

1

**Выбор данных с использованием команды
SELECT языка SQL**

Рассматриваемые вопросы

- Перечень возможностей команды SELECT языка SQL
- Выполнение базовой команды SELECT

Возможности команды SELECT языка SQL

Проекция

Таблица 1

Выбор

Таблица 1

Соединение

Таблица 1

Таблица 2

Базовая команда SELECT

```
SELECT * | { [DISTINCT] столбец | выражение  
[псевдоним] , ... }  
FROM таблица ;
```

- SELECT указывает, *какие* столбцы выводить;
- FROM указывает, из *какой* таблицы.

Выбор всех столбцов

```
SELECT *  
FROM departments;
```

	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
1	10	Administration	200	1700
2	20	Marketing	201	1800
3	50	Shipping	124	1500
4	60	IT	103	1400
5	80	Sales	149	2500
6	90	Executive	100	1700
7	110	Accounting	205	1700
8	190	Contracting	(null)	1700

Выбор конкретных столбцов

```
SELECT department_id, location_id  
FROM departments;
```

	DEPARTMENT_ID	LOCATION_ID
1	10	1700
2	20	1800
3	50	1500
4	60	1400
5	80	2500
6	90	1700
7	110	1700
8	190	1700

Написание команд SQL

- Команды SQL не различают регистры символов.
- Команды SQL могут занимать одну или несколько строк.
- Ключевые слова нельзя сокращать и размещать на двух строках.
- Предложения обычно пишутся на отдельных строках.
- Для облегчения чтения используются табуляция и отступы.
- В SQL Developer после команды SQL необязательно ставить точку с запятой (;). Однако при вводе нескольких команд SQL их необходимо завершать точкой с запятой.
- В SQL*plus каждая команда SQL должна заканчиваться точкой с запятой (;).

Заголовки столбцов по умолчанию

- SQL Developer:
 - Выравнивание по умолчанию: слева
 - Вывод по умолчанию: в символах верхнего регистра
- SQL*Plus:
 - Слева: даты и символьные данные
 - Справа: цифровые данные
 - Вывод по умолчанию: в символах верхнего регистра

Арифметические выражения

Создаются из данных типа NUMBER и DATE с помощью арифметических операторов.

Operator	Description
+	Add
-	Subtract
*	Multiply
/	Divide

Использование арифметических операторов

```
SELECT last_name, salary, salary + 300  
FROM employees;
```

	LAST_NAME	SALARY	SALARY+300
1	Whalen	4400	4700
2	Hartstein	13000	13300
3	Fay	6000	6300
4	Higgins	12000	12300
5	Gietz	8300	8600
6	King	24000	24300
7	Kochhar	17000	17300
8	De Haan	17000	17300
9	Hunold	9000	9300
10	Ernst	6000	6300

...

Приоритеты операторов

```
SELECT last_name, salary, 12*salary+100
FROM employees;
```

1

R2	LAST_NAME	R2	SALARY	R2	12*SALARY+100
1	Whalen		4400		52900
2	Hartstein		13000		156100
3	Fay		6000		72100

...

```
SELECT last_name, salary, 12*(salary+100)
FROM employees;
```

2

R2	LAST_NAME	R2	SALARY	R2	12*(SALARY+100)
1	Whalen		4400		54000
2	Hartstein		13000		157200
3	Fay		6000		73200

...

Неопределенное значение (NULL)

- Неопределенное значение (NULL) – это значение, которое недоступно, не присвоено, неизвестно или неприменимо.
- Это не ноль и не пробел.

```
SELECT last_name, job_id,  
salary, commission_pct  
FROM employees;
```



	LAST_NAME	JOB_ID	SALARY	COMMISSION_PCT
1	Whalen	AD_ASST	4400	(null)
2	Hartstein	MK_MAN	13000	(null)

• • •

17	Zlotkey	SA_MAN	10500	0.2
18	Abel	SA_REP	11000	0.3
19	Taylor	SA_REP	8600	0.2
20	Grant	SA_REP	7000	0.15

Неопределенные значения в арифметических выражениях

Результат вычисления выражения, содержащего неопределенное значение, также будет неопределенным.

```
SELECT last_name, 12*salary*commission_pct  
FROM employees;
```

	LAST_NAME	12*SALARY*COMMISSION_PCT
1	Whalen	(null)
2	Hartstein	(null)
3	Fay	(null)

...

17	Zlotkey	25200
18	Abel	39600
19	Taylor	20640
20	Grant	12600

Определение псевдонимов столбцов

Псевдонимы столбцов:

- Переименовывают заголовки столбцов.
- Удобны при вычислениях.
- Следуют сразу за именем столбца; ключевое слово AS между именем столбца и псевдонимом необязательно.
- Заключаются в двойные кавычки, если содержат пробелы, специальные символы или необходимо различать регистры символов.

Использование псевдонимов столбцов

```
SELECT last_name AS name,  
       commission_pct comm  
FROM   employees;
```

	NAME	COMM
1	Whalen	(null)
2	Hartstein	(null)
3	Fay	(null)

...

```
SELECT last_name "Name", salary*12 "Annual  
Salary"  
FROM   employees;
```

	Name	Annual Salary
1	Whalen	52800
2	Hartstein	156000
3	Fay	72000

...

Оператор конкатенации

- Соединяет столбцы или символьные строки с другими столбцами.
- Изображается двумя вертикальными линиями (||).
- Создает столбец с результатом, представляющим символьное выражение

```
SELECT last_name||job_id AS "Employees"  
FROM employees;
```

	Employees
1	AbelSA_REP
2	DaviesST_CLERK
3	De HaanAD_VP
4	ErnstIT_PROG
5	FayMK_REP
6	GietzAC_ACCOUNT

...

Символьные литералы

- Литерал – это символ, число или дата, включенные в SELECT список.
- Даты и символьные литералы должны быть заключены в апострофы.
- Каждая символьная строка выводится один раз для каждой возвращаемой строки таблицы.

Использование символьных литералов

```
SELECT last_name || ' is a ' || job_id  
       AS "Employee Details"  
FROM   employees;
```

	Employee Details
1	Abel is a SA_REP
2	Davies is a ST_CLERK
3	De Haan is a AD_VP
4	Ernst is a IT_PROG
5	Fay is a MK_REP
6	Gietz is a AC_ACCOUNT
7	Grant is a SA_REP
8	Hartstein is a MK_MAN
9	Higgins is a AC_MGR
10	Hunold is a IT_PROG

...

Оператор, задающий временно действующие кавычки (q)

- Задание собственного ограничителя в качестве кавычек
- Выбор ограничителя
- Улучшение читабельности и простота использования

```
SELECT department_name || q' [ Department's Manager Id: ] '  
      || manager_id  
      AS "Department and Manager"  
FROM departments;
```

	Department and Manager
1	Administration Department's Manager Id: 200
2	Marketing Department's Manager Id: 201
3	Shipping Department's Manager Id: 124
4	IT Department's Manager Id: 103
5	Sales Department's Manager Id: 149
6	Executive Department's Manager Id: 100
7	Accounting Department's Manager Id: 205
8	Contracting Department's Manager Id:

Дублирование строк

По умолчанию выдаются все строки, включая дубликаты.

1

```
SELECT department_id  
FROM employees;
```

	DEPARTMENT_ID
1	10
2	20
3	20
4	110
5	110

...

2

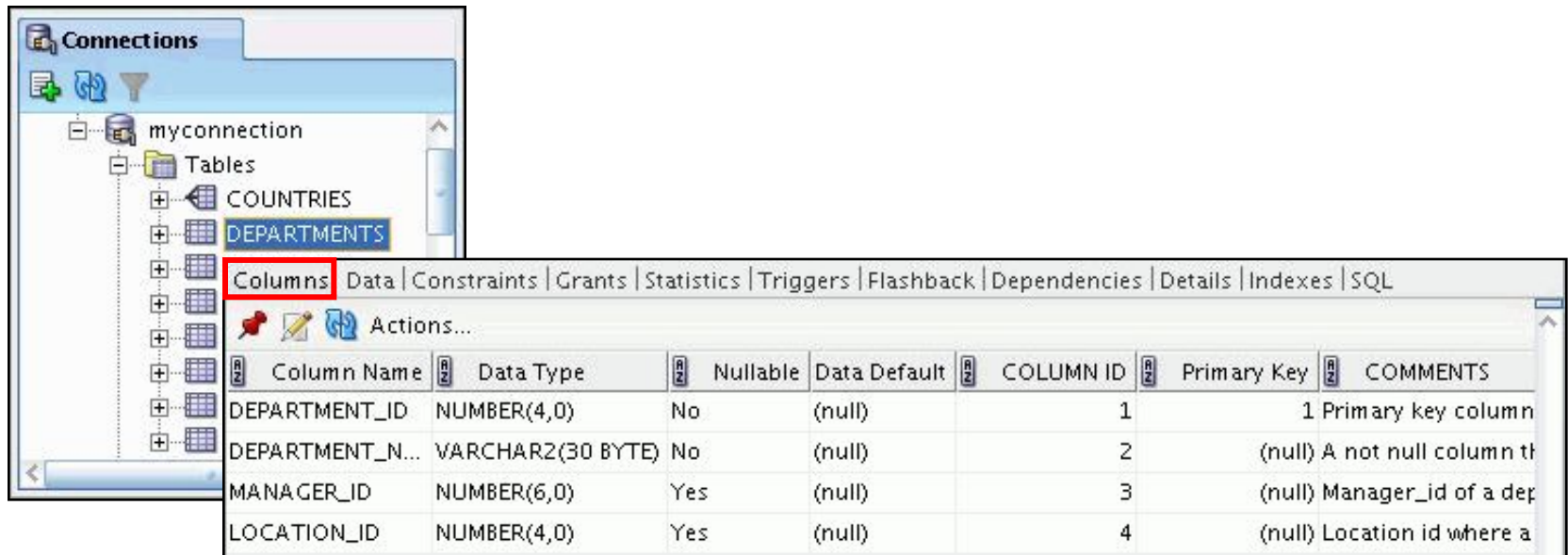
```
SELECT DISTINCT department_id  
FROM employees;
```

	DEPARTMENT_ID
1	(null)
2	20
3	90
4	110
5	50
6	80
7	10
8	60

Вывод структуры таблицы

- Вывод структуры таблицы производится с помощью команды `DESCRIBE`
- Или, выбрать таблицу в дереве «Connections» и использовать вкладку «Columns».

```
DESC[RIBE] tablename
```



The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' tree is expanded to show a connection named 'myconnection'. Underneath, the 'Tables' folder is expanded, and the 'DEPARTMENTS' table is selected. The 'Columns' tab is active, displaying the following table structure:

Column Name	Data Type	Nullable	Data Default	COLUMN ID	Primary Key	COMMENTS
DEPARTMENT_ID	NUMBER(4,0)	No	(null)	1	1	Primary key column
DEPARTMENT_N...	VARCHAR2(30 BYTE)	No	(null)	2		(null) A not null column th
MANAGER_ID	NUMBER(6,0)	Yes	(null)	3		(null) Manager_id of a dep
LOCATION_ID	NUMBER(4,0)	Yes	(null)	4		(null) Location id where a

Использование команды DESCRIBE

```
DESCRIBE employees
```

```
DESCRIBE employees
Name                               Null    Type
-----
EMPLOYEE_ID                        NOT NULL NUMBER(6)
FIRST_NAME                          VARCHA2(20)
LAST_NAME                           NOT NULL VARCHA2(25)
EMAIL                               NOT NULL VARCHA2(25)
PHONE_NUMBER                        VARCHA2(20)
HIRE_DATE                           NOT NULL DATE
JOB_ID                              NOT NULL VARCHA2(10)
SALARY                              NUMBER(8,2)
COMMISSION_PCT                      NUMBER(2,2)
MANAGER_ID                           NUMBER(6)
DEPARTMENT_ID                       NUMBER(4)

11 rows selected
```

Тест

Какие команды будут выполнены успешно?

1.

```
SELECT first_name, last_name, job_id, salary*12
AS Yearly Sal
FROM employees;
```

2.

```
SELECT first_name, last_name, job_id, salary*12
"yearly sal"
FROM employees;
```

3.

```
SELECT first_name, last_name, job_id, salary AS
"yearly sal"
FROM employees;
```

4.

```
SELECT first_name+last_name AS name, job_Id,
salary*12 yearly sal
FROM employees;
```

Итоги

В этом уроке вы познакомились с тем, как:

- Задавать команду `SELECT`, по которой:
 - возвращаются все строки и столбцы таблицы
 - возвращаются определенные столбцы таблицы
 - используются псевдонимы для изменения наименований заголовков столбцов
- Использовать среду `iSQL*Plus` для ввода, сохранения и выполнения команд `SQL` и `iSQL*Plus`

```
SELECT * | { [DISTINCT] столбец | выражение  
[псевдоним] , . . . }  
FROM   таблица ;
```


Обзор практического занятия 1

- Выборка данных из разных таблиц
- Описание структуры таблиц
- Выполнение арифметических вычислений и задание имен столбцов