

Федеральное агентство связи
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Сибирский государственный университет
телекоммуникаций и информатики»

Лабораторная работа №1
«Язык SQL. Запрос данных.
Использование реляционных, булевых и специальных операторов»
Вариант №9

Выполнил: студент IV курса

ИВТ, гр. ИП-713

Михеев Н.А.

Проверил: доцент кафедры ПМИК

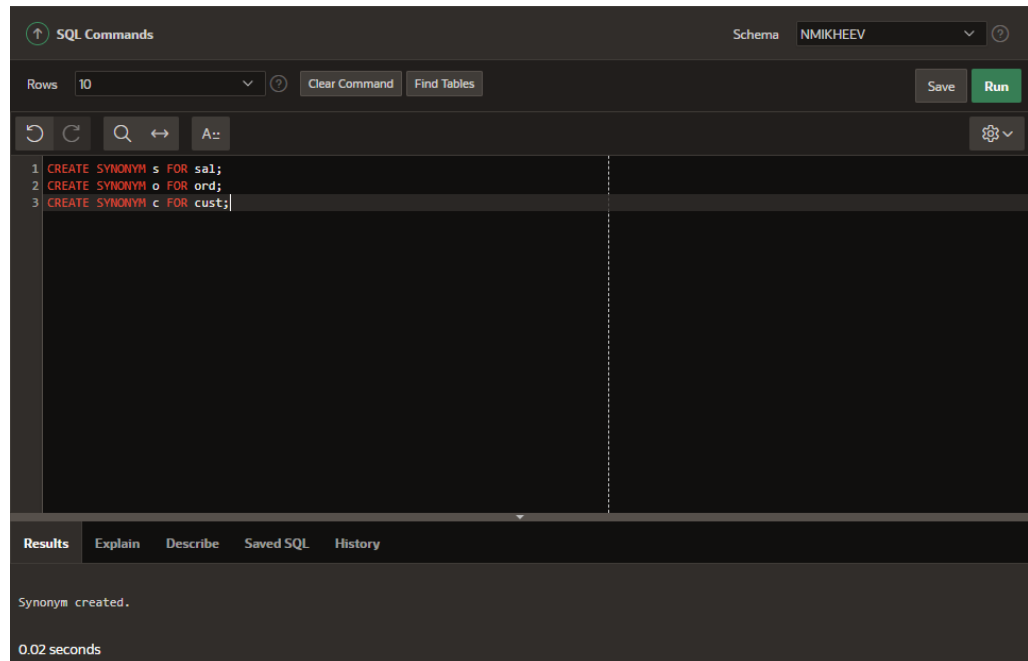
Приставка П. А.

Новосибирск, 2020 г.

В ходе выполнения лабораторной работы были созданы и заполнены три учебные базы данных – таблицы sal, cust и ord. Все работы были проведены в среде разработки apex.oracle.com. Далее конкретно по заданиям.

Задание №2.

Были заданы синонимы для баз данных:



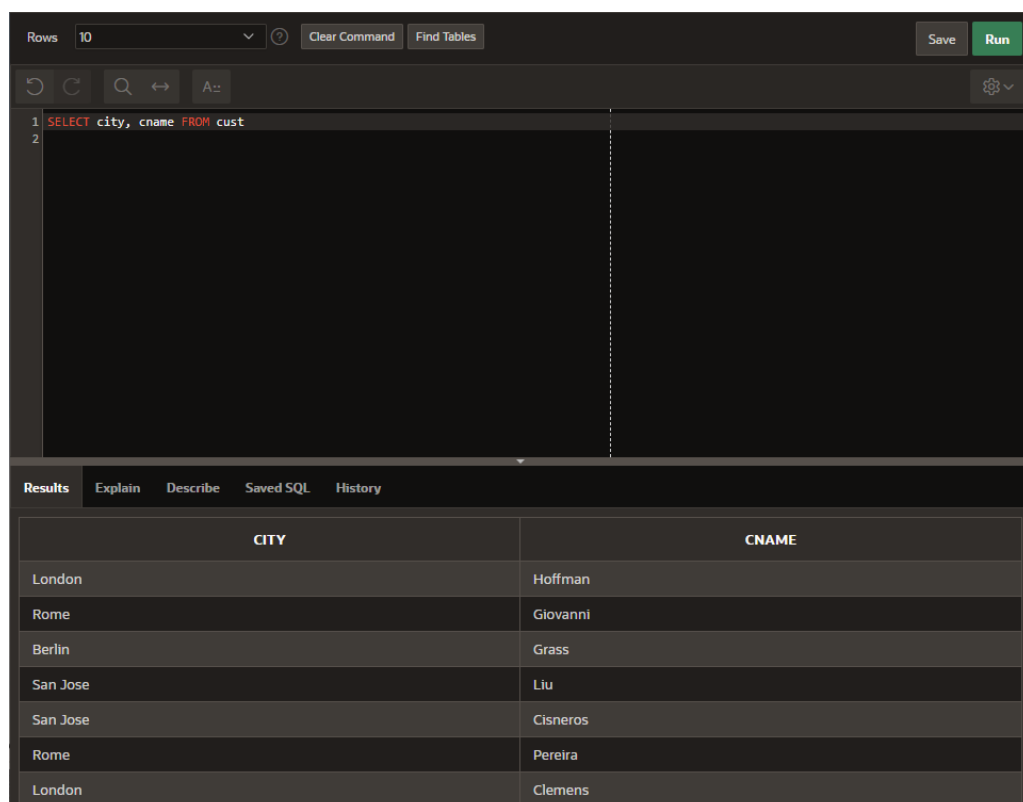
```
1 CREATE SYNONYM s FOR sal;
2 CREATE SYNONYM o FOR ord;
3 CREATE SYNONYM c FOR cust;
```

Results Explain Describe Saved SQL History

Synonym created.

0.02 seconds

Сравнение команд SELECT * FROM cust и SELECT city, cname FROM cust. Где в 1 случае выводится вся таблица, а во втором только поля city и cname.



```
1 SELECT city, cname FROM cust
```

Results Explain Describe Saved SQL History

CITY	CNAME
London	Hoffman
Rome	Giovanni
Berlin	Grass
San Jose	Liu
San Jose	Cisneros
Rome	Pereira
London	Clemens

Команда DISTINCT позволяет вывести значения столбца без повторений данных, так из таблицы sal были получены все уникальные города:

The screenshot shows a SQL interface with the command `SELECT DISTINCT city FROM sal` entered. The results tab is active, displaying a table with one column named 'CITY' and four rows: London, Barcelona, San Jose, and New York. The status bar indicates '4 rows returned in 0.01 seconds'.

CITY
London
Barcelona
San Jose
New York

Далее я подробно ознакомился с теоретической частью задания 2 и 3.

Задание №4.

Первая команда - `SELECT * FROM cust WHERE NOT city = 'San Jose' OR rating > 200;` - делает выборку где город НЕ Сан-Хосе или где рейтинг больше 200, отрицание применяется только к городам.

The screenshot shows a SQL interface with the command `SELECT * FROM cust WHERE NOT city = 'San Jose' OR rating > 200;` entered. The results tab is active, displaying a table with five columns: CNUM, CNAME, CITY, RATING, and SNUM. The status bar indicates '6 rows returned in 0.02 seconds'.

CNUM	CNAME	CITY	RATING	SNUM
2001	Hoffman	London	100	1001
2002	Giovanni	Rome	200	1003
2004	Grass	Berlin	300	1002
2008	Cisneros	San Jose	300	1007
2007	Pereira	Rome	100	1004
2006	Clemens	London	100	1001

Во второй же команде - `SELECT * FROM cust WHERE NOT (city = 'San Jose' OR rating > 200)` - отрицание применяется к обоим условиям выборки.

Rows: 10 Clear Command Find Tables Save Run

↶ ↷ 🔍 ↔ A.. ⚙️

```

1 SELECT * FROM cust
2 WHERE NOT (city = 'San Jose' OR rating > 200)
3

```

Results Explain Describe Saved SQL History

CNUM	CNAME	CITY	RATING	SNUM
2001	Hoffman	London	100	1001
2002	Giovanni	Rome	200	1003
2007	Pereira	Rome	100	1004
2006	Clemens	London	100	1001

4 rows returned in 0.01 seconds Download

Задание №5-6.

В первом случае команда сработала корректно и вывелись нужные продавцы. Во втором случае был задан неверный промежуток от большего к меньшему, соответственно данные не были обнаружены в таблицах.

Rows: 10 Clear Command Find Tables Save Run

↶ ↷ 🔍 ↔ A.. ⚙️

```

1 SELECT * FROM sal WHERE comm BETWEEN .10 AND .12
2
3

```

Results Explain Describe Saved SQL History

SNUM	SNAME	CITY	COMM
1001	Peel	London	.12
1004	Motica	London	.11
1003	Axelrod	New York	.1

3 rows returned in 0.00 seconds Download

Rows: 10 Clear Command Find Tables Save Run

↶ ↷ 🔍 ↔ A.. ⚙️

```

1 SELECT * FROM sal WHERE comm BETWEEN .12 AND .10
2
3

```

Results Explain Describe Saved SQL History

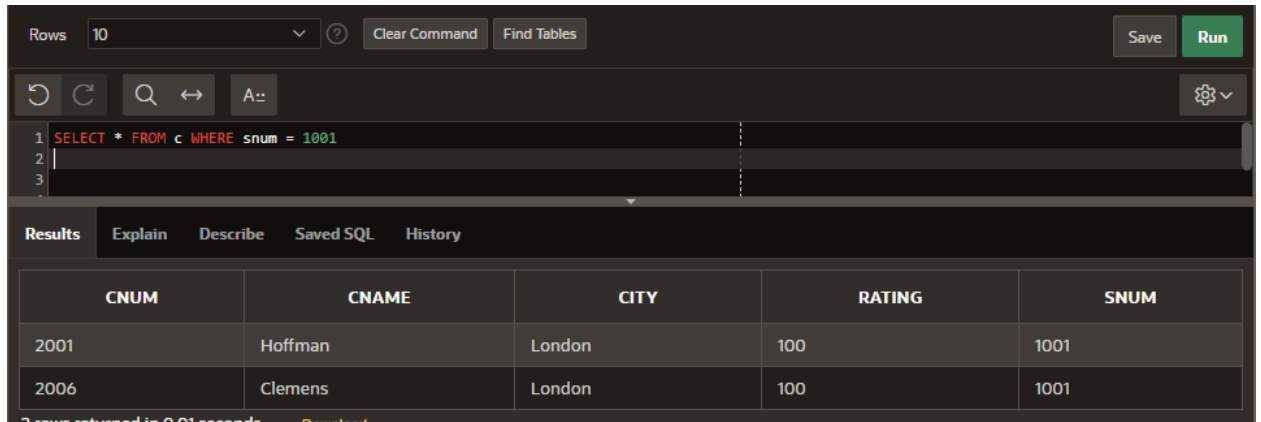
no data found

Задания бригады №1.

Номер бригады выбирался в соответствии с вариантом в учебном журнале.

1. Напишите запрос, который выводит все строки из таблицы Покупателей, для которых номер продавца равен 1001.

Для выполнения данного пункта был составлен запрос: `SELECT * FROM c WHERE snum = 1001`. Результат выполнения запроса на скриншоте:

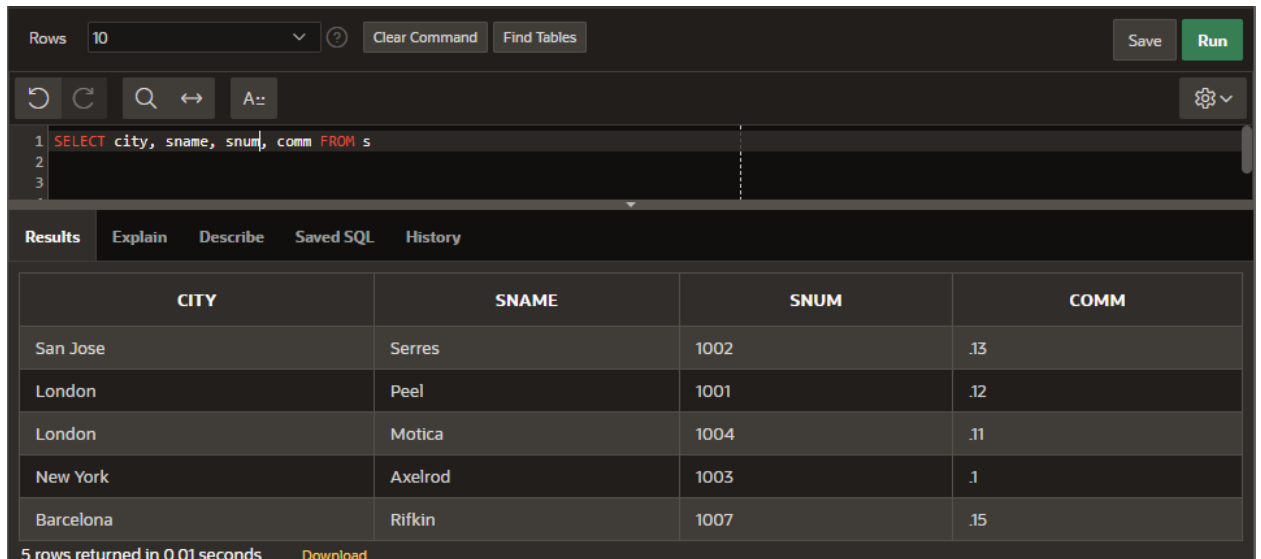


The screenshot shows a SQL query editor with the query `SELECT * FROM c WHERE snum = 1001` entered. The results are displayed in a table with 5 columns: CNUM, CNAME, CITY, RATING, and SNUM. Two rows are returned, both with SNUM = 1001.

CNUM	CNAME	CITY	RATING	SNUM
2001	Hoffman	London	100	1001
2006	Clemens	London	100	1001

2. Напишите запрос, который выводит таблицу Продавцов со столбцами в следующем порядке: city, sname, snum, comm.

Был составлен запрос `SELECT city, sname, snum, comm FROM s`.



The screenshot shows a SQL query editor with the query `SELECT city, sname, snum, comm FROM s` entered. The results are displayed in a table with 4 columns: CITY, SNAME, SNUM, and COMM. Five rows are returned.

CITY	SNAME	SNUM	COMM
San Jose	Serres	1002	.13
London	Peel	1001	.12
London	Motica	1004	.11
New York	Axelrod	1003	.1
Barcelona	Rifkin	1007	.15

3. Напишите запрос, который выводит оценку (rating), сопровождаемую именем каждого покупателя в городе San Jose.

Для выполнения был составлен запрос по выбору рейтинга и имени покупателя из таблицы cust, с выборкой по городу = San Jose.

`SELECT rating, custname FROM c WHERE city = 'San Jose'`

Скриншот работы запроса:

The screenshot shows a SQL query execution interface. At the top, there's a 'Rows' dropdown set to 10, and buttons for 'Clear Command' and 'Find Tables'. The query entered is `SELECT rating, cname FROM c WHERE city = 'San Jose'`. Below the query editor, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is active, displaying a table with two columns: 'RATING' and 'CNAME'. The table contains two rows of data. At the bottom, it states '2 rows returned in 0.13 seconds' and has a 'Download' button.

RATING	CNAME
200	Liu
300	Cisneros

4. Напишите запрос, который выводит значение номера продавца всех продавцов из таблицы Заказов без каких бы то ни было повторений.

Для получения номеров продавцов без повторений было использован макрос DISTINCT:

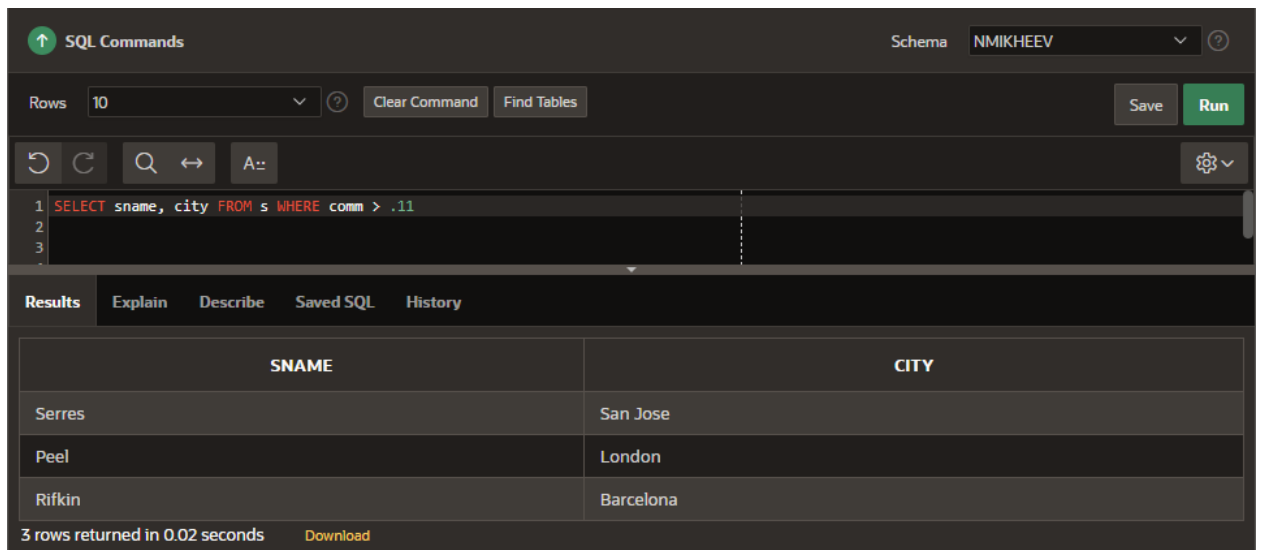
The screenshot shows a SQL query execution interface. At the top, there's a 'Schema' dropdown set to 'NMIKHEEV'. The query entered is `SELECT DISTINCT snum FROM s`. Below the query editor, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is active, displaying a table with one column: 'SNUM'. The table contains five rows of data. At the bottom, it states '5 rows returned in 0.01 seconds' and has a 'Download' button.

SNUM
1001
1002
1003
1004
1007

5. Напишите запрос, который может выдать вам sname и city для всех продавцов в Лондоне с комиссиями строго больше .11

Запрос: `SELECT sname, city FROM s WHERE comm > .11`, был использован макрос WHERE с булевым оператором строгого неравенства.

Результат работы на скриншоте:



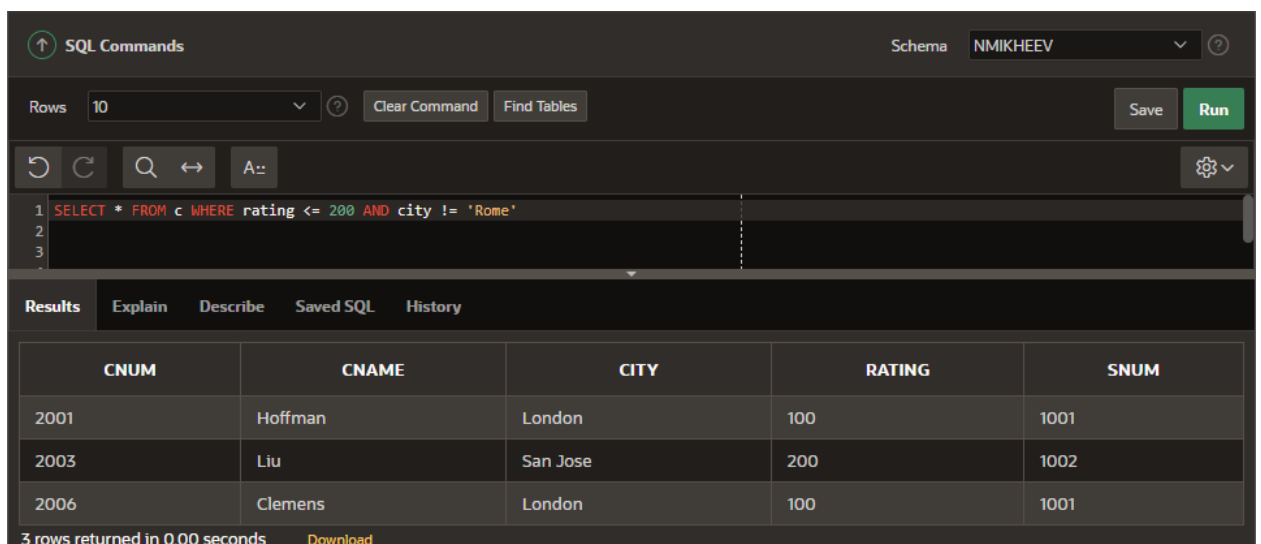
The screenshot shows a SQL interface with the following components:

- SQL Commands:** Schema: NMIKHEEV. Rows: 10. Buttons: Clear Command, Find Tables, Save, Run.
- Query:** `SELECT sname, city FROM s WHERE comm > .11`
- Results:** 3 rows returned in 0.02 seconds. Download button.

SNAME	CITY
Serres	San Jose
Peel	London
Rifkin	Barcelona

6. Напишите запрос к таблице Покупателей, чей вывод может включить всех покупателей с рейтингом меньше или равным 200, если они не находятся в Риме

Разработанный запрос включает в себя выборку по рейтингу меньше или равным 200, и чтобы покупатель был не из Рима.



The screenshot shows a SQL interface with the following components:

- SQL Commands:** Schema: NMIKHEEV. Rows: 10. Buttons: Clear Command, Find Tables, Save, Run.
- Query:** `SELECT * FROM c WHERE rating <= 200 AND city != 'Rome'`
- Results:** 3 rows returned in 0.00 seconds. Download button.

CNUM	CNAME	CITY	RATING	SNUM
2001	Hoffman	London	100	1001
2003	Liu	San Jose	200	1002
2006	Clemens	London	100	1001

7. Запросите двумя способами все заказы на 3 и 5 Октября 1990 г.

Первый вариант запроса: `SELECT * FROM o WHERE odate = '10-03-1990' or odate = '10-05-1990'`.

Примечание: из-за американского варианта представления даты, месяц и день поменяны местами в выводе таблицы.

SQL Commands

Schema: NMIKHEEV

Rows: 10

Clear Command Find Tables Save Run

```
1 SELECT * FROM o WHERE odate = '10-03-1990' or odate = '10-05-1990'
```

Results Explain Describe Saved SQL History

ONUM	AMT	ODATE	CNUM	SNUM
3006	1098.16	10/03/1990	2008	1007
3001	18.69	10/03/1990	2008	1007
3008	4723	10/05/1990	2006	1001
3002	1900.1	10/03/1990	2007	1004
3005	5160.45	10/03/1990	2003	1002
3003	76719	10/03/1990	2001	1001

6 rows returned in 0.00 seconds Download

Второй вариант запроса, с использованием оператора IN: SELECT * FROM o WHERE odate IN ('10-03-1990', '10-05-1990'). Как видно на скриншоте – результат абсолютно одинаковый

SQL Commands

Schema: NMIKHEEV

Rows: 10

Clear Command Find Tables Save Run

```
1 SELECT * FROM o WHERE odate IN ('10-03-1990', '10-05-1990')
```

Results Explain Describe Saved SQL History

ONUM	AMT	ODATE	CNUM	SNUM
3006	1098.16	10/03/1990	2008	1007
3001	18.69	10/03/1990	2008	1007
3008	4723	10/05/1990	2006	1001
3002	1900.1	10/03/1990	2007	1004
3005	5160.45	10/03/1990	2003	1002
3003	76719	10/03/1990	2001	1001

6 rows returned in 0.01 seconds Download