

# Агрегирование с использованием групповых функций

# Цели

- Изучить функции группировки
- Научиться применять функции группировки
- Группировать информацию с использованием секции GROUP BY
- Ограничивать выборку сгруппированных строк секцией HAVING

# Функции группировки

- Функции группировки работают с набором данных для получения единственного результата

**EMPLOYEES**

DEPARTMENT_ID	SALARY
90	24000
90	17000
90	17000
60	9000
60	6000
60	4200
50	5800
50	3500
50	3100
50	2600
50	2500
80	10500
80	11000
80	8600
	7000
10	4400

Maximum salary in EMPLOYEES table

MAX(SALARY)
24000

20 rows selected.

# Функции группировки

- AVG
- COUNT
- MAX
- MIN
- STDDEV
- SUM
- VARIANCE



# Синтаксис

```
SELECT [column,] group_function(column), ...  
FROM table  
[WHERE condition]  
[GROUP BY column]  
[ORDER BY column];
```

# Функции AVG и SUM

```
SELECT AVG(salary), MAX(salary),  
MIN(salary), SUM(salary)  
FROM employees  
WHERE job_id LIKE '%REP%';
```

AVG(SALARY)	MAX(SALARY)	MIN(SALARY)	SUM(SALARY)
8150	11000	6000	32600

# Функции MIN и MAX

```
SELECT MIN(hire_date), MAX(hire_date)  
FROM employees;
```

MIN(HIRE_	MAX(HIRE_
17-JUN-87	29-JAN-00

# Функция COUNT

COUNT(\*) возвращает число строк в таблице:

```
SELECT COUNT(*)  
FROM employees  
WHERE department_id = 50;
```

COUNT(expr) возвращает число строк not null в таблице:

```
SELECT COUNT(commission_pct)  
FROM employees  
WHERE department_id = 80;
```

# Ключевое слово DISTINCT

- COUNT(DISTINCT expr) возвращает число уникальных not null строк
- Отображение количества уникальных подразделений в таблице EMPLOYEES

```
SELECT COUNT(DISTINCT department_id)
FROM employees;
```

COUNT(DISTINCTDEPARTMENT\_ID)

# Функции группировки и NULL значения

- Функции группировки игнорируют NULL значения

```
SELECT AVG(commission_pct) FROM employees;
```

AVG(COMMISSION\_PCT)

.2125

- Функция NVL заставляет функцию группировки использовать NULL значения

```
SELECT AVG(NVL(commission_pct, 0)) FROM employees;
```

AVG(NVL(COMMISSION\_PCT,0))

.0425

# Группировка данных

## EMPLOYEES

DEPARTMENT_ID	SALARY
10	4400
20	13000
20	6000
50	5800
50	3500
50	3100
50	2500
50	2600
60	9000
60	6000
60	4200
80	10500
80	8600
80	11000
90	24000
90	17000

4400

9500

3500

6400

10033

**Average  
salary in  
EMPLOYEES  
table for each  
department**

DEPARTMENT_ID	AVG(SALARY)
10	4400
20	9500
50	3500
60	6400
80	10033.3333
90	19333.3333
110	10150
	7000

...

20 rows selected.

# Секция GROUP BY

- Можно разделять строки в таблицах на группы используя секцию GROUP BY

```
SELECT column, group_function(column)
FROM table
[WHERE condition]
[GROUP BY group_by_expression]
[ORDER BY column];
```

# Секция GROUP BY

- Все колонки в секции SELECT, к которым не применяется группировка, должны быть в секции GROUP BY.

```
SELECT department_id, AVG(salary)
FROM employees
GROUP BY department_id ;
```

DEPARTMENT_ID	AVG(SALARY)
10	4400
20	9500
50	3500
60	6400
80	10033.3333
90	19333.3333
110	10150
	7000

8 rows selected.

# Секция GROUP BY

- Колонки, находящиеся в секции GROUP BY, не должны находиться в секции SELECT без функции группировки.

```
SELECT AVG(salary)
FROM employees
GROUP BY department_id ;
```

AVG(SALARY)
4400
9500
3500
6400
10033.3333
19333.3333
10150
7000

# Группировка по нескольким колонкам

**EMPLOYEES**

DEPARTMENT_ID	JOB_ID	SALARY
90	AD_PRES	24000
90	AD_VP	17000
90	AD_VP	17000
60	IT_PROG	9000
60	IT_PROG	6000
60	IT_PROG	4200
50	ST_MAN	5800
50	ST_CLERK	3500
50	ST_CLERK	3100
50	ST_CLERK	2600
50	ST_CLERK	2500
80	SA_MAN	10500
80	SA_REP	11000
80	SA_REP	8600
...		
20	MK_REP	6000
110	AC_MGR	12000
110	AC_ACCOUNT	8300

20 rows selected.

**Add the salaries in the EMPLOYEES table for each job, grouped by department**

DEPARTMENT_ID	JOB_ID	SUM(SALARY)
10	AD_ASST	4400
20	MK_MAN	13000
20	MK_REP	6000
50	ST_CLERK	11700
50	ST_MAN	5800
60	IT_PROG	19200
80	SA_MAN	10500
80	SA_REP	19600
90	AD_PRES	24000
90	AD_VP	34000
110	AC_ACCOUNT	8300
110	AC_MGR	12000
	SA_REP	7000

13 rows selected.

# Группировка по нескольким колонкам

```
SELECT department_id dept_id, job_id, SUM(salary)
FROM employees
GROUP BY department_id, job_id ;
```

DEPT_ID	JOB_ID	SUM(SALARY)
10	AD_ASST	4400
20	MK_MAN	13000
20	MK_REP	6000
50	ST_CLERK	11700
50	ST_MAN	5800
60	IT_PROG	19200
80	SA_MAN	10500
80	SA_REP	19600
90	AD_PRES	24000
90	AD_VP	34000
110	AC_ACCOUNT	8300
110	AC_MGR	12000
	SA_REP	7000

13 rows selected.

# Часто встречаемые ошибки

- Колонка, находящаяся в секции SELECT, не обернута групповой функцией и не находится в секции GROUP BY

```
SELECT department_id, COUNT(last_name)
FROM employees;
```

```
ERROR at line 1:
```

```
ORA-00937: not a single-group group function
```

# Часто встречаемые ошибки

- Нельзя использовать секцию WHERE для ограничения сгруппированных строк
- Используйте HAVING
- Функцию группировки нельзя использовать в секции WHERE.

```
SELECT department_id, AVG(salary)
FROM employees
WHERE AVG(salary) > 8000
GROUP BY department_id;
```

ERROR at line 3:

ORA-00934: group function is not allowed here

# Ограничение результатов группировки

## EMPLOYEES

DEPARTMENT_ID	SALARY
90	24000
90	17000
90	17000
60	9000
60	6000
60	4200
50	5800
50	3500
50	3100
50	2600
50	2500
80	10500
80	11000
80	8600
...	...
20	6000
110	12000
110	8300

20 rows selected.

The maximum salary per department when it is greater than \$10,000

DEPARTMENT_ID	MAX(SALARY)
20	13000
80	11000
90	24000
110	12000

# Секция HAVING

- При использовании секции HAVING сервер ORACLE выполняет следующие шаги:
  - Группирует строки
  - Применяет функцию группировки
  - Сравнивает группы с ограничением в секции HAVING

```
SELECT column, group_function  
FROM table  
[WHERE condition]  
[GROUP BY group_by_expression]  
[HAVING group_condition]  
[ORDER BY column];
```

# Секция HAVING

```
SELECT department_id, MAX(salary)
FROM employees
GROUP BY department_id
HAVING MAX(salary)>10000;
```

DEPARTMENT_ID	MAX(SALARY)
20	13000
80	11000
90	24000
110	12000

# Секция HAVING

```
SELECT job_id, SUM(salary) PAYROLL
FROM employees
WHERE job_id NOT LIKE '%REP%'
GROUP BY job_id
HAVING SUM(salary) > 13000
ORDER BY SUM(salary);
```

JOB_ID	PAYROLL
IT_PROG	19200
AD_PRES	24000
AD_VP	34000

# Вложенность

```
SELECT MAX(AVG(salary))  
FROM employees  
GROUP BY department_id;
```

MAX(AVG(SALARY))
19333.3333