

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра технічної кібернетики

Звіти до комп'ютерних практикумів з кредитного модуля “Бази даних”

Прийняв

доцент кафедри АСОІУ

Ліщук К.І.

29.12.2020р

Виконав

Студент групи ІТ-02

Макаров І.С.

Київ – 2020

Комп'ютерний практикум № 4

Мета:

- Вивчити оператор, котрий використовується в реляційних СУБД, для вибірки даних з таблиць, групування та сортування даних
- Навчитись використовувати вбудовані функції в запитах
- Вивчити призначення представлень (view) баз даних, синтаксису та семантики команд SQL для їх створення, зміни та видалення, системних збережених процедур для отримання інформації про представлення

Постановка задачі комп'ютерного практикуму № 4

При виконанні комп'ютерного практикуму необхідно виконати наступні дії:

1) Створити наступні запити:

- запит з використанням функції COUNT;
- запит з використанням функції SUM;
- запит з використанням функцій UPPER, LOWER;
- запит з використанням функцій для роботи з датами;
- запит з використанням групування по одному стовпцю;
- запит з використанням групування по декільком стовпцям;
- запит з використанням умови відбору груп HAVING;
- запит з використанням HAVING без GROUP BY;
- запит з використанням сортування по стовпцю.

2) Робота з представленнями (view):

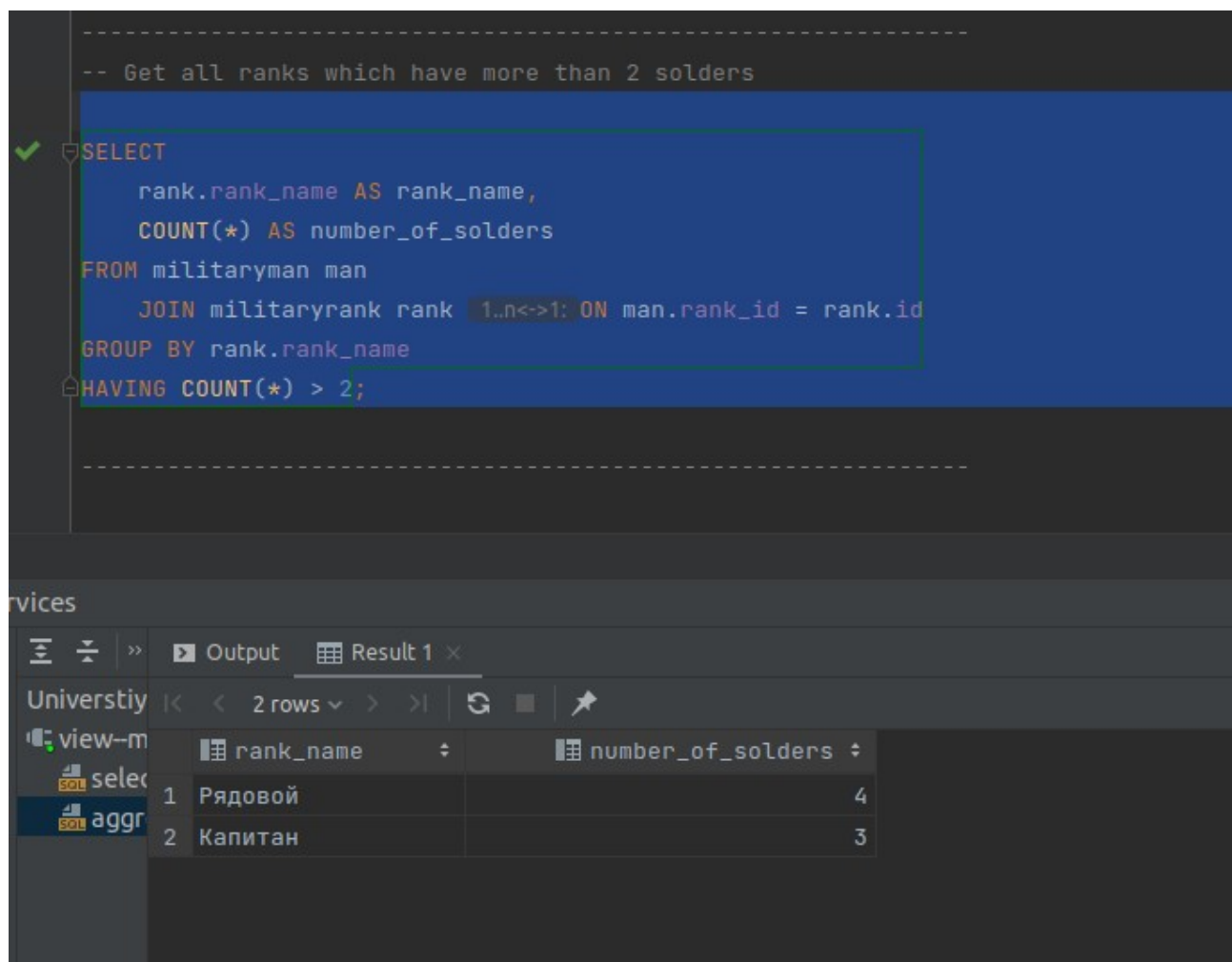
- створити представлення, котре містить дані з декількох таблиць;
- створити представлення, котре містить дані з декількох таблиць та посилання, котре створене в п.а;
- модифікувати запити з використанням команди ALTER VIEW;
- отримати довідникову інформацію про ці представлення з використанням вбудованих процедур (наприклад в MsSQL sp_help, sp_helptext и sp_depends).

РЕЗУЛЬТАТЫ

Ссылка на код: <https://github.com/kinfi4/Ministry-of-Defence-database-scripts>

В рамках работы были написанны aggregation-queries.sql, views/

Примеры выполнения запросов:



The screenshot shows a SQL IDE interface. The top panel displays a SQL query with a comment: "-- Get all ranks which have more than 2 solders". The query is as follows:

```
SELECT
    rank.rank_name AS rank_name,
    COUNT(*) AS number_of_solders
FROM militaryman man
    JOIN militaryrank rank 1..n<->1: ON man.rank_id = rank.id
GROUP BY rank.rank_name
HAVING COUNT(*) > 2;
```

The bottom panel shows the results of the query in a table format. The table has two columns: "rank_name" and "number_of_solders". There are two rows of data:

	rank_name	number_of_solders
1	Рядовой	4
2	Капитан	3

```
-----  
-- Getting number of solder for each specialization for each rank
```

```
SELECT  
    s.name AS specialization_name,  
    rank.rank_name AS rank_name,  
    COUNT(*) AS number_of_solders  
FROM militaryman man  
    JOIN militaryman__specialization ms 1<->1..n: ON man.id = ms.military_man_id  
    JOIN specialization s 1..n<->1: ON ms.specialization_id = s.id  
    JOIN militaryrank rank 1..n<->1: ON man.rank_id = rank.id  
GROUP BY s.name, rank.rank_name  
ORDER BY number_of_solders DESC;
```

vices

Output Result 2 x			
Universtiy			
	specialization_name	rank_name	number_of_solders
1	Сапер	Рядовой	3
2	Десантник	Рядовой	3
3	Водитель	Капитан	2
4	Инженер	Майор	2
5	Сапер	Прапорщик	2
6	Десантник	Капитан	1
7	Инженер	Генерал	1
8	Снайпер	Старшина	1

```
1 CREATE OR REPLACE VIEW view_get_basic_military_man_info AS
2 SELECT
3     info.solder_name,
4     info.rank_name,
5     district.region_name
6 FROM view_get_military_man_full_info info
7     JOIN district ON info.district_id = district.id
8 WHERE info.graduation_year >= 2000;
9
10 SELECT * FROM view_get_basic_military_man_info;
```

Services

UniverstiyStudies.pu...sic_military_man_info			
	solder_name	rank_name	region_name
1	Роберт	Майор	Львов
2	Борис	Капитан	Львов
3	Гарик	Капитан	Донецк
4	Марк	Капитан	Киев
5	Марк	Майор	Чернигов
6	Илья	Генерал	Чернигов
7	Ира	Генерал	Чернигов

