo set inn = 9281137 where inn = 173322912

Query		Query History	
1		<code>update prodo set inn = 9281137 where inn = 173322912</code>	

Data Output		Messages	Notifications
UPDATE 1			

Query returned successfully in 83 msec.

insert into prodo values
(4, 'Киш-Миш', 3312986)

Query		Query History	
1		<code>insert into prodo values</code>	
2		<code>(4, 'Киш-Миш', 3312986)</code>	

Data Output		Messages	Notifications
INSERT 0 1			

Query returned successfully in 83 msec.

update prodo set inn = 9110204 where inn = 228133133

Query		Query History	
1		<code>update prodo set inn = 9110204 where inn = 228133133</code>	

Data Output		Messages	Notifications
UPDATE 1			

Query returned successfully in 83 msec.

Query

Query History

1


select * from prodo

Data Output


Messages


Notifications


≡+





▼











	id_producer integer	name character varying (64)	inn integer
1	3	Жыж-Пыж	992371265
2	1	Чих-Пых	9110204
3	2	Пух-Плюх	9281137
4	4	Киш-Миш	3312986

3. refresh materialized view allpro

Query	Query History	
1	refresh materialized view allpro	
Data Output	Messages	Notifications
REFRESH MATERIALIZED VIEW		
Query returned successfully in 170 msec.		

4. a.create index idxpri on drug(price)

Query	Query History	
1	create index idxpri on drug(price)	
Data Output	Messages	Notifications
CREATE INDEX		
Query returned successfully in 81 msec.		

b. create index idxwh on post(name, pay) where pay > 60000

Query		Query History	
1	create index idxwh on post(name, pay)	2	where pay > 60000
Data Output		Messages	Notifications
CREATE INDEX			
Query returned successfully in 75 msec.			

c. create unique index idxpred on allpro(name)

Query	Query History	
1	create unique index indxpred on allpro(name)	
2		
Data Output	Messages	Notifications
CREATE INDEX		
Query returned successfully in 84 msec.		

d. create index idx4 on drug(name, price desc)

Query	Query History	
1	<code>create index idx4 on drug(name, price desc)</code>	
2		
Data Output	Messages	Notifications
CREATE INDEX		
Query returned successfully in 177 msec.		

5.
alter view prodo rename to frodo

Query	Query History	
1	<code>alter view prodo rename to frodo</code>	
Data Output	Messages	Notifications
ALTER VIEW		
Query returned successfully in 95 msec.		

6.
drop index idxpri;
drop view frodo;

Query	Query History	
1	drop index idxpri;	
2	drop view frodo;	
Data Output	Messages	Notifications
DROP VIEW		
Query returned successfully in 76 msec.		