

Федеральное государственное бюджетное
образовательное учреждение высшего
образования «Казанский национальный
исследовательский технический университет им.

А.Н. Туполева-КАИ»

СПО ИКТЗИ

Колледж информационных технологий

Лекция № 8

**на тему «Функции и операторы для
работы со строками и текстом.**

Объединение таблиц»

**По дисциплине «Технология
разработки и защиты баз данных»**

ПРЕПОДАВАТЕЛЬ:

КАСИМОВА АЛИНА РИНАДОВНА

Казань, 2021
2021/2022 учебный год
Осенний семестр

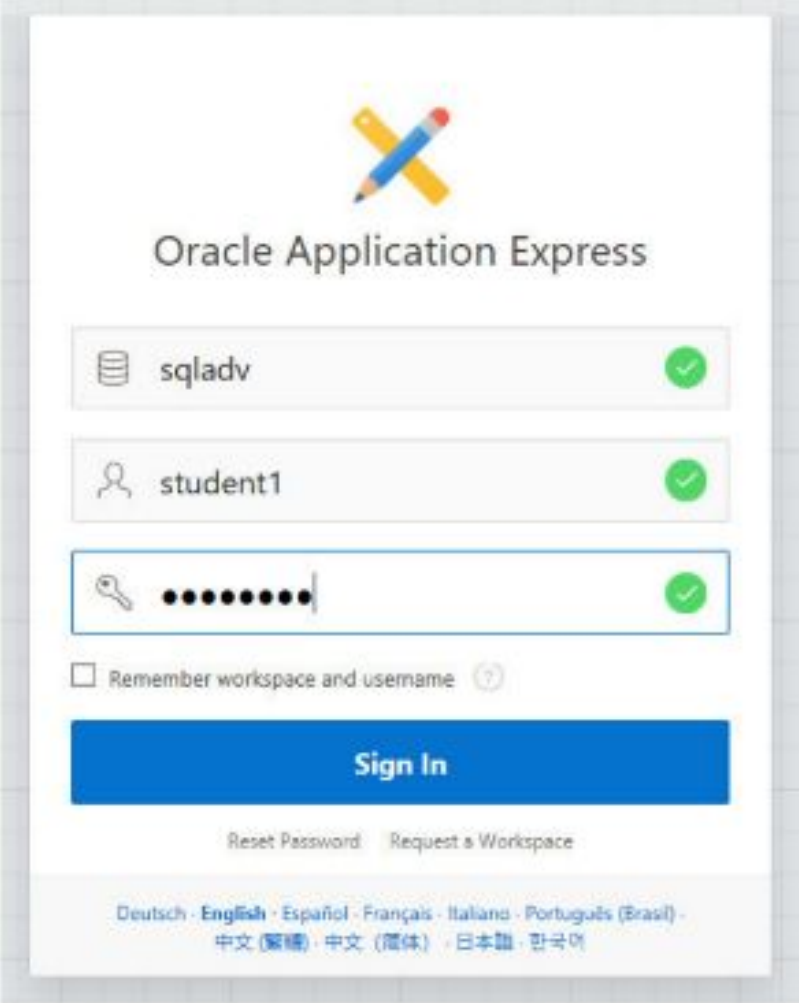
Учебные вопросы

1. Используемая БД
2. Функции и операторы для работы со строками и текстом

Используемая база данных

ИСПОЛЬЗУЕТСЯ ДЛЯ ЛЕКЦИЙ 8-12

<https://apex.oracle.com/pls/apex/>



The image shows the Oracle Application Express login interface. At the top, there is a logo consisting of two crossed pencils, one yellow and one blue. Below the logo, the text "Oracle Application Express" is displayed. The login form consists of three input fields, each with a green checkmark on the right side, indicating successful validation. The first field is for the workspace name, labeled "sqladv". The second field is for the username, labeled "student1". The third field is for the password, represented by a series of black dots. Below the password field, there is a checkbox labeled "Remember workspace and username" with a help icon to its right. A large blue button labeled "Sign In" is positioned below the checkbox. At the bottom of the form, there are two links: "Reset Password" and "Request a Workspace". At the very bottom, there is a list of supported languages: Deutsch, English, Español, Français, Italiano, Português (Brasil), 中文 (繁體), 中文 (简体), 日本語, and 한국어.

Oracle Application Express

sqladv ✓

student1 ✓

•••••••• ✓

Remember workspace and username ?

Sign In

[Reset Password](#) [Request a Workspace](#)

Deutsch · **English** · Español · Français · Italiano · Português (Brasil) ·
中文 (繁體) · 中文 (简体) · 日本語 · 한국어

Структура БД

Первая таблица MAN содержит сведения о людях, которые приобрели машины.

Колонки таблицы MAN:

- PHONEnum – уникальный телефонный номер человека, первичный ключ для таблицы MAN, содержит текстовые данные;
- CITYCode – код города, вторичный ключ для связи с таблицей CITY;
- FirstName – имя человека (текстовые данные);
- LAsTName – фамилия человека (текстовые данные);
- YearOld – возраст человека (числовые данные).

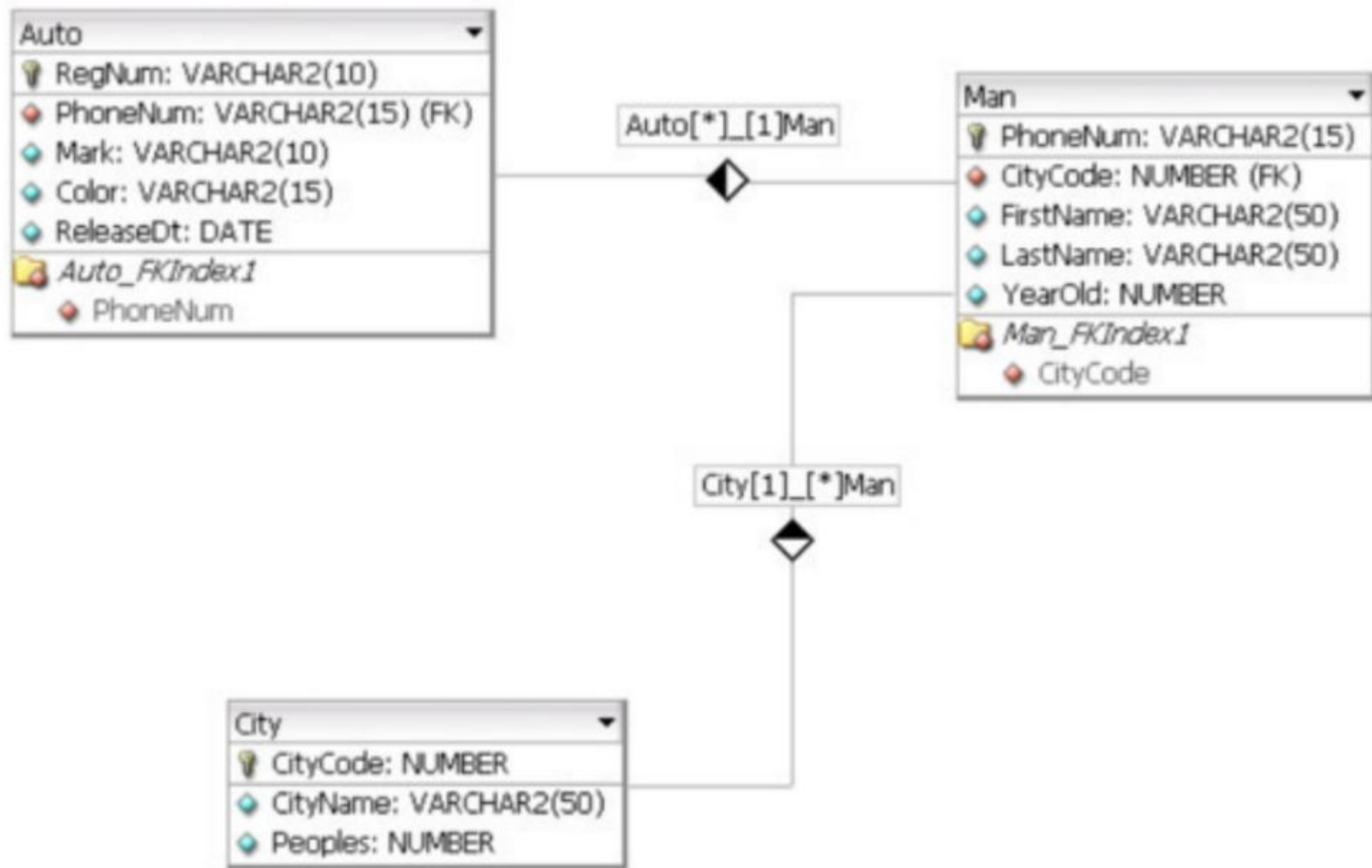
Таблица CITY – справочник городов, состоит из трех колонок:

- CITYCODE – уникальный код города, ключевое поле для таблицы CITY (числовые данные);
- CITYNAME – наименование города (текстовые данные);
- PEOPLES – население города, количество человек, которые проживают в городе (числовые данные).

Таблица AUTO – сведения об автомобилях автосалона.

Колонки таблицы AUTO:

- REGnum – уникальный регистрационный номер автомобиля (содержит текстовые данные);



Функции и операторы для работы со строками и текстом

Теория и практика

Мы уже познакомились с оператором LIKE, позволяющим выбирать из базы данных строки с текстом, соответствующие определенному шаблону.

В SQL ORACLE также есть множество полезных сервисных функций для модификации

строк, работы с подстроками, объединения строк.

Объединение строк – по-правильному конкатенация.

Для объединения строк в языке SQL д синтаксическая конструкция ||.

```
select 'теле' || 'визор' as q from DUAL
- телевизор
select 'LA' || 'DA' as q from DUAL
- LADA
select firstname || ' ' || lastname as q from man
```

INSTR– поиск позиции подстроки в строке.

INSTR (STR1, STR2, POSn, DIRECTION) – возвращает позицию STR2 в строке STR1,

где поиск осуществляется в позиции POSn

в направлении DIRECTION 1 – от начала строки, 0 – от окончания строки, то есть откуда

мы начинаем поиск – от начала строки или с конца строки.

```
SELECT INSTR('AAABAAAAABA','AB',1) FROM DUAL
-- 3
SELECT INSTR('AAABAAAAABA','AB',5) FROM DUAL
-- 9
SELECT INSTR('AAABAAAAABA','AB',1,1) FROM DUAL
-- 3
```

Найти первое вхождение буквы «а» в название городов в таблице CITY, вывести на экран

и вхождение символа «а».

Примеры

```
select cityname, instr(cityname,'a',1) iname from city
```

CITYNAME	INAME
Казань	0
Котлас	0
Мурманск	0
Санкт-Петербург	0
Орел	0
Курск	0
Ярославль	0
Москва	0
Владимир	0

Найти последнее вхождение буквы «а» в название городов в таблице CITY, вывести на экран и номер последнего вхождения символа «а».

```
select cityname, instr(cityname,'a',2) iname from city
```

Results Explain Describe Saved SQL History

CITYNAME

INAME

Казань	0
Котлас	0
Мурманск	0
Санкт-Петербург	0
Орел	0
Курск	0
Ярославль	0
Москва	0
Владимир	0

Length – длина строки в символах.

LENGTH (str1) возвращает длину строки

Str1 в символах.

Примеры

Длина строки «AAA»

```
select length('AAA') as ln from dual
```

– 3.
Длина строки «AAA1234»

```
select length('AAA1234') as ln from dual
```

– 7.
Длина строки «привет мир»

```
select length('привет мир') as ln from dual
```

– 10.

Вывести из таблицы MAN имя, фамилию, длину имени и фамилии в символах.

```
select firstname,lastname,length(firstname) as fnl, length(lastname) ln1 from man
```

FIRSTNAME	LASTNAME	FNL	LNL
Максим	Москитов	6	8
Олег	Денисов	4	7
Максим	Иванов	6	6
Андрей	Николаев	6	8
Костя	Пупкин	5	6
Алексей	Галкин	7	6
Максим	Леонидов	6	8
Алиса	Никина	5	6
Андрей	Некрасов	6	8
Михаил	Васильев	4	7

Выбор подстроки из строки SUBSTR

SUBSTR (STR1, POS, LEN) выбирает LEN символов в строке str1, начиная с позиции POS.

STR1 – оригинальная строка.

POS – позиция, с которой начинается выделение.

NEWSUB – подстрока, на которую заменяем по умолчанию.

Примеры

```
SELECT SUBSTR('ABCDEF',2,3) FROM DUAL
```

– BCD.

Выбрать все имена из MAN, которые начинаются с «Ан».

```
select firstname from man where substr(firstname,1,2)='Ан'
```

Results

Explain

Describe

Saved SQL

History

FIRSTNAME

Андрей

Андрей

Замена подстроки в строке

REPLACE

REPLACE (SRCSTR, OLDSUB, NEWSUB) – функция, которая возвращает преобразованную строку SRCSTR, где подстрока OLDSUB из строки SRCSTR заменяется на подстроку

NEWSUB

SRCSTR – оригинальная строка.

OLDSUB – заменяемая подстрока.

NEWSUB – подстрока, на которую заменяем, по умолчанию NULL.

Заменить в имени в таблице MAN все буквы а на #.

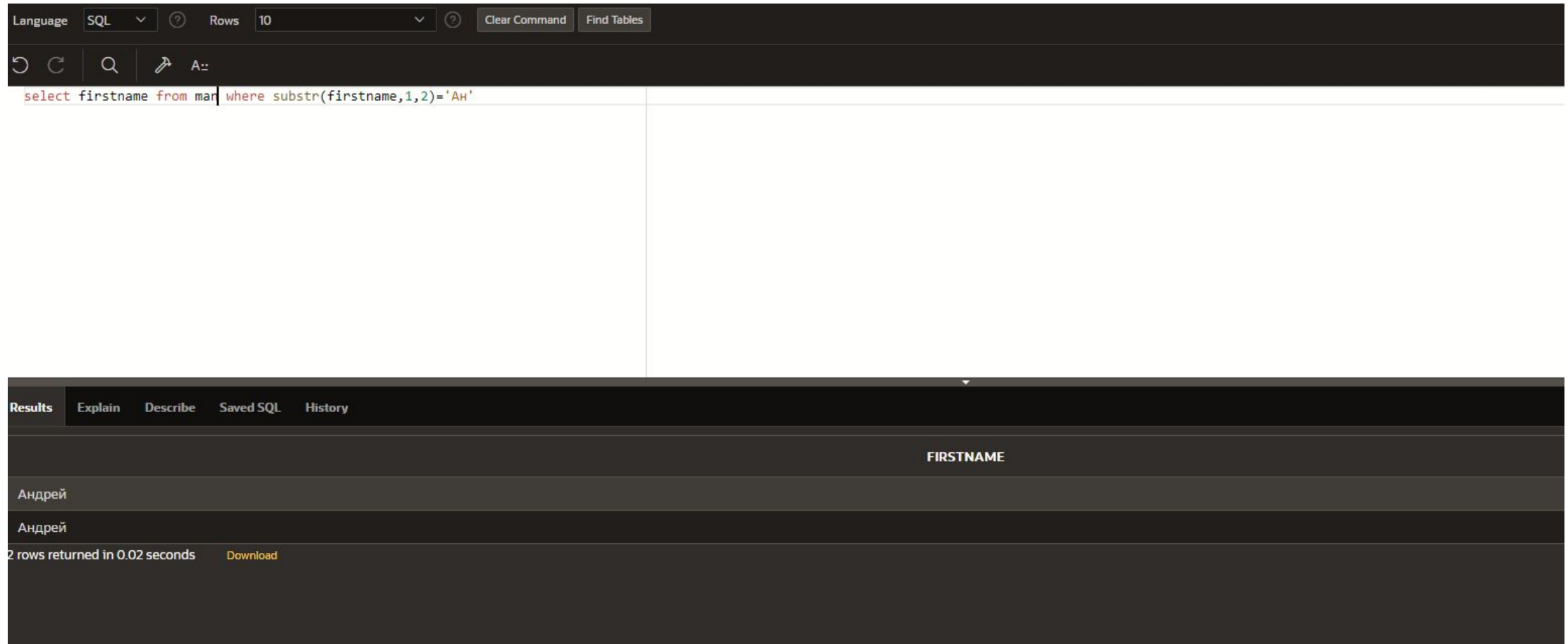
```
select firstname, replace(firstname,'a','#') as fn from man
```

FIRSTNAME	FN
Максим	М#ксим
Олег	Олег
Максим	М#ксим
Андрей	Андрей
Костя	Костя
Алексей	Алексей
Максим	М#ксим
Алиса	Алис#
Андрей	Андрей

Контрольные вопросы и задания для самостоятельного выполнения

-
1. Найти записи из таблицы **MAN**, начинающиеся на **Ан**, использовать **SUBSTR**.
 2. Найти записи из таблицы **MAN**, где количество символов в фамилии человека больше 10.
 3. Вывести из таблицы **MAN** имя, фамилию, количество символов в фамилии, где количество символов в фамилии человека больше 5.
 4. Заменить буквы «о» в слове «молоко» на @, вывести результат, использовать **REPLACE** и **DUAL**.
 5. Подсчитать количество букв «о» в слове «молоко», вывести результат, использовать **REPLACE**, **DUAL** и математику.
 6. Найти первое вхождение буквы «о» в городе с кодом 1 из таблицы **CITY**.
 7. Найти первое и последнее вхождения буквы «и» в городе с кодом 2 из таблицы **CITY**.

1. Найти записи из таблицы MAN, начинающиеся на Ан, использовать SUBSTR.



The screenshot shows a SQL IDE interface. At the top, there is a toolbar with 'Language SQL', 'Rows 10', 'Clear Command', and 'Find Tables'. Below the toolbar is a command area containing the SQL query: `select firstname from man where substr(firstname,1,2)='Ан'`. The results pane at the bottom shows a table with one column named 'FIRSTNAME' and two rows, both containing the name 'Андрей'. The status bar at the bottom of the results pane indicates '2 rows returned in 0.02 seconds' and includes a 'Download' link.

```
select firstname from man where substr(firstname,1,2)='Ан'
```

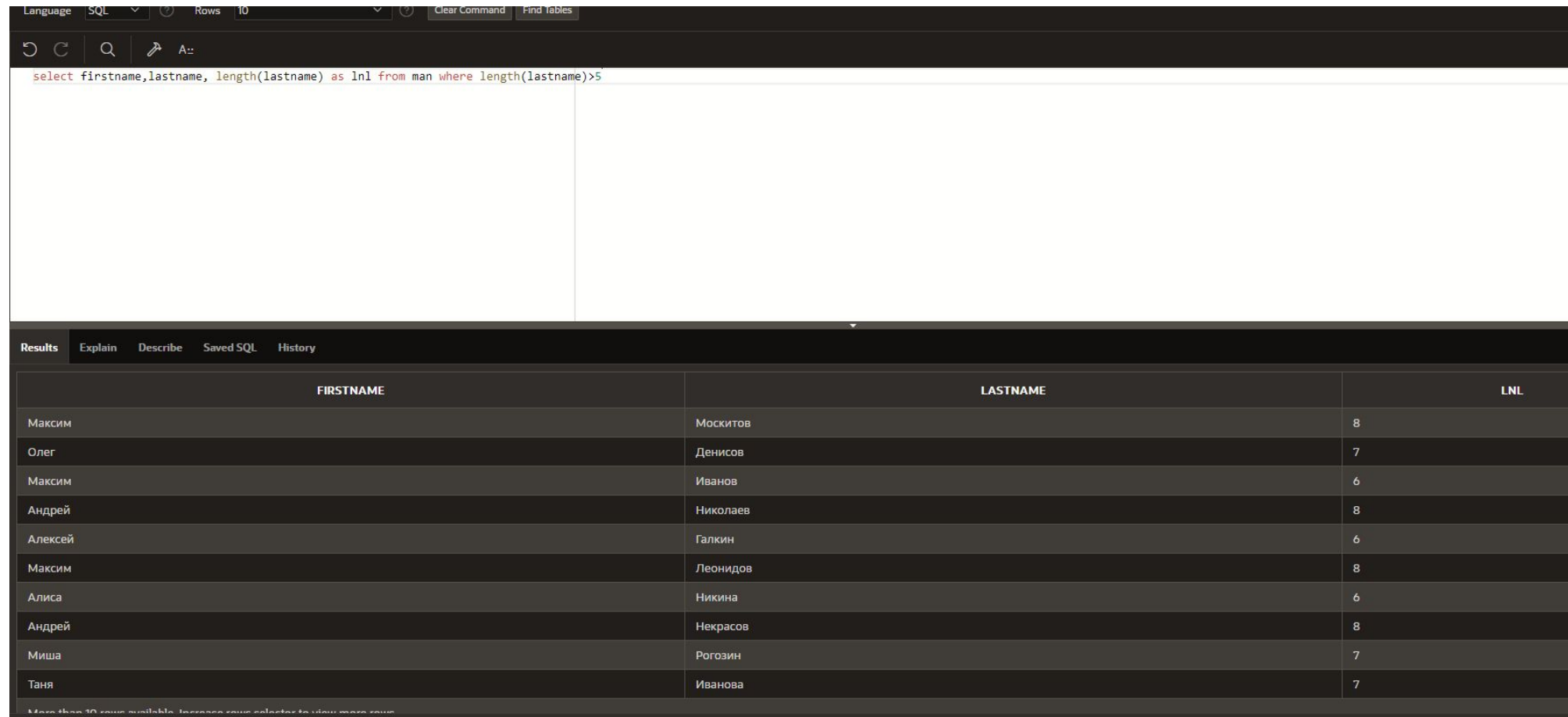
FIRSTNAME
Андрей
Андрей

2 rows returned in 0.02 seconds [Download](#)

КОЛИЧЕСТВО СИМВОЛОВ В ФАМИЛИИ ЧЕЛОВЕКА БОЛЬШЕ 10.

The image shows a screenshot of a SQL query editor interface. At the top, there is a toolbar with the following elements: 'Language' set to 'SQL', 'Rows' set to '10', 'Clear Command', and 'Find Tables'. Below the toolbar is a command input area containing the SQL query: `select * from man where length(lastname)>10`. The main area of the editor is empty, indicating that no data was found for the query. At the bottom, there is a 'Results' section with tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is active, and it displays the text 'no data found'.

3. Вывести из таблицы MAN имя, фамилию, количество символов в фамилии, где количество символов в фамилии человека больше 5.



The screenshot shows a SQL client interface with a dark theme. At the top, there's a toolbar with 'Language SQL', 'Rows 10', 'Clear Command', and 'Find Tables'. Below that is a search bar with 'A:'. The main area contains a SQL query: `select firstname, lastname, length(lastname) as lnl from man where length(lastname)>5`. Below the query is a 'Results' section with tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is active, showing a table with three columns: 'FIRSTNAME', 'LASTNAME', and 'LNL'. The table contains 11 rows of data. At the bottom, there is a message: 'More than 10 rows available. Press any column to view more rows.'

FIRSTNAME	LASTNAME	LNL
Максим	Москитов	8
Олег	Денисов	7
Максим	Иванов	6
Андрей	Николаев	8
Алексей	Галкин	6
Максим	Леонидов	8
Алиса	Никина	6
Андрей	Некрасов	8
Миша	Рогозин	7
Таня	Иванова	7

4. Заменить буквы «о» в слове «МОЛОКО» на @, вывести результат, использовать REPLACE и DUAL.

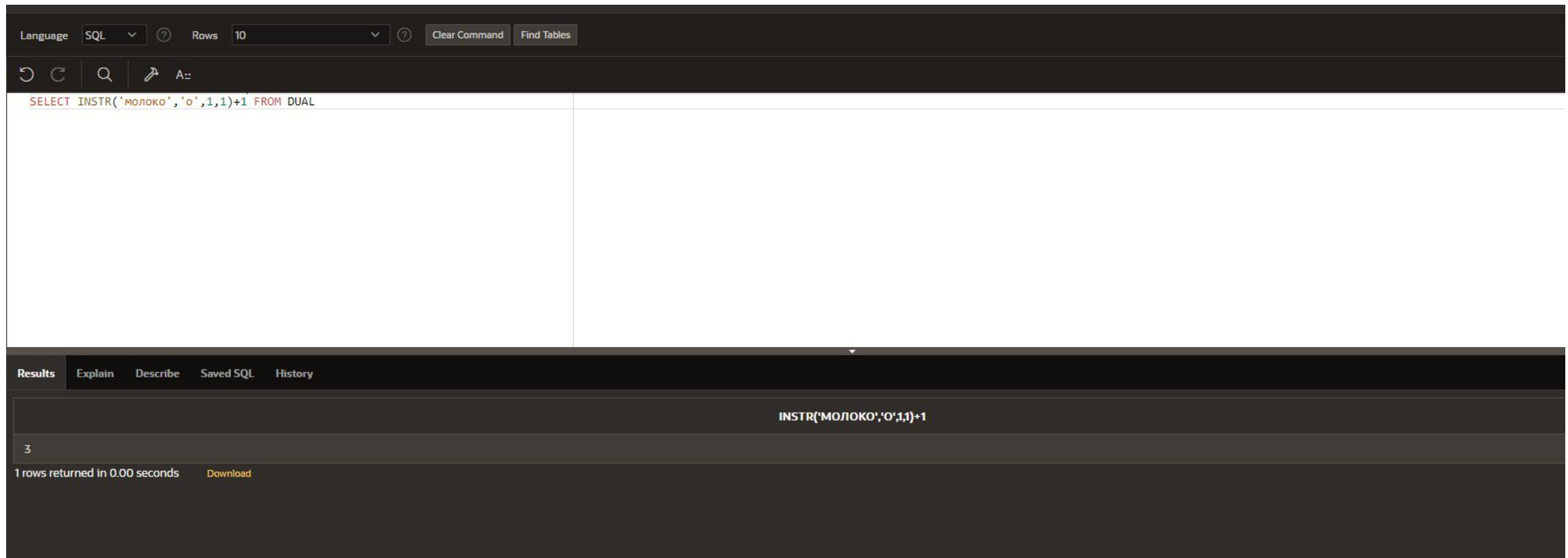
The image shows a SQL query editor and its results. The query entered is `select replace ('молоко', 'о', '@') From dual`. The results window shows the output of the `REPLACE('МОЛОКО';'О';'@')` function, which is `М@л@к@`. The results window also indicates that 1 row was returned in 0.00 seconds.

```
select replace ('молоко', 'о', '@') From dual
```

REPLACE('МОЛОКО';'О';'@')
М@л@к@

1 rows returned in 0.00 seconds [Download](#)

СЛОВЕ «МОЛОКО», ВЫВЕСТИ РЕЗУЛЬТАТ,
ИСПОЛЬЗОВАТЬ
REPLACE, DUAL И МАТЕМАТИКУ.



The screenshot shows a SQL query editor interface. At the top, there is a toolbar with 'Language' set to 'SQL', 'Rows' set to '10', and buttons for 'Clear Command' and 'Find Tables'. Below the toolbar is a search bar with a magnifying glass icon and the text 'A:'. The main area contains the SQL query: `SELECT INSTR('молоко', 'о', 1, 1) + 1 FROM DUAL`. Below the query is a results pane with tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is active, showing a single row with the value '3'. The column header is `INSTR('МОЛОКО','О',1,1)+1`. At the bottom of the results pane, it says '1 rows returned in 0.00 seconds' and has a 'Download' link.

```
SELECT INSTR('молоко', 'о', 1, 1) + 1 FROM DUAL
```

INSTR('МОЛОКО','О',1,1)+1
3

1 rows returned in 0.00 seconds [Download](#)

6. Найти первое вхождение буквы «О» в городе с кодом 1 из таблицы CITY.

The screenshot shows a SQL IDE interface. At the top, the 'SQL Commands' panel displays the following query:

```
SELECT cityname,CITYCODE, instr(cityname,'o', 2) lname from city where CITYCODE=1
```

Below the query editor, the 'Results' panel shows the output of the query. The results are displayed in a table with three columns: CITYNAME, CITYCODE, and LNAME. The first row contains the city name 'Москва', the city code '1', and the position of the first 'o' in the city name, which is '2'. Below the table, it indicates that 1 row was returned in 0.00 seconds.

CITYNAME	CITYCODE	LNAME
Москва	1	2

1 rows returned in 0.00 seconds [Download](#)

7. Найти первое и последнее вхождения буквы «и» в городе с кодом 2 из таблицы CITY.

The screenshot shows a SQL IDE interface. At the top, the 'SQL Commands' panel displays the following query:

```
SELECT cityname,CITYCODE, instr(cityname,'и', 1,2) lname from city where CITYCODE=2
```

Below the query, the 'Results' panel shows a table with the following data:

CITYNAME	CITYCODE	LNAME
Владимир	2	7

1 rows returned in 0.01 seconds [Download](#)

Математика и пустые
значения в запросах.

Случайность –

RANDOM

Теория и практика

Для математических выражений используются следующие операции:

- + сложение,
- вычитание,
- / деление,
- * умножение.

А также знакомые нам со школы функции:

`sqrt` – квадратный корень,

`mod`– остаток от деления,

`trunc` – округление до целого

Вывести из таблицы MAN имя, фамилию и возраст (FIRSTNAME, LASTNAME, YEAROLD) человека, разделенный на 10.

```
select firstname, lastname, yearold/10 as y10 from man
```

FIRSTNAME	LASTNAME	
Максим	Москитов	3.5
Олег	Денисов	4
Максим	Иванов	3.5
Андрей	Николаев	3.3
Костя	Пупкин	3.9
Алексей	Галкин	3.1
Максим	Леонидов	3.5
Алиса	Никина	7.8
Андрей	Некрасов	3.3

Вывести из таблицы MAN имя, фамилию и возраст человека (FIRSTNAME, LASTNAME, YEAROLD), умноженный на $\sin(1)$, округлить до целого.

```
select firstname, lastname, trunc(yearold*sin(1)) as ysin from man
```

Results Explain Describe Saved SQL History

FIRSTNAME	LASTNAME	YSIN
Максим	Москидов	29
Олег	Денисов	33
Максим	Иванов	29
Андрей	Николаев	27
Костя	Пупкин	32
Алексей	Галкин	26
Максим	Леонидов	29
Алиса	Никина	65
Андрей	Некрасов	27
Миня	Василин	70

Математика и пустые значения NULL

Если в математическом выражении используется пустое значение NULL, тогда значение любого математического выражения также будет NULL.

Например

– $10 + \text{NULL} = \text{NULL}$;

– $11 * \text{NULL} + 52 + \sin(1) = \text{NULL}$.

Эту особенность следует учитывать при построении запросов.

Генерация случайных чисел

SQL ORACLE диалекта также позволяет генерировать случайные значения, для этого используется специальный встроенный пакет (набор функций и процедур) `dbms_random`.

Для генерации случайного числа используется специальная функция `Value`.

Функция `VALUE` возвращает случайное число, большее или равное 0 и меньшее 1, с 38 цифрами справа от десятичной части (38 знаков после запятой). Кроме того, вы можете получить случайное число x , где x больше или равно `LOW` и менее `HIGH`.

Синтаксис

DBMS_RANDOM.VALUE RETURN NUMBER

```
DBMS_RANDOM.VALUE( low IN NUMBER, high IN NUMBER) RETURN NUMBER;
```

Параметры:

LOW – наименьшее количество в диапазоне для генерации случайного числа. Номер,

который генерируется, может быть равен LOW;

HIGH – наибольшее число для генерации случайного числа. Номер, который генерируется, будет меньше, чем HIGH. Возвращаемое значение – NUMBER.

пример

-0,777585712081073.

```
select dbms_random.value from dual
```

- 11,3383710413575.

```
select dbms_random.value(10,15) from dual
```

- 3,67901998206503.

```
select dbms_random.value(3,5) from dual
```

Контрольные вопросы и задания для самостоятельного выполнения

-
1. Вывести из таблицы MAN имя, фамилию и квадратный корень из 133.
 2. Вывести из таблицы MAN имя, фамилию и возраст человека, умноженный на $\cos(5)$.
 3. Вывести из таблицы CITY записи (*), где популяция делится без остатка на 10 000.
 4. Вывести из таблицы CITY название города, квадратный корень от популяции, умноженный на 10, где значение кода города делится нацело на 5.

1. Вывести из таблицы MAN имя, фамилию и квадратный корень из 133.

```
SELECT firstname, lastname, sqrt(133) as coren from man
```

FIRSTNAME	LASTNAME	COREN
Максим	Москитов	11.5325625946707958893541832388178725006
Олег	Денисов	11.5325625946707958893541832388178725006
Максим	Иванов	11.5325625946707958893541832388178725006
Андрей	Николаев	11.5325625946707958893541832388178725006
Алексей	Галкин	11.5325625946707958893541832388178725006
Максим	Леонидов	11.5325625946707958893541832388178725006
Алиса	Никина	11.5325625946707958893541832388178725006
Андрей	Некрасов	11.5325625946707958893541832388178725006
Миша	Рогозин	11.5325625946707958893541832388178725006
Таня	Иванова	11.5325625946707958893541832388178725006

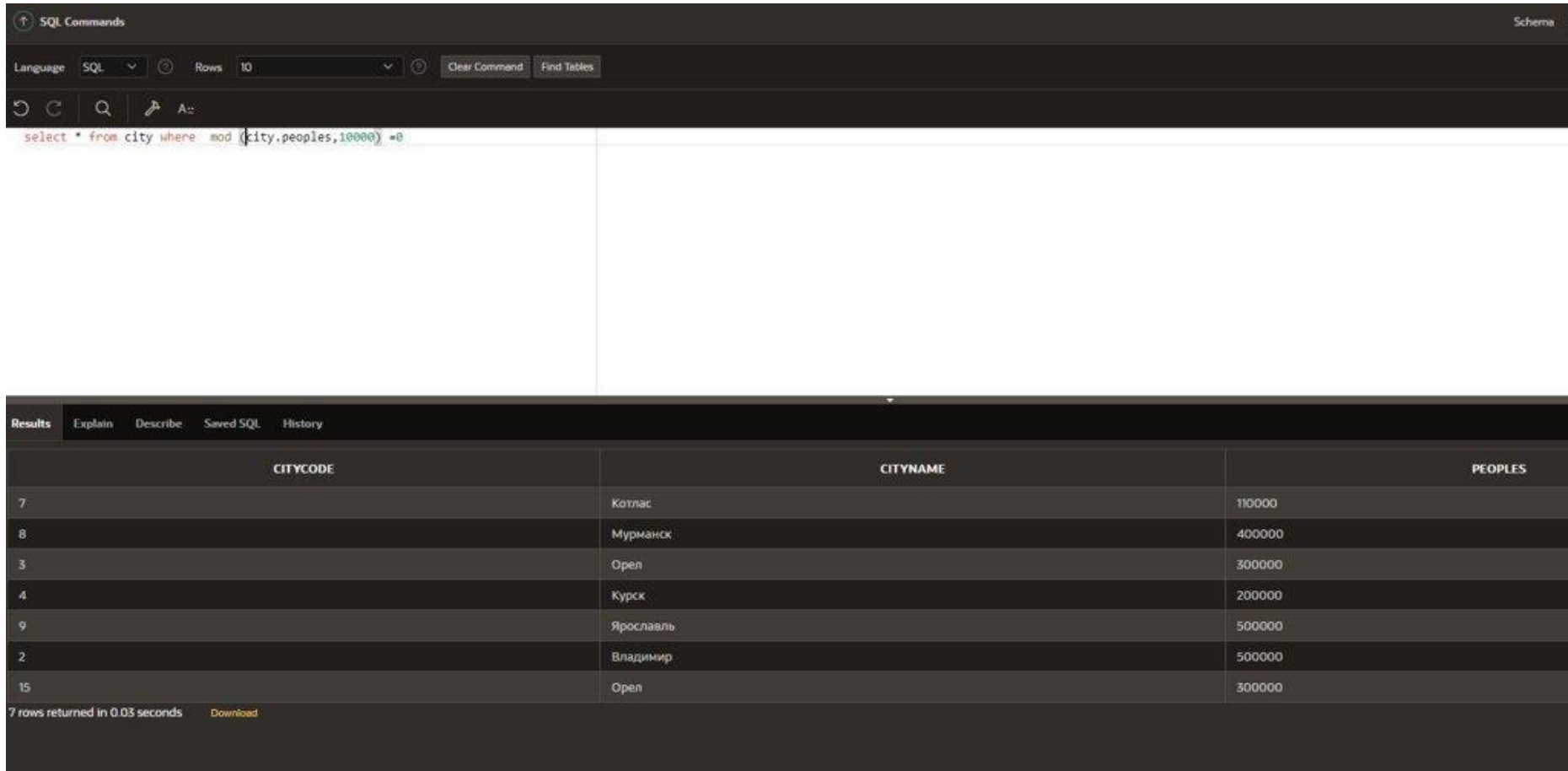
2. Вывести из таблицы MAN имя, фамилию и возраст человека, умноженный на cos (5).

```
select firstname,lastname,yearold*cos(5) as ycos from man
```

Results Explain Describe Saved SQL History

FIRSTNAME	LASTNAME	YCOS
Максим	Москитов	9.07718993482324046293245348843383386645
Олег	Денисов	14.1851092751613132233319585756778654163
Максим	Иванов	7.09155465658065661366597928783893270816
Андрей	Николаев	6.52423026565420408275270094481181809151
Алексей	Галкин	10.7791630476025980497322885175151777164
Алиса	Никона	8.79352774936001419846581431692027655812
Андрей	Некрасов	7.65887900750710914059925763086604732482
Миша	Рогозин	5.95690589472775155379942260178470347486
Таня	Иванова	8.79352774936001419846581431692027655812
Евгений	Леонидов	12.7647983458451819009987627181100788747

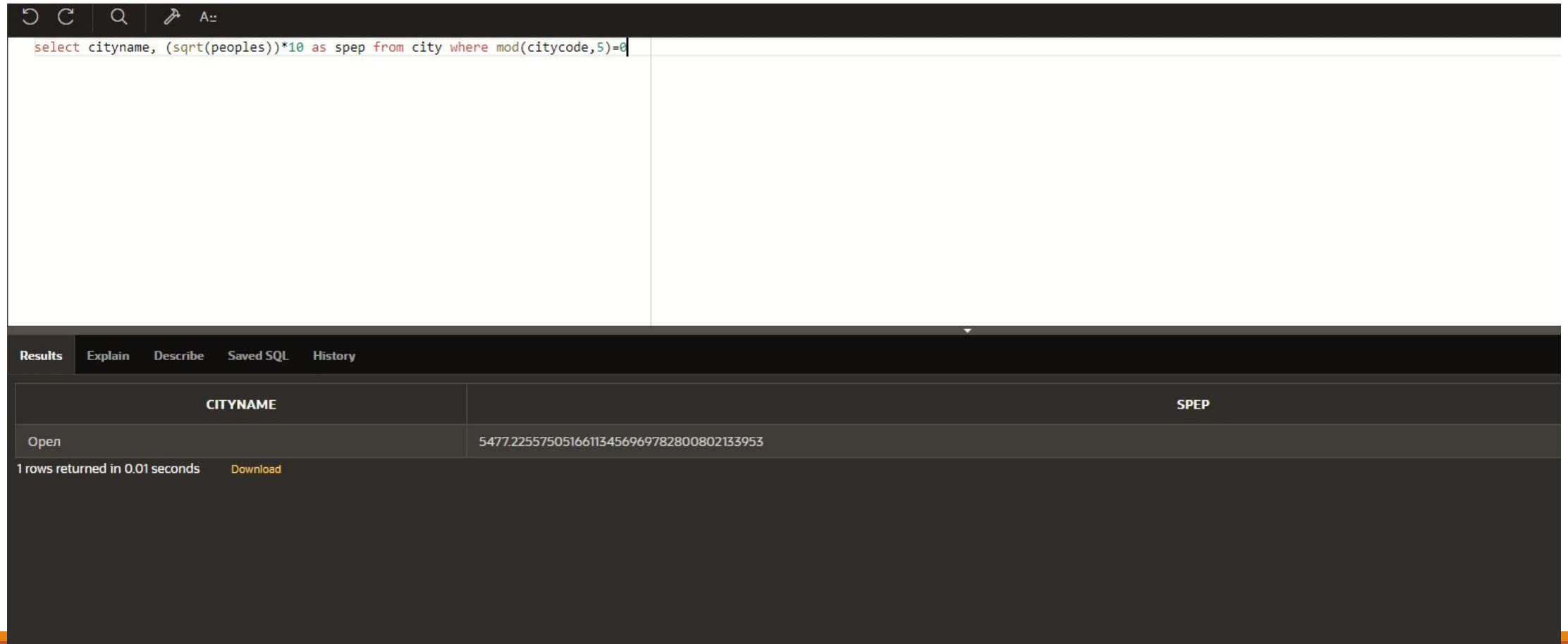
3. Вывести из таблицы CITY записи (*), где популяция делится без остатка на 10 000.



The screenshot shows a SQL client interface with a dark theme. At the top, the 'SQL Commands' window is active, displaying the query: `select * from city where mod(city.peoples,10000) = 0`. Below the query editor, the 'Results' window shows a table with three columns: CITYCODE, CITYNAME, and PEOPLES. The table contains seven rows of data. At the bottom of the results window, it indicates '7 rows returned in 0.03 seconds' and provides a 'Download' link.

CITYCODE	CITYNAME	PEOPLES
7	Котлас	110000
8	Мурманск	400000
3	Орел	300000
4	Курск	200000
9	Ярославль	500000
2	Владимир	500000
15	Орел	300000

4. Вывести из таблицы CITY название города, квадратный корень от популяции, умноженный на 10, где значение кода города делится нацело на 5.



The screenshot shows a SQL query execution interface. The query entered is: `select cityname, (sqrt(peoples))*10 as spep from city where mod(citycode,5)=0`. The results table shows one row for the city 'Орел' with a value of 5477.22557505166113456969782800802133953 for the 'SPEP' column.

```
select cityname, (sqrt(peoples))*10 as spep from city where mod(citycode,5)=0
```

CITYNAME	SPEP
Орел	5477.22557505166113456969782800802133953

1 rows returned in 0.01 seconds [Download](#)

Оператор IN

Теория и практика

Для удобной фильтрации выборки по списку значений в SQL существует специальный

оператор IN.

Он позволяет сравнить значение заданного поля со списком значений и выбирать данные

по результатам сравнения.

Синтаксис

```
SELECT перечень полей или * FROM таблица WHERE  
поле IN (значение1, значение2, значение )
```

фамилии людей (FIRSTNAME,
LASTNAME), которым 22, 31, 34, 27 лет
(YEAROLD).

```
select firstname, lastname from man where yearold in (22,31,34,27)
```

FIRSTNAME	LASTNAME
Алексей	Галкин

1 rows returned in 0.00 seconds [Download](#)

кодами 3, 5, 7 (CITYCODE), где население (PEOPLES) больше 100 000 человек.

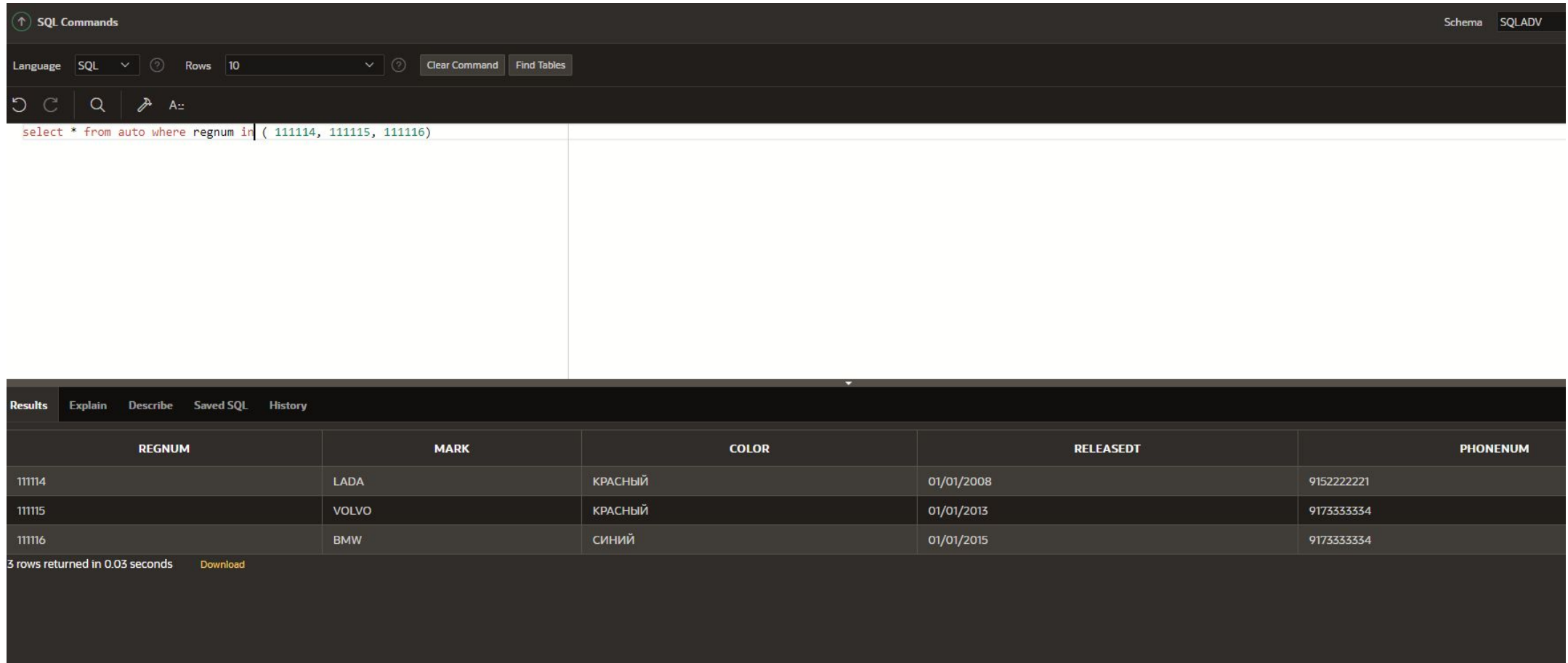
```
select * from city where citycode in (3,5,7) and peoples > 10000
```

CITYCODE	CITYNAME	PEOPLES
3	Орел	300000
7	Котлас	110000

Контрольные вопросы и задания для самостоятельного выполнения

-
1. Выбрать из таблицы AUTO, где REGNUM – 111114, 111115, 111116.
 2. Выбрать из таблицы MAN * людей с именами (FIRSTNAME) Андрей, Максим, Алиса.
 3. Выбрать из таблицы CITY * города (CITYNAME) Москва, Владимир, Казань.
 4. Выбрать из таблицы CITY * города с кодами (CITYCODE) 1, 3, 5, 7.

1. Выбрать из таблицы AUTO, где REGNUM – 111114, 111115, 111116.



The screenshot shows a SQL IDE interface. At the top, the title bar reads "SQL Commands" and "Schema SQLADV". Below the title bar, there are controls for "Language SQL", "Rows 10", and buttons for "Clear Command" and "Find Tables". The main area contains the SQL query: `select * from auto where regnum in (111114, 111115, 111116)`. Below the query, the results are displayed in a table with the following columns: REGNUM, MARK, COLOR, RELEASEDT, and PHONENUM. The results table contains three rows of data. At the bottom of the results area, it says "3 rows returned in 0.03 seconds" and has a "Download" link.

REGNUM	MARK	COLOR	RELEASEDT	PHONENUM
111114	LADA	КРАСНЫЙ	01/01/2008	915222221
111115	VOLVO	КРАСНЫЙ	01/01/2013	917333334
111116	BMW	СИНИЙ	01/01/2015	917333334

2. Выбрать из таблицы MAN * людей с именами (FIRSTNAME) Андрей, Максим, Алиса.

```
select * from man where firstname in ('Андрей', 'Максим', 'Алиса')
```

PHONENUM	FIRSTNAME	LASTNAME	CITYCODE	YEAROLD
9152222222*	Максим	Москитов	1	32
9213333336	Максим	Иванов	7	25
9152222221*	Андрей	Николаев	1	23
212121231	Максим	Леонидов	2	25
9173333334	Алиса	Никина	4	31
9213333331	Андрей	Некрасов	2	27

6 rows returned in 0.01 seconds [Download](#)

3. Выбрать из таблицы CITY * города (CITYNAME) Москва, Владимир, Казань.

```
select * from city where cityname in ('Казань', 'Владимир', 'Москва')
```

CITYCODE	CITYNAME	PEOPLES
16	Казань	2000010
1	Москва	10000010
2	Владимир	500000

3 rows returned in 0.01 seconds [Download](#)

4. Выбрать из таблицы CITY * города с кодами (CITYCODE) 1, 3, 5, 7.

```
select * from city where citycode in (1,3,5,7)
```

Results Explain Describe Saved SQL History

CITYCODE	CITYNAME	PEOPLES
1	Москва	1000010
3	Орел	300000
7	Котлас	110000

3 rows returned in 0.01 seconds [Download](#)

Объединение нескольких таблиц в запросе

Теория и практика

Давайте рассмотрим записи из таблицы города CITY, записи из таблицы MAN нашей схемы.

Мы видим, что и в одной, и в другой таблице есть колонка «код города» (CITYCODE).

Если посмотреть значения этой колонки (CITYCODE) и в той, и в другой таблице, то мы

увидим, что числа, значения в этих колонках совпадают.

Например, в MAN есть записи с CITYCODE = 1 и в CITY есть записи CITYCODE = 1,

то есть эта колонка является колонкой связи для таблиц CITY и MAN. По этим значениям

мы можем выбрать данные из указанных таблиц, поэтому, используя эту колонку, мы можем извлечь данные из обеих таблиц.

Первый вариант синтаксиса

```
select перечень полей или * from табл1, табл2  
where табл1.kod = табл2.kod
```

Запомните, как объединяются таблицы в нашей схеме:

AUTO – > MAN = PHONENUM

CITY – > MAN = CITYCODE

Правое и левое объединение таблиц

Теория и практика

Если внимательно присмотреться, то можно заметить, что в таблице CITY есть коды городов, которых нет в таблице MAN.

Точно так же и в таблице MAN есть номера телефонов, которых нет в таблице AUTO.

А что если нам необходимо выбрать из таблицы CITY все записи, а из таблицы MAN только те записи, которые совпадают с таблицей CITY по коду города (CITYCODE)?

Разумеется, в запросе, который объединяет обе эти таблицы.

Для этого в SQL ORACLE диалекта предусмотрен синтаксис правого и левого объединения таблиц, или RIGHT JOIN и LEFT JOIN.

Синтаксис LEFT JOIN

SELECT – перечень полей или * FROM – таблица, из которой мы извлекаем все записи;

LEFT JOIN – таблица, где мы извлекаем только совпадающие записи; on – условие объединения

ON (т1.код=т2.код).

Синтаксис RIGHT JOIN

SELECT перечень полей или * FROM – мы извлекаем только совпадающие записи;

RIGHT JOIN – таблица, из которой мы извлекаем все записи ON (т1.код=т2.код).

Итак, если мы используем правое объединение RIGHT JOIN, из правой таблицы от конструкции JOIN будут выбраны все записи, а из левой таблицы только совпадающие записи.

Если мы используем левое соединение LEFT JOIN, из левой таблицы от конструкции JOIN будут выбраны все записи, а из правой таблицы будут выбраны совпадающие записи.

Выбрать все записи из MAN и ТОЛЬКО совпадающие из AUTO.

```
select * from man m left join auto a on m.phonenum=a.phonenum
```

PHONENUM	FIRSTNAME	LASTNAME	CITYCODE	YEAROLD	REGNUM	MARK	COLOR	RELEASEDT	PHONE
915222221	Андрей	Николаев	1	33	11114	LADA	КРАСНЫЙ	01/01/2008	915222221
917333334	Алиса	Никина	4	78	11115	VOLVO	КРАСНЫЙ	01/01/2013	917333334
917333334	Алиса	Никина	4	78	11116	BMW	СИНИЙ	01/01/2015	917333334
921444444	Алексей	Галкин	1	31	111131	BMW	ЗЕЛЕНый	01/01/2007	921444444
921444444	Алексей	Галкин	1	31	11113	BMW	ЗЕЛЕНый	01/01/2007	921444444
921333336	Максим	Иванов	7	35	111122	AUDI	СИНИЙ	01/01/2011	921333336
921333331	Андрей	Некрасов	2	33	11119	LADA	СИНИЙ	01/01/2017	921333331
9789999901	Евгений	Леонидов	2	78	-	-	-	-	-
915222222	Максим	Москитов	1	35	-	-	-	-	-

CITYCODE, выведем значения всех колонок из этих таблиц

```
select * from man m, city c where c.citycode=m.citycode
```

PHONENUM	FIRSTNAME	LASTNAME	CITYCODE	YEAROLD	CITYCODE	CITYNAME	PEOPLES
9152222222	Максим	Москитов	1	35	1	Москва	1000010
9153333333	Олег	Денисов	3	40	3	Орел	300000
9213333336	Максим	Иванов	7	35	7	Котлас	110000
9152222221	Андрей	Николаев	1	33	1	Москва	1000010
56789876543	Костя	Пупкин	3	39	3	Орел	300000
9214444444	Алексей	Галкин	1	31	1	Москва	1000010
212121231	Максим	Леонидов	2	35	2	Владимир	500000
9173333334	Алиса	Никина	4	78	4	Курск	200000
9213333331	Андрей	Некрасов	2	33	2	Владимир	500000

Объединим MAN и CITY по колонке CITYCODE, выведем наименование и население (CITYNAME, PEOPLES) из таблицы CITY и имя и фамилию из таблицы MAN (FIRSTNAME, LASTNAME).

```
select m.firstname, m.lastname, c.cityname, c.peoples from man m, city c where c.citycode=m.citycode
```

FIRSTNAME	LASTNAME	CITYNAME	PEOPLES
Максим	Москитов	Москва	1000010
Олег	Денисов	Орел	300000
Максим	Иванов	Котлас	110000
Андрей	Николаев	Москва	1000010
Костя	Пупкин	Орел	300000
Алексей	Галкин	Москва	1000010
Максим	Леонидов	Владимир	500000
Алиса	Никина	Курск	200000
Андрей	Некрасов	Владимир	500000
Михаил	Ворожанин	Владимир	500000

Контрольные вопросы и задания для самостоятельного выполнения

1. Можно ли сделать объединение таблиц по нескольким полям?
2. Вывести все значения из таблицы MAN, из таблицы AUTO, объединить эти две таблицы посредством поля «номер телефона» (PHONENUM).
3. Вывести все значения из таблицы MAN, из таблицы CITY, объединить таблицы по коду города.
4. Вывести имя и фамилию из таблицы MAN, а также марку из таблицы AUTO, объединить две таблицы по номеру телефона (PHONENUM).

Контрольные вопросы и задания для самостоятельного выполнения

1. Повторите самостоятельно примеры из данного шага.
2. Используя RIGHT JOIN, выберите все записи из CITY и только совпадающие из MAN.
3. Используя LEFT JOIN, выберите все записи из CITY и только совпадающие из MAN.