Методы сбора, хранения, обработки и анализа данных

Лекция 3

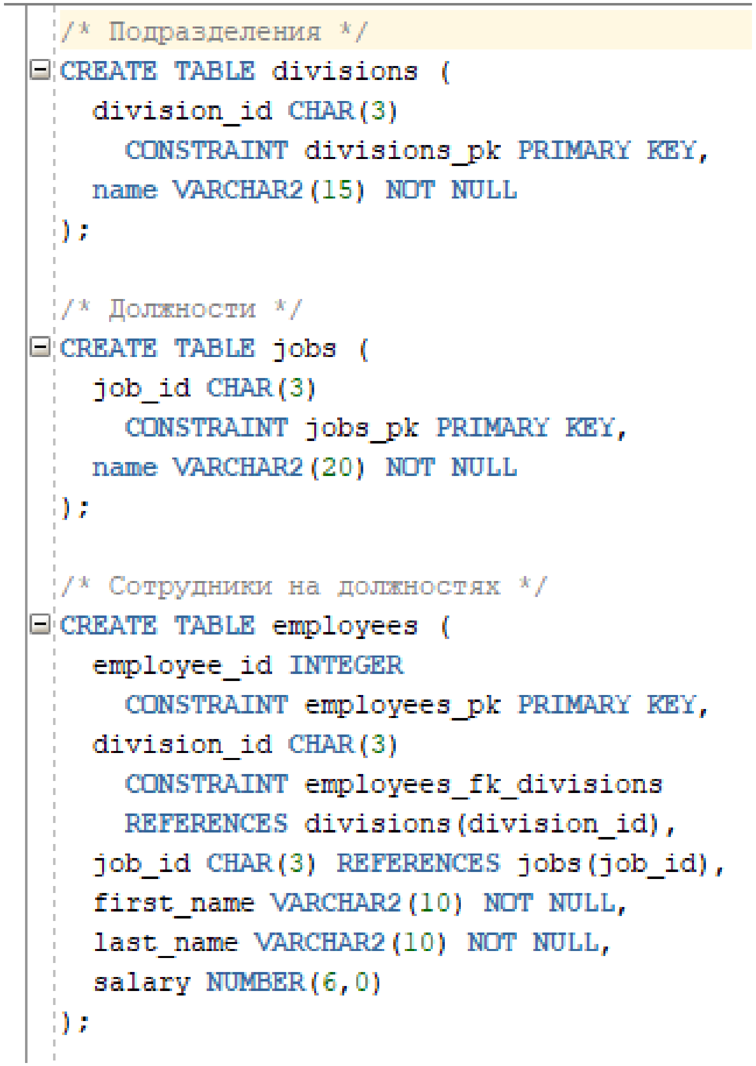
Расширенные группировки

РАСШИРЕННЫЕ ГРУППИРОВКИ

Расширенные группировки

- ROLLUP – возвращает строки итогов и предварительных итогов
- CUBE – предварительные итоги для каждой комбинации строк
- GROUPING – является ли агрегированным?
- GROUPING_ID – уровень группировки
- GROUPING SETS – наборы столбцов для группировки

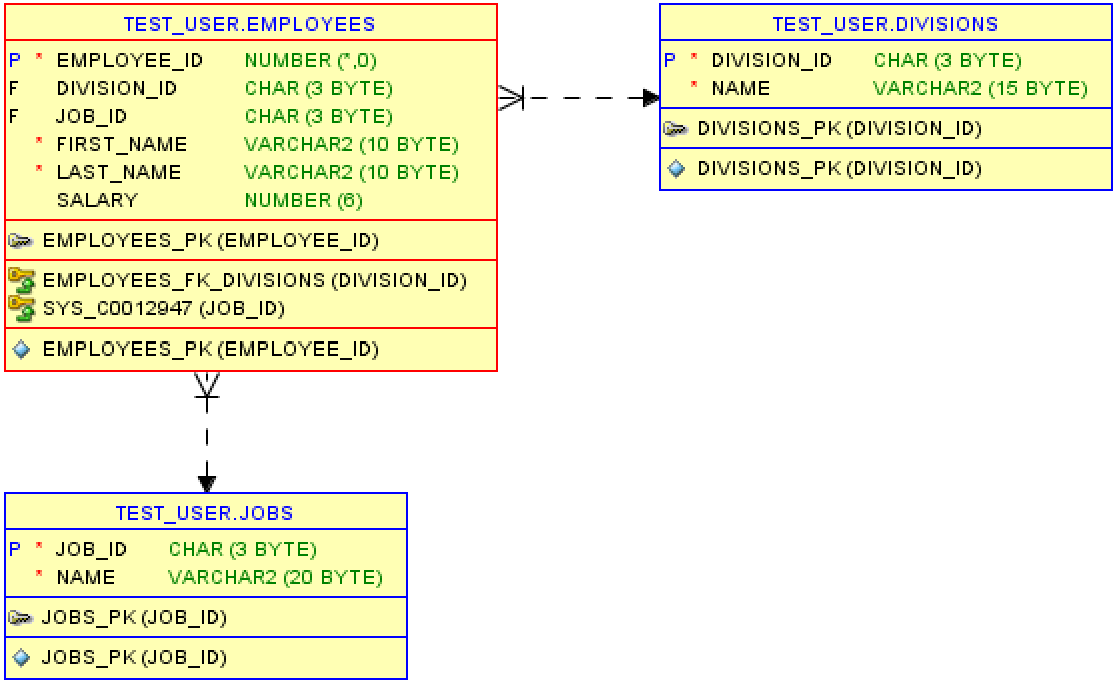
Таблицы



```
/* Подразделения */
CREATE TABLE divisions (
  division_id CHAR(3)
    CONSTRAINT divisions_pk PRIMARY KEY,
  name VARCHAR2(15) NOT NULL
);

/* Должности */
CREATE TABLE jobs (
  job_id CHAR(3)
    CONSTRAINT jobs_pk PRIMARY KEY,
  name VARCHAR2(20) NOT NULL
);

/* Сотрудники на должностях */
CREATE TABLE employees (
  employee_id INTEGER
    CONSTRAINT employees_pk PRIMARY KEY,
  division_id CHAR(3)
    CONSTRAINT employees_fk_divisions
    REFERENCES divisions(division_id),
  job_id CHAR(3) REFERENCES jobs(job_id),
  first_name VARCHAR2(10) NOT NULL,
  last_name VARCHAR2(10) NOT NULL,
  salary NUMBER(6,0)
);
```



TEST_USER.EMPLOYEES

P	*	EMPLOYEE_ID	NUMBER (*,0)
F		DIVISION_ID	CHAR (3 BYTE)
F		JOB_ID	CHAR (3 BYTE)
	*	FIRST_NAME	VARCHAR2 (10 BYTE)
	*	LAST_NAME	VARCHAR2 (10 BYTE)
		SALARY	NUMBER (6)

EMPLOYEES_PK (EMPLOYEE_ID)

EMPLOYEES_FK_DIVISIONS (DIVISION_ID)
SYS_C0012947 (JOB_ID)

◆ EMPLOYEES_PK (EMPLOYEE_ID)

TEST_USER.DIVISIONS

P	*	DIVISION_ID	CHAR (3 BYTE)
	*	NAME	VARCHAR2 (15 BYTE)

DIVISIONS_PK (DIVISION_ID)

◆ DIVISIONS_PK (DIVISION_ID)

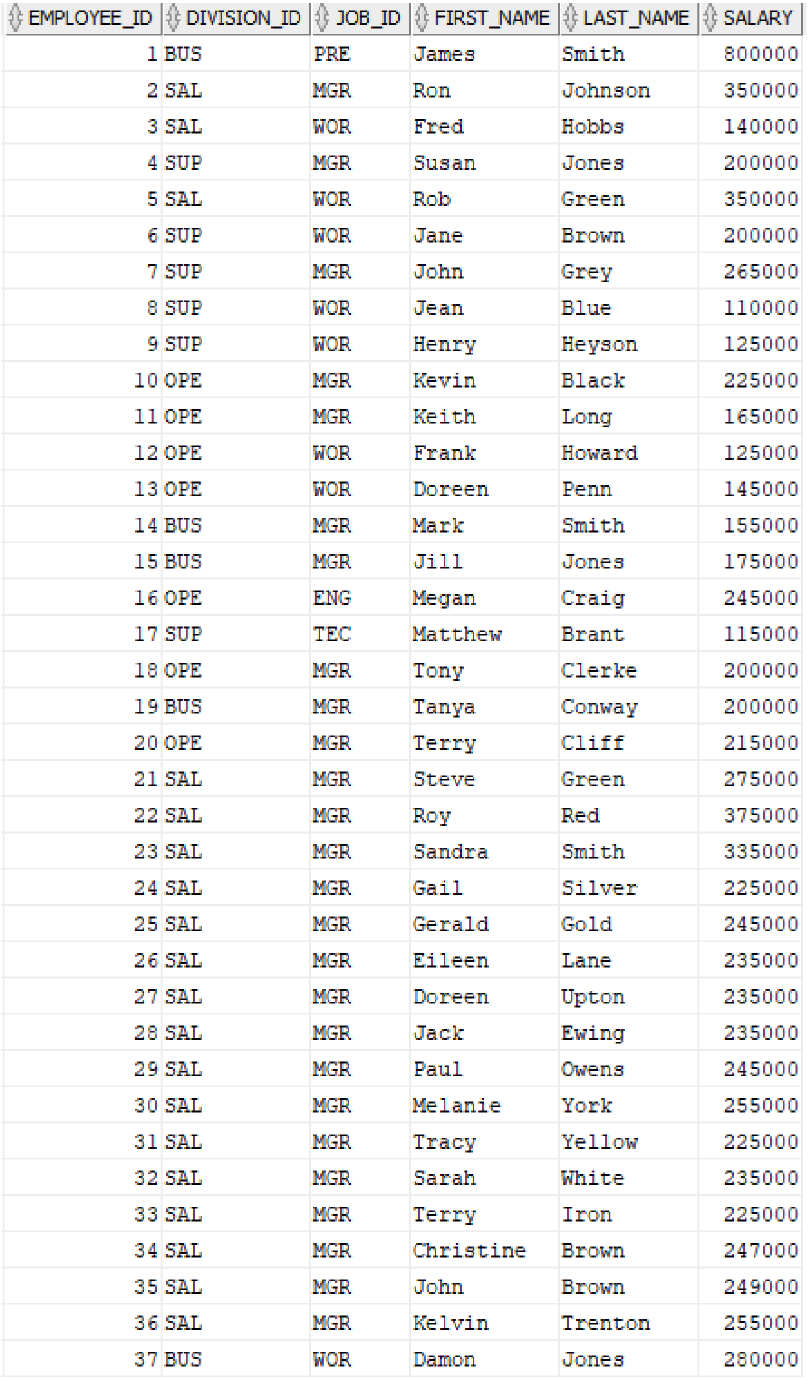
TEST_USER.JOBS

P	*	JOB_ID	CHAR (3 BYTE)
	*	NAME	VARCHAR2 (20 BYTE)

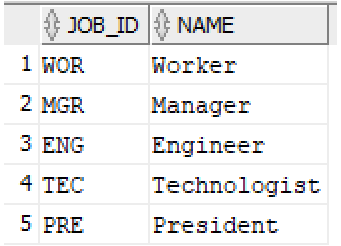
JOBS_PK (JOB_ID)

◆ JOBS_PK (JOB_ID)

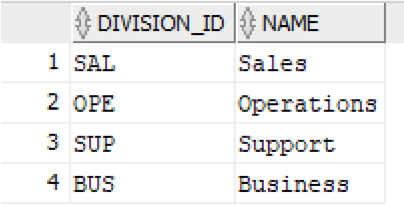
Таблицы



EMPLOYEE_ID	DIVISION_ID	JOB_ID	FIRST_NAME	LAST_NAME	SALARY
1	BUS	PRE	James	Smith	800000
2	SAL	MGR	Ron	Johnson	350000
3	SAL	WOR	Fred	Hobbs	140000
4	SUP	MGR	Susan	Jones	200000
5	SAL	WOR	Rob	Green	350000
6	SUP	WOR	Jane	Brown	200000
7	SUP	MGR	John	Grey	265000
8	SUP	WOR	Jean	Blue	110000
9	SUP	WOR	Henry	Heyson	125000
10	OPE	MGR	Kevin	Black	225000
11	OPE	MGR	Keith	Long	165000
12	OPE	WOR	Frank	Howard	125000
13	OPE	WOR	Doreen	Penn	145000
14	BUS	MGR	Mark	Smith	155000
15	BUS	MGR	Jill	Jones	175000
16	OPE	ENG	Megan	Craig	245000
17	SUP	TEC	Matthew	Brant	115000
18	OPE	MGR	Tony	Clerke	200000
19	BUS	MGR	Tanya	Conway	200000
20	OPE	MGR	Terry	Cliff	215000
21	SAL	MGR	Steve	Green	275000
22	SAL	MGR	Roy	Red	375000
23	SAL	MGR	Sandra	Smith	335000
24	SAL	MGR	Gail	Silver	225000
25	SAL	MGR	Gerald	Gold	245000
26	SAL	MGR	Eileen	Lane	235000
27	SAL	MGR	Doreen	Upton	235000
28	SAL	MGR	Jack	Ewing	235000
29	SAL	MGR	Paul	Owens	245000
30	SAL	MGR	Melanie	York	255000
31	SAL	MGR	Tracy	Yellow	225000
32	SAL	MGR	Sarah	White	235000
33	SAL	MGR	Terry	Iron	225000
34	SAL	MGR	Christine	Brown	247000
35	SAL	MGR	John	Brown	249000
36	SAL	MGR	Kelvin	Trenton	255000
37	BUS	WOR	Damon	Jones	280000

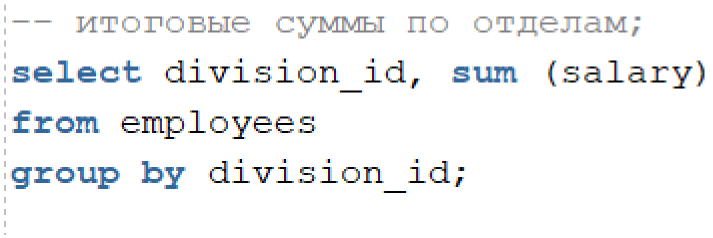


	JOB_ID	NAME
1	WOR	Worker
2	MGR	Manager
3	ENG	Engineer
4	TEC	Technologist
5	PRE	President

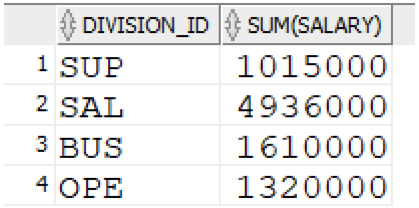


	DIVISION_ID	NAME
1	SAL	Sales
2	OPE	Operations
3	SUP	Support
4	BUS	Business

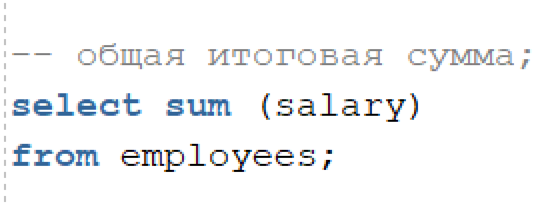
Группировка



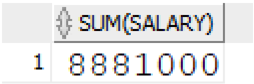
```
-- итоговые суммы по отделам;
select division_id, sum (salary)
from employees
group by division_id;
```



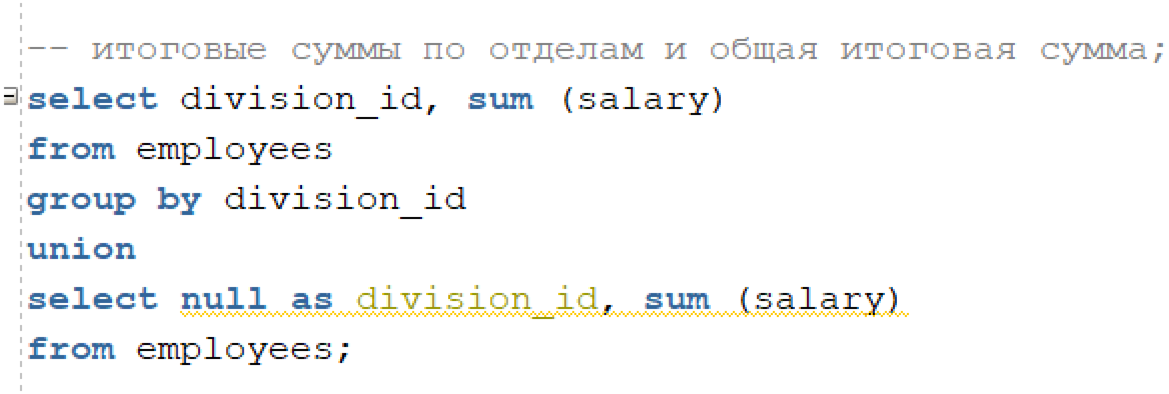
	DIVISION_ID	SUM(SALARY)
1	SUP	1015000
2	SAL	4936000
3	BUS	1610000
4	OPE	1320000



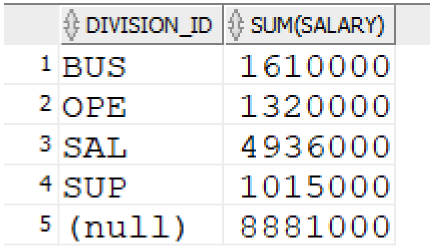
```
-- общая итоговая сумма;
select sum (salary)
from employees;
```



	SUM(SALARY)
1	8881000

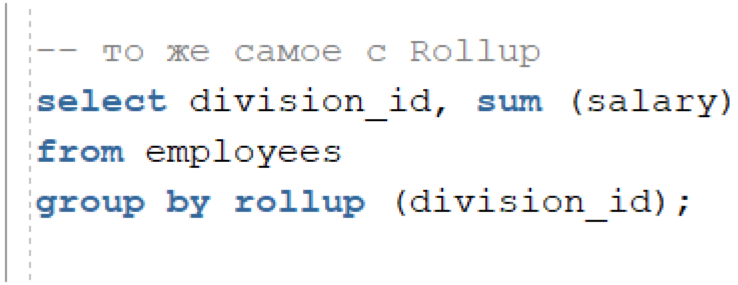


```
-- итоговые суммы по отделам и общая итоговая сумма;
select division_id, sum (salary)
from employees
group by division_id
union
select null as division_id, sum (salary)
from employees;
```

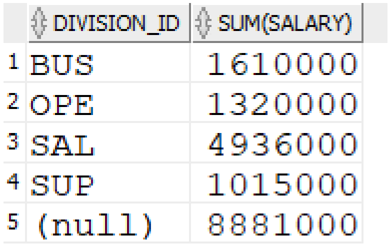


	DIVISION_ID	SUM(SALARY)
1	BUS	1610000
2	OPE	1320000
3	SAL	4936000
4	SUP	1015000
5	(null)	8881000

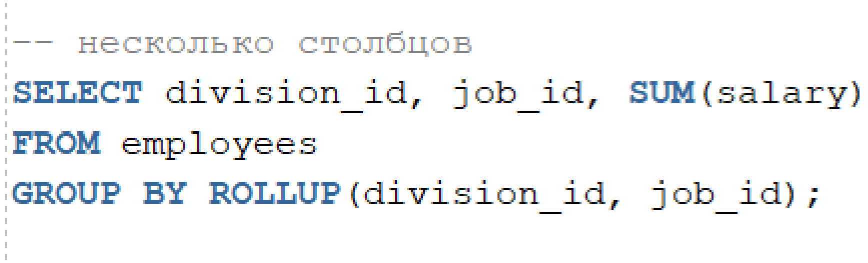
GROUP BY ROLLUP



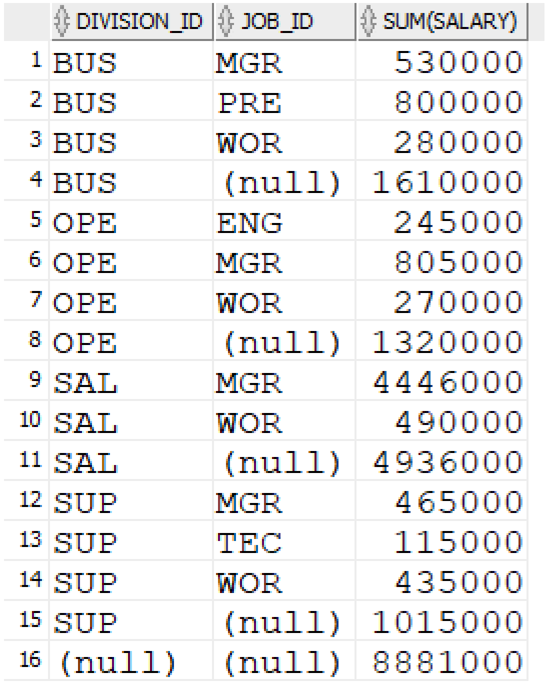
```
-- то же самое с Rollup
select division_id, sum (salary)
from employees
group by rollup (division_id);
```



	DIVISION_ID	SUM(SALARY)
1	BUS	1610000
2	OPE	1320000
3	SAL	4936000
4	SUP	1015000
5	(null)	8881000

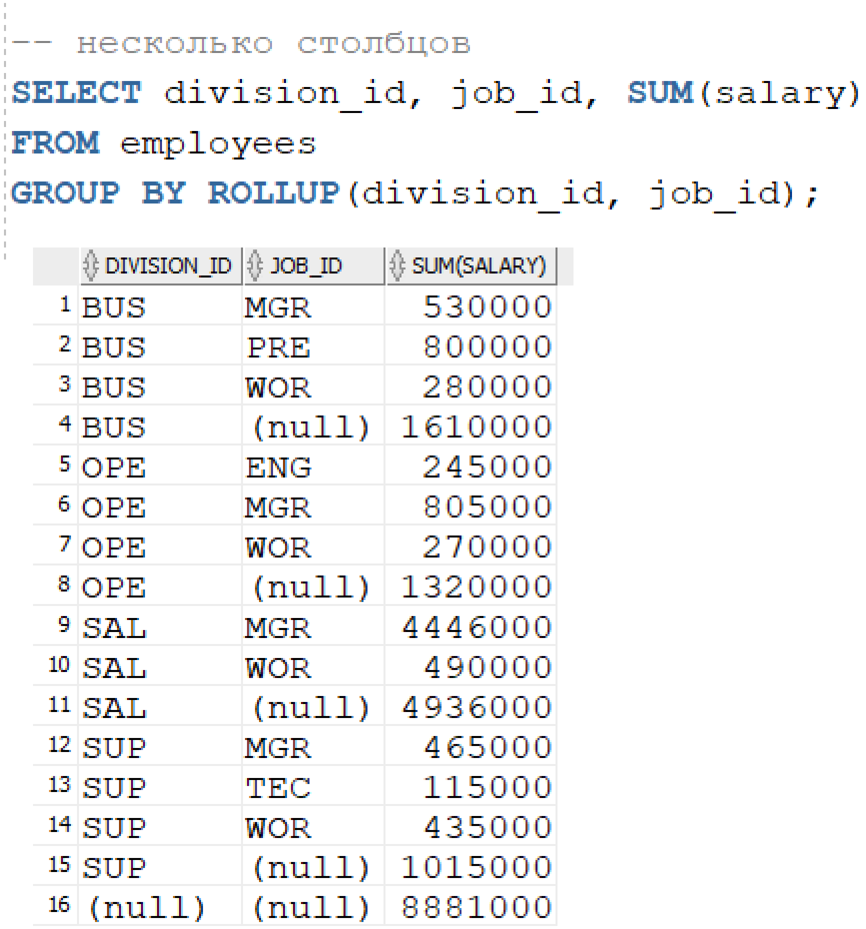


```
-- несколько столбцов
SELECT division_id, job_id, SUM(salary)
FROM employees
GROUP BY ROLLUP(division_id, job_id);
```



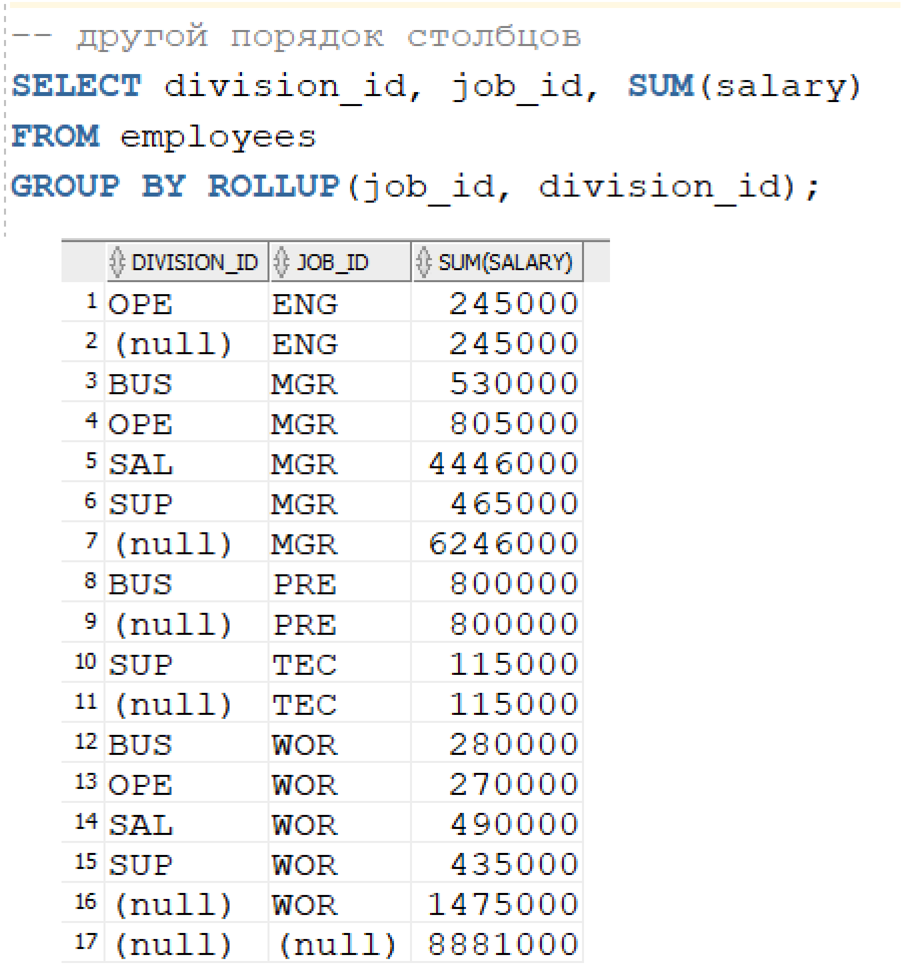
	DIVISION_ID	JOB_ID	SUM(SALARY)
1	BUS	MGR	530000
2	BUS	PRE	800000
3	BUS	WOR	280000
4	BUS	(null)	1610000
5	OPE	ENG	245000
6	OPE	MGR	805000
7	OPE	WOR	270000
8	OPE	(null)	1320000
9	SAL	MGR	4446000
10	SAL	WOR	490000
11	SAL	(null)	4936000
12	SUP	MGR	465000
13	SUP	TEC	115000
14	SUP	WOR	435000
15	SUP	(null)	1015000
16	(null)	(null)	8881000

GROUP BY ROLLUP



```
-- несколько столбцов
SELECT division_id, job_id, SUM(salary)
FROM employees
GROUP BY ROLLUP(division_id, job_id);
```

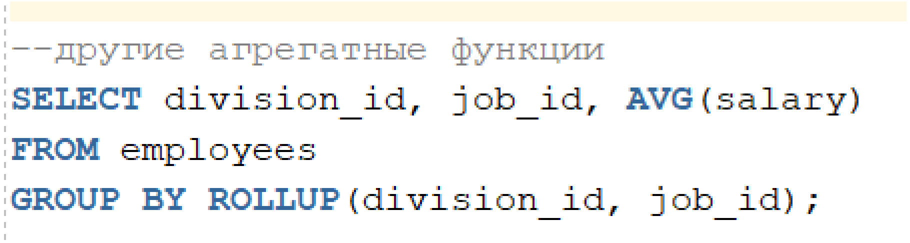
	DIVISION_ID	JOB_ID	SUM(SALARY)
1	BUS	MGR	530000
2	BUS	PRE	800000
3	BUS	WOR	280000
4	BUS	(null)	1610000
5	OPE	ENG	245000
6	OPE	MGR	805000
7	OPE	WOR	270000
8	OPE	(null)	1320000
9	SAL	MGR	4446000
10	SAL	WOR	490000
11	SAL	(null)	4936000
12	SUP	MGR	465000
13	SUP	TEC	115000
14	SUP	WOR	435000
15	SUP	(null)	1015000
16	(null)	(null)	8881000



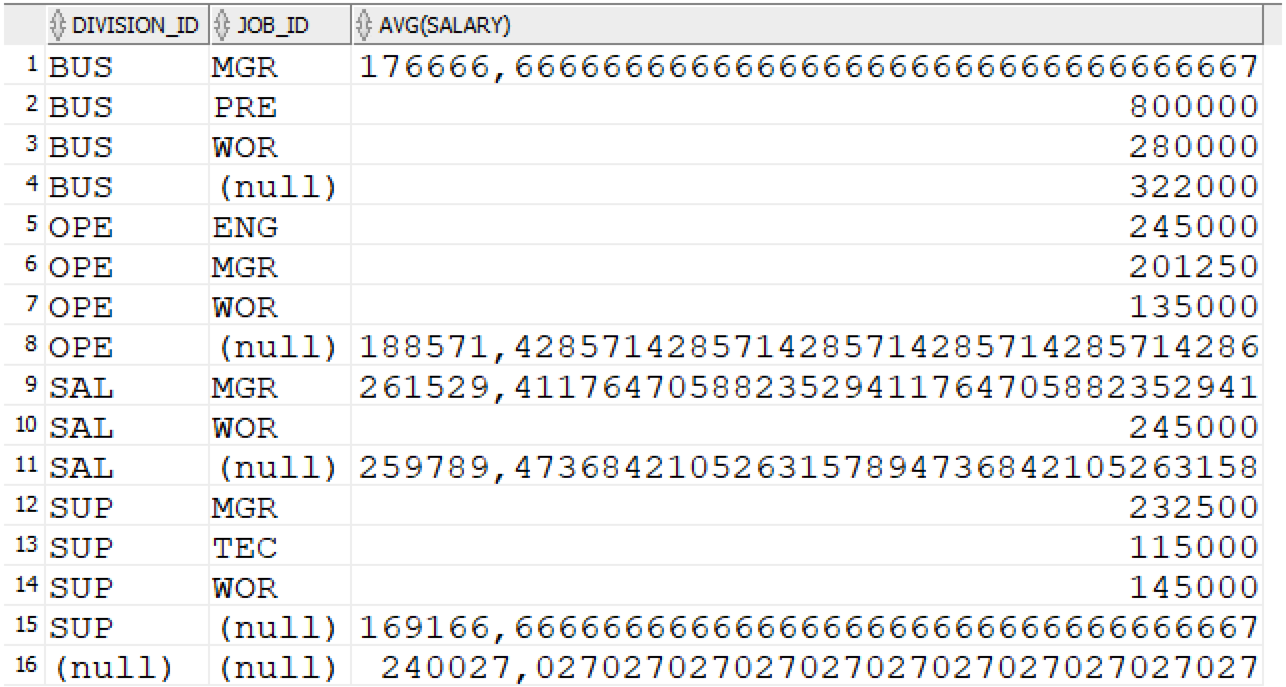
```
-- другой порядок столбцов
SELECT division_id, job_id, SUM(salary)
FROM employees
GROUP BY ROLLUP(job_id, division_id);
```

	DIVISION_ID	JOB_ID	SUM(SALARY)
1	OPE	ENG	245000
2	(null)	ENG	245000
3	BUS	MGR	530000
4	OPE	MGR	805000
5	SAL	MGR	4446000
6	SUP	MGR	465000
7	(null)	MGR	6246000
8	BUS	PRE	800000
9	(null)	PRE	800000
10	SUP	TEC	115000
11	(null)	TEC	115000
12	BUS	WOR	280000
13	OPE	WOR	270000
14	SAL	WOR	490000
15	SUP	WOR	435000
16	(null)	WOR	1475000
17	(null)	(null)	8881000

GROUP BY ROLLUP

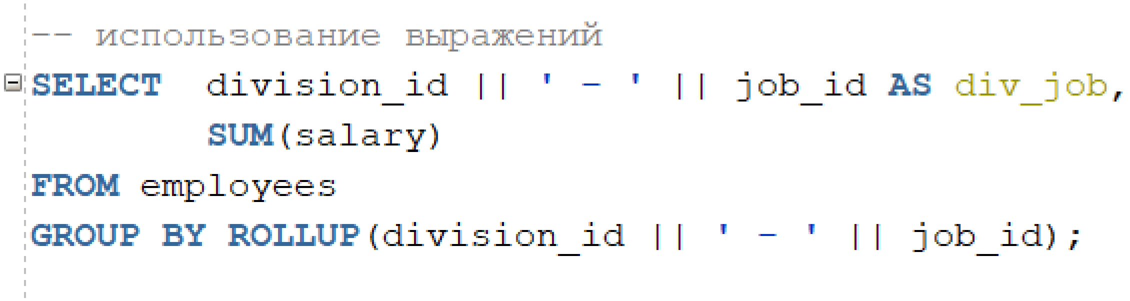


```
--другие агрегатные функции
SELECT division_id, job_id, AVG(salary)
FROM employees
GROUP BY ROLLUP(division_id, job_id);
```

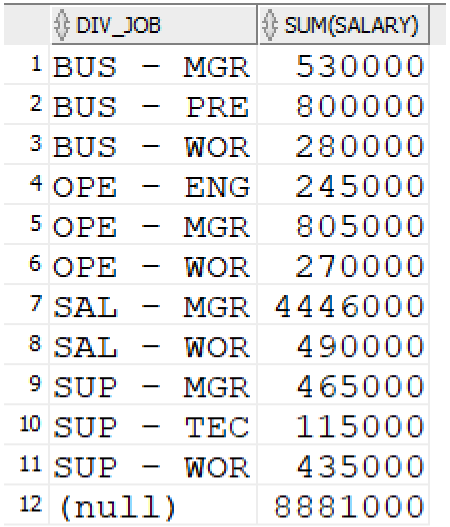


	DIVISION_ID	JOB_ID	AVG(SALARY)
1	BUS	MGR	176666,6666666666666666666666666666666667
2	BUS	PRE	800000
3	BUS	WOR	280000
4	BUS	(null)	322000
5	OPE	ENG	245000
6	OPE	MGR	201250
7	OPE	WOR	135000
8	OPE	(null)	188571,4285714285714285714285714285714286
9	SAL	MGR	261529,4117647058823529411764705882352941
10	SAL	WOR	245000
11	SAL	(null)	259789,4736842105263157894736842105263158
12	SUP	MGR	232500
13	SUP	TEC	115000
14	SUP	WOR	145000
15	SUP	(null)	169166,6666666666666666666666666666666667
16	(null)	(null)	240027,027027027027027027027027027027027

GROUP BY ROLLUP

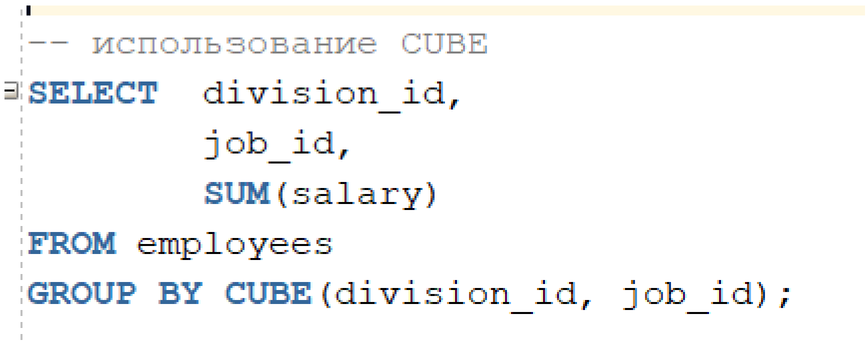


```
-- использование выражений
SELECT  division_id || ' - ' || job_id AS div_job,
        SUM(salary)
FROM employees
GROUP BY ROLLUP(division_id || ' - ' || job_id);
```

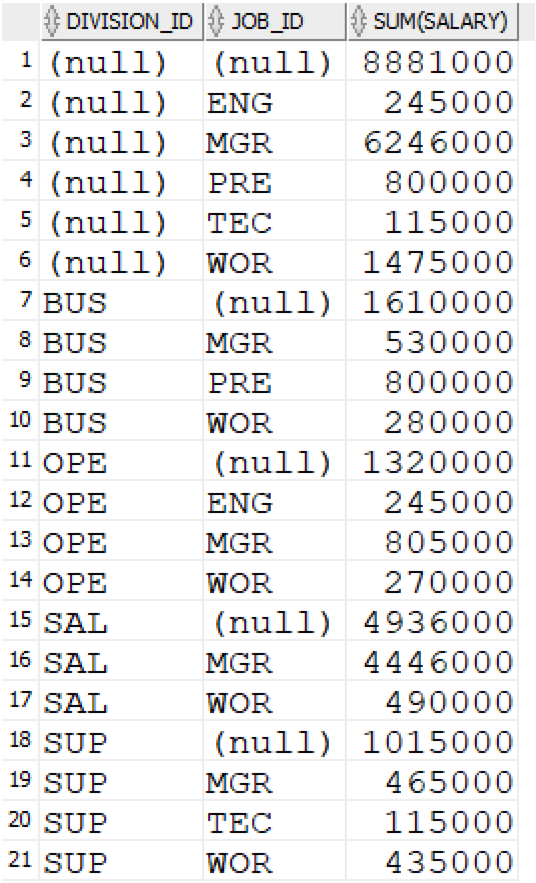


	DIV_JOB	SUM(SALARY)
1	BUS - MGR	530000
2	BUS - PRE	800000
3	BUS - WOR	280000
4	OPE - ENG	245000
5	OPE - MGR	805000
6	OPE - WOR	270000
7	SAL - MGR	4446000
8	SAL - WOR	490000
9	SUP - MGR	465000
10	SUP - TEC	115000
11	SUP - WOR	435000
12	(null)	8881000

GROUP BY CUBE

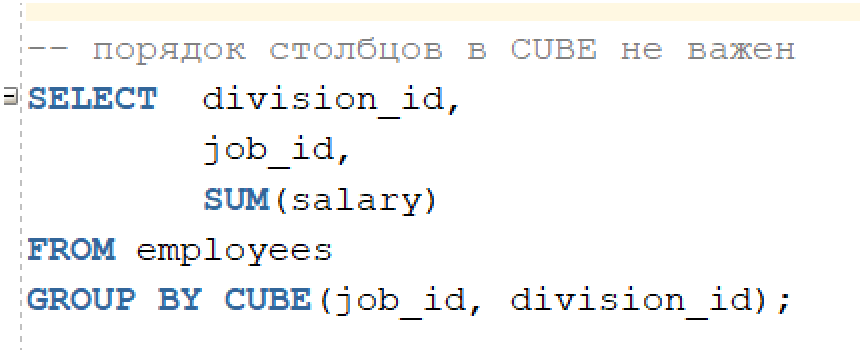


```
-- использование CUBE
SELECT  division_id,
        job_id,
        SUM(salary)
FROM employees
GROUP BY CUBE(division_id, job_id);
```

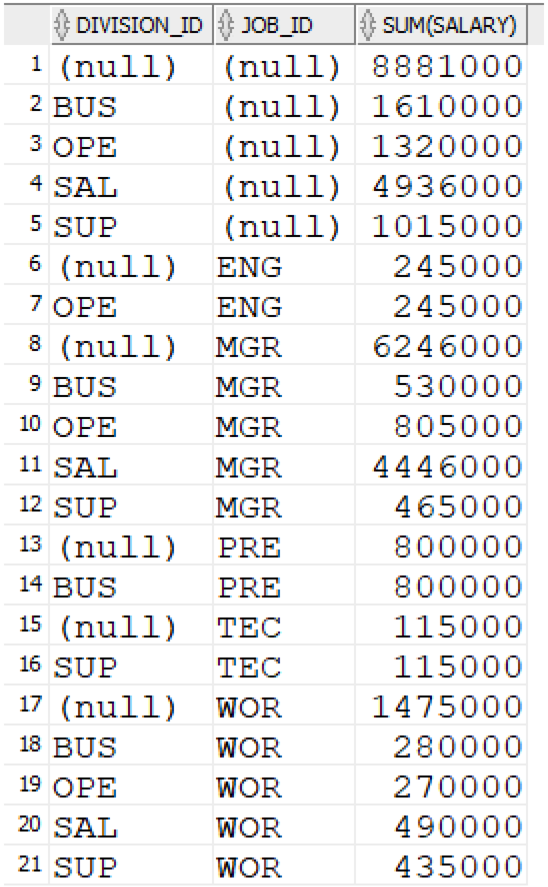


	DIVISION_ID	JOB_ID	SUM(SALARY)
1	(null)	(null)	8881000
2	(null)	ENG	245000
3	(null)	MGR	6246000
4	(null)	PRE	800000
5	(null)	TEC	115000
6	(null)	WOR	1475000
7	BUS	(null)	1610000
8	BUS	MGR	530000
9	BUS	PRE	800000
10	BUS	WOR	280000
11	OPE	(null)	1320000
12	OPE	ENG	245000
13	OPE	MGR	805000
14	OPE	WOR	270000
15	SAL	(null)	4936000
16	SAL	MGR	4446000
17	SAL	WOR	490000
18	SUP	(null)	1015000
19	SUP	MGR	465000
20	SUP	TEC	115000
21	SUP	WOR	435000

GROUP BY CUBE

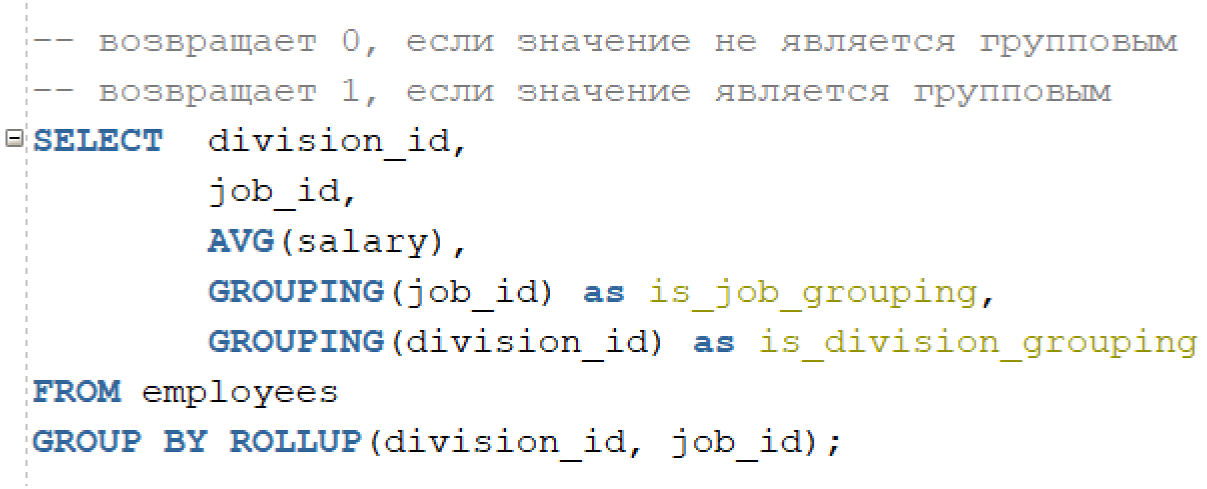


```
-- порядок столбцов в CUBE не важен
SELECT  division_id,
        job_id,
        SUM(salary)
FROM employees
GROUP BY CUBE(job_id, division_id);
```

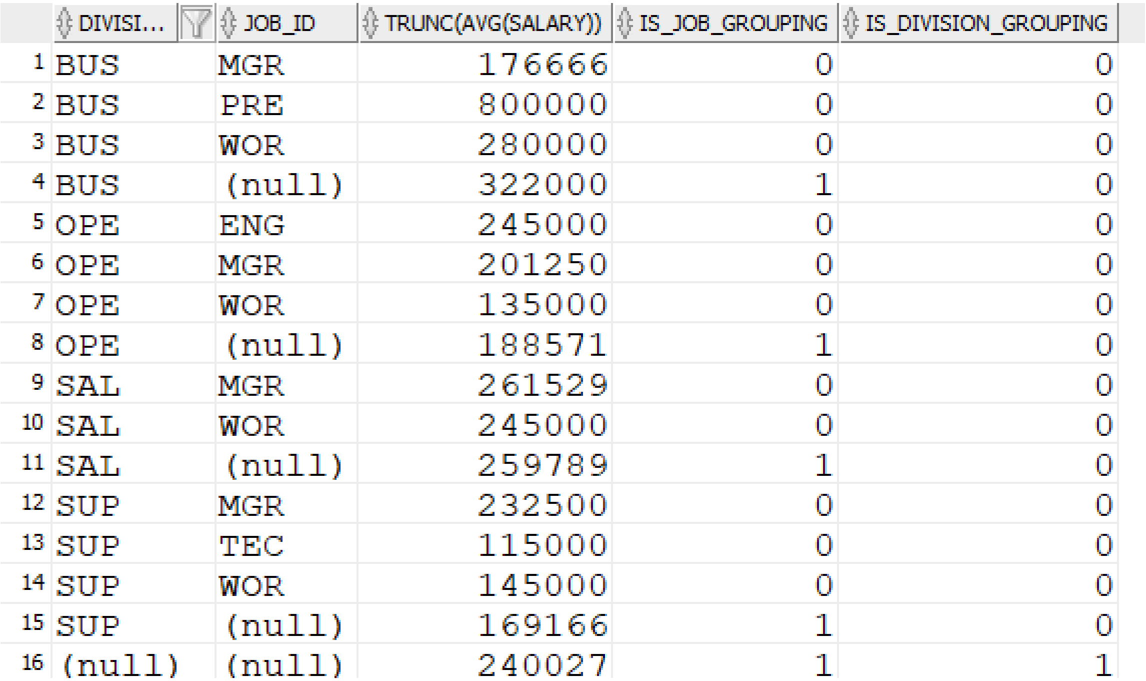


	DIVISION_ID	JOB_ID	SUM(SALARY)
1	(null)	(null)	8881000
2	BUS	(null)	1610000
3	OPE	(null)	1320000
4	SAL	(null)	4936000
5	SUP	(null)	1015000
6	(null)	ENG	245000
7	OPE	ENG	245000
8	(null)	MGR	6246000
9	BUS	MGR	530000
10	OPE	MGR	805000
11	SAL	MGR	4446000
12	SUP	MGR	465000
13	(null)	PRE	800000
14	BUS	PRE	800000
15	(null)	TEC	115000
16	SUP	TEC	115000
17	(null)	WOR	1475000
18	BUS	WOR	280000
19	OPE	WOR	270000
20	SAL	WOR	490000
21	SUP	WOR	435000

Функция GROUPING

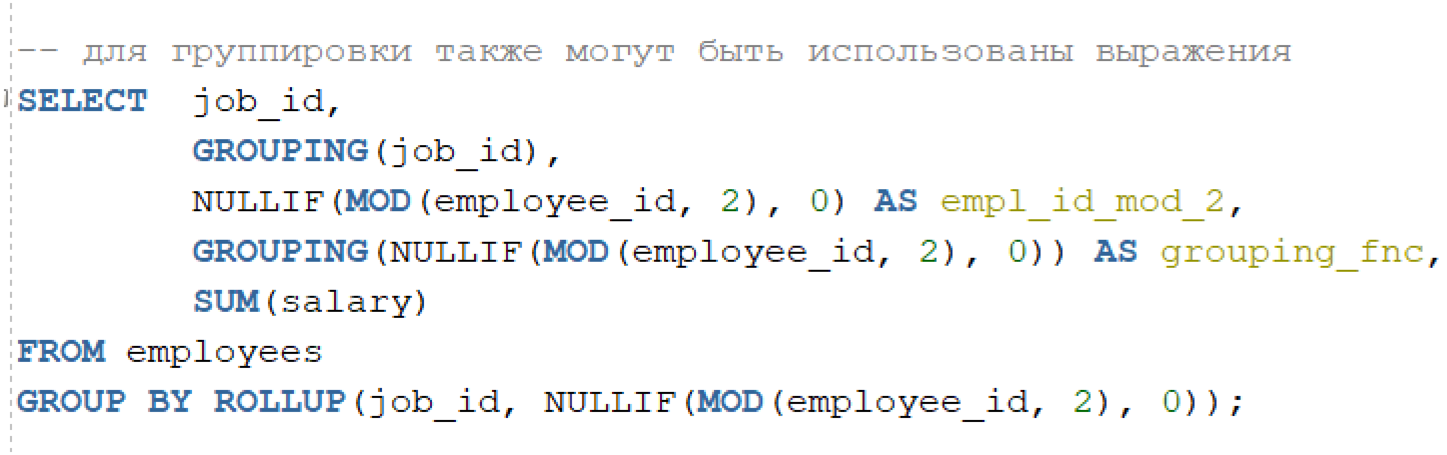


```
-- возвращает 0, если значение не является групповым
-- возвращает 1, если значение является групповым
SELECT  division_id,
        job_id,
        AVG(salary),
        GROUPING(job_id) as is_job_grouping,
        GROUPING(division_id) as is_division_grouping
FROM employees
GROUP BY ROLLUP(division_id, job_id);
```

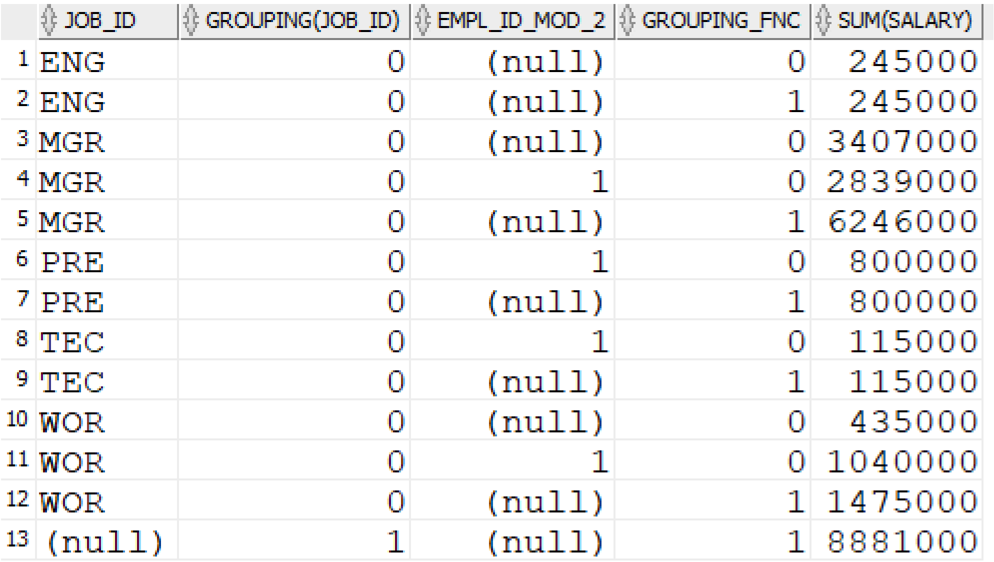


	DIVISI...	JOB_ID	TRUNC(AVG(SALARY))	IS_JOB_GROUPING	IS_DIVISION_GROUPING
1	BUS	MGR	176666	0	0
2	BUS	PRE	800000	0	0
3	BUS	WOR	280000	0	0
4	BUS	(null)	322000	1	0
5	OPE	ENG	245000	0	0
6	OPE	MGR	201250	0	0
7	OPE	WOR	135000	0	0
8	OPE	(null)	188571	1	0
9	SAL	MGR	261529	0	0
10	SAL	WOR	245000	0	0
11	SAL	(null)	259789	1	0
12	SUP	MGR	232500	0	0
13	SUP	TEC	115000	0	0
14	SUP	WOR	145000	0	0
15	SUP	(null)	169166	1	0
16	(null)	(null)	240027	1	1

Функция GROUPING

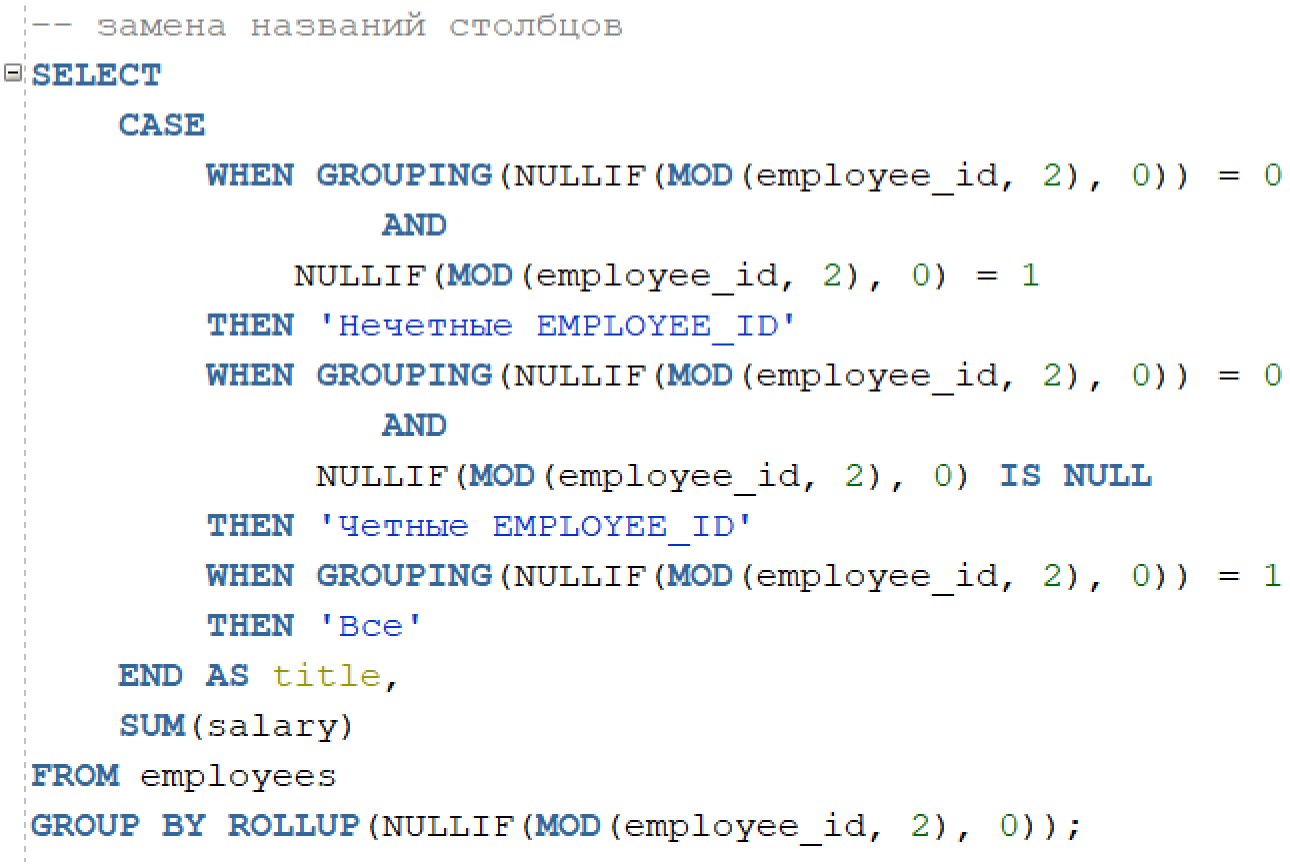


```
-- для группировки также могут быть использованы выражения
SELECT  job_id,
        GROUPING(job_id),
        NULLIF(MOD(employee_id, 2), 0) AS empl_id_mod_2,
        GROUPING(NULLIF(MOD(employee_id, 2), 0)) AS grouping_fnc,
        SUM(salary)
FROM employees
GROUP BY ROLLUP(job_id, NULLIF(MOD(employee_id, 2), 0));
```

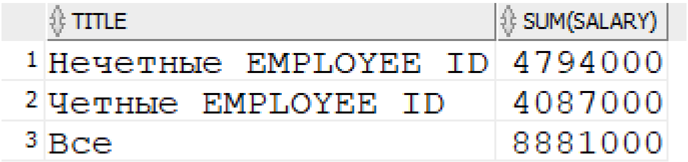


	JOB_ID	GROUPING(JOB_ID)	EMPL_ID_MOD_2	GROUPING_FNC	SUM(SALARY)
1	ENG	0	(null)	0	245000
2	ENG	0	(null)	1	245000
3	MGR	0	(null)	0	3407000
4	MGR	0	1	0	2839000
5	MGR	0	(null)	1	6246000
6	PRE	0	1	0	800000
7	PRE	0	(null)	1	800000
8	TEC	0	1	0	115000
9	TEC	0	(null)	1	115000
10	WOR	0	(null)	0	435000
11	WOR	0	1	0	1040000
12	WOR	0	(null)	1	1475000
13	(null)	1	(null)	1	8881000

Функция GROUPING

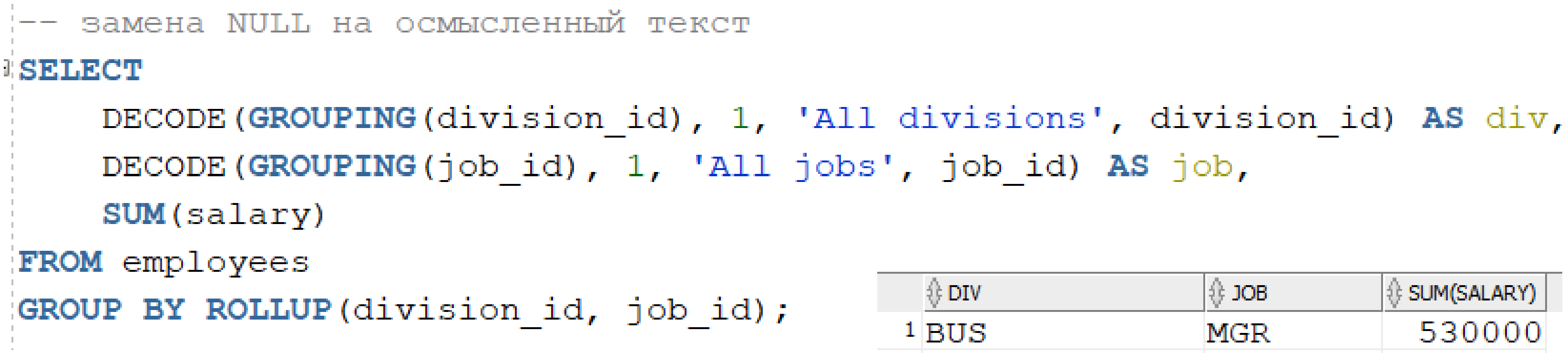


```
-- замена названий столбцов
SELECT
    CASE
        WHEN GROUPING(NULLIF(MOD(employee_id, 2), 0)) = 0
                AND
            NULLIF(MOD(employee_id, 2), 0) = 1
        THEN 'Нечетные EMPLOYEE_ID'
        WHEN GROUPING(NULLIF(MOD(employee_id, 2), 0)) = 0
                AND
             NULLIF(MOD(employee_id, 2), 0) IS NULL
        THEN 'Четные EMPLOYEE_ID'
        WHEN GROUPING(NULLIF(MOD(employee_id, 2), 0)) = 1
        THEN 'Все'
    END AS title,
    SUM(salary)
FROM employees
GROUP BY ROLLUP(NULLIF(MOD(employee_id, 2), 0));
```

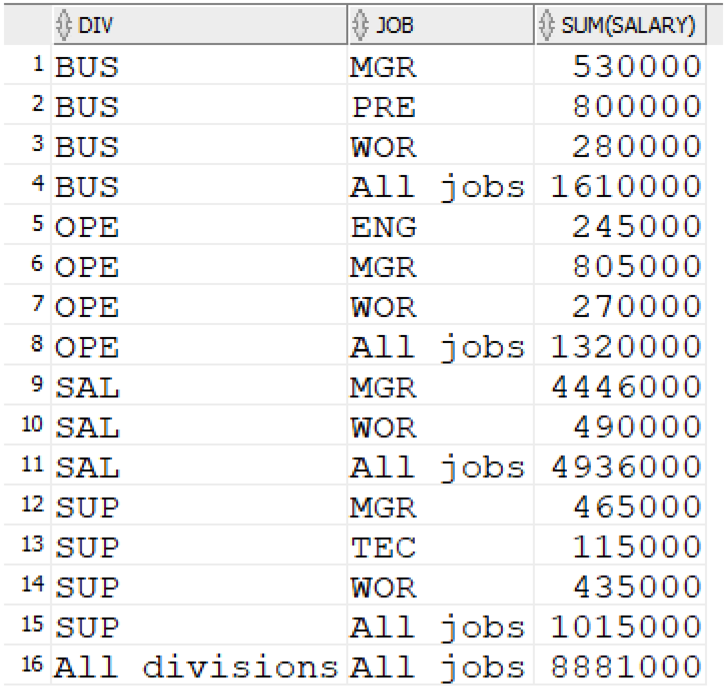


	TITLE	SUM(SALARY)
1	Нечетные EMPLOYEE ID	4794000
2	Четные EMPLOYEE ID	4087000
3	Все	8881000

Функция GROUPING

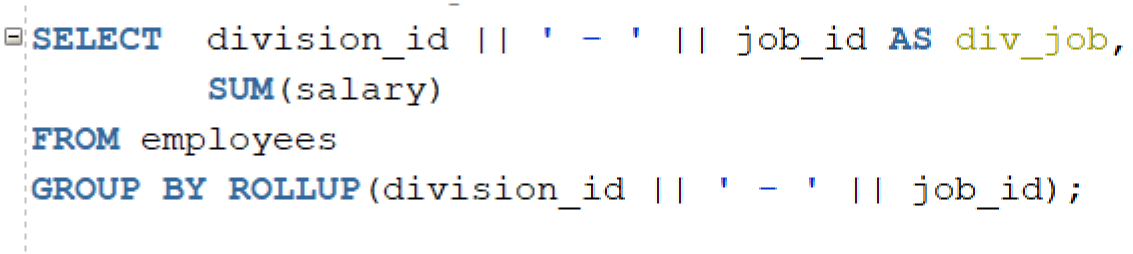


```
-- замена NULL на осмысленный текст
SELECT
    DECODE(GROUPING(division_id), 1, 'All divisions', division_id) AS div,
    DECODE(GROUPING(job_id), 1, 'All jobs', job_id) AS job,
    SUM(salary)
FROM employees
GROUP BY ROLLUP(division_id, job_id);
```

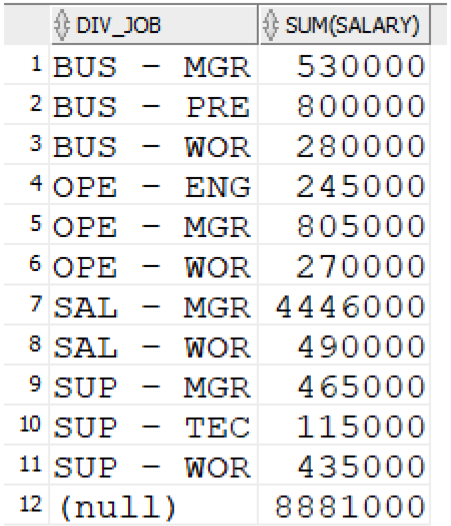


	DIV	JOB	SUM(SALARY)
1	BUS	MGR	530000
2	BUS	PRE	800000
3	BUS	WOR	280000
4	BUS	All jobs	1610000
5	OPE	ENG	245000
6	OPE	MGR	805000
7	OPE	WOR	270000
8	OPE	All jobs	1320000
9	SAL	MGR	4446000
10	SAL	WOR	490000
11	SAL	All jobs	4936000
12	SUP	MGR	465000
13	SUP	TEC	115000
14	SUP	WOR	435000
15	SUP	All jobs	1015000
16	All divisions	All jobs	8881000

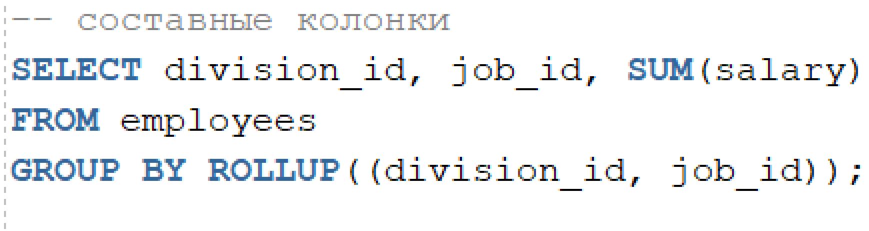
Составные колонки



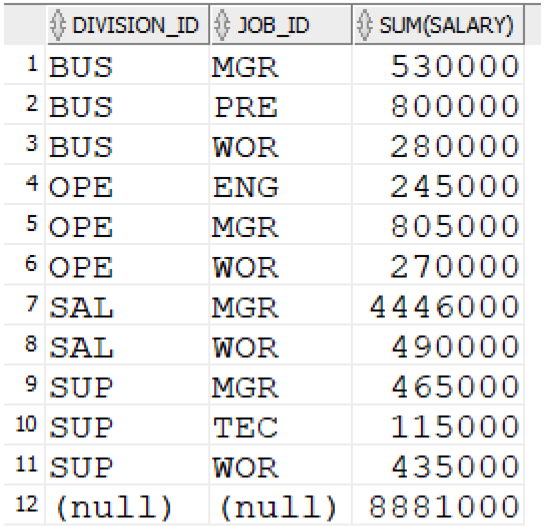
```
SELECT  division_id || ' - ' || job_id AS div_job,
        SUM(salary)
FROM employees
GROUP BY ROLLUP(division_id || ' - ' || job_id);
```



	DIV_JOB	SUM(SALARY)
1	BUS - MGR	530000
2	BUS - PRE	800000
3	BUS - WOR	280000
4	OPE - ENG	245000
5	OPE - MGR	805000
6	OPE - WOR	270000
7	SAL - MGR	4446000
8	SAL - WOR	490000
9	SUP - MGR	465000
10	SUP - TEC	115000
11	SUP - WOR	435000
12	(null)	8881000

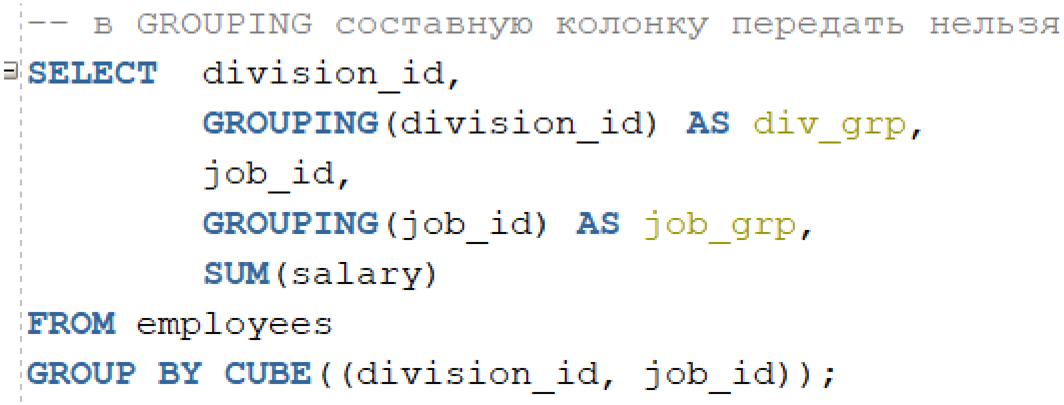


```
-- составные колонки
SELECT division_id, job_id, SUM(salary)
FROM employees
GROUP BY ROLLUP((division_id, job_id));
```

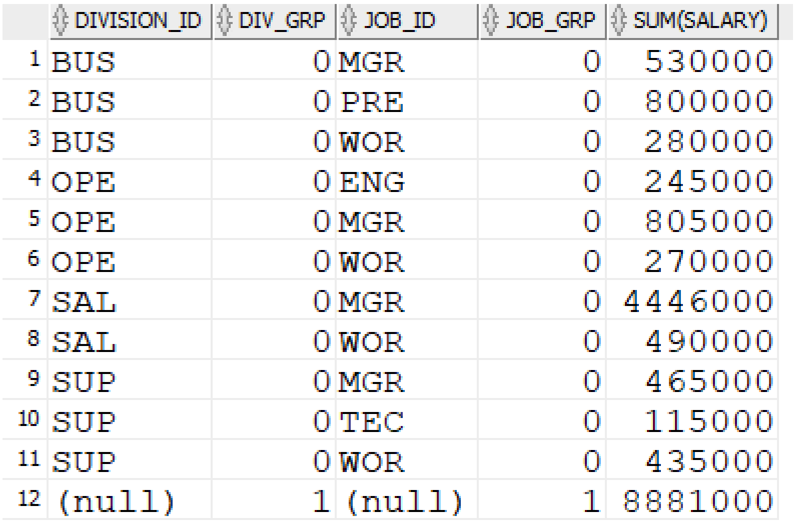


	DIVISION_ID	JOB_ID	SUM(SALARY)
1	BUS	MGR	530000
2	BUS	PRE	800000
3	BUS	WOR	280000
4	OPE	ENG	245000
5	OPE	MGR	805000
6	OPE	WOR	270000
7	SAL	MGR	4446000
8	SAL	WOR	490000
9	SUP	MGR	465000
10	SUP	TEC	115000
11	SUP	WOR	435000
12	(null)	(null)	8881000

Составные колонки

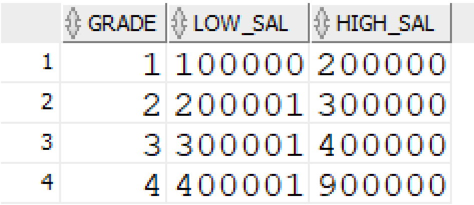


```
-- в GROUPING составную колонку передать нельзя
SELECT  division_id,
        GROUPING(division_id) AS div_grp,
        job_id,
        GROUPING(job_id) AS job_grp,
        SUM(salary)
FROM employees
GROUP BY CUBE((division_id, job_id));
```

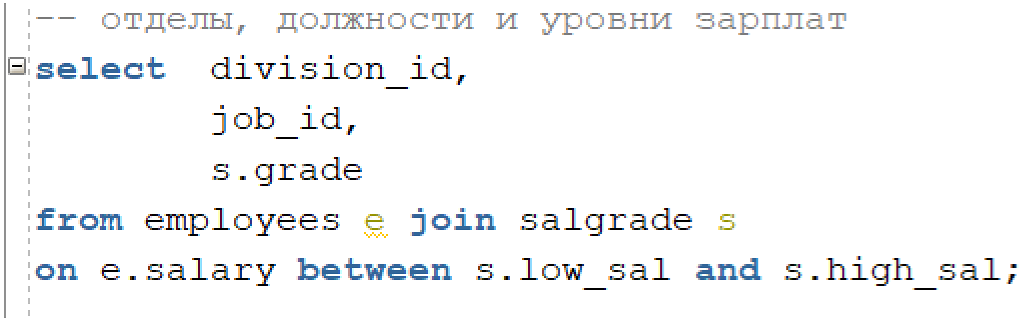


	DIVISION_ID	DIV_GRP	JOB_ID	JOB_GRP	SUM(SALARY)
1	BUS	0	MGR	0	530000
2	BUS	0	PRE	0	800000
3	BUS	0	WOR	0	280000
4	OPE	0	ENG	0	245000
5	OPE	0	MGR	0	805000
6	OPE	0	WOR	0	270000
7	SAL	0	MGR	0	4446000
8	SAL	0	WOR	0	490000
9	SUP	0	MGR	0	465000
10	SUP	0	TEC	0	115000
11	SUP	0	WOR	0	435000
12	(null)	1	(null)	1	8881000

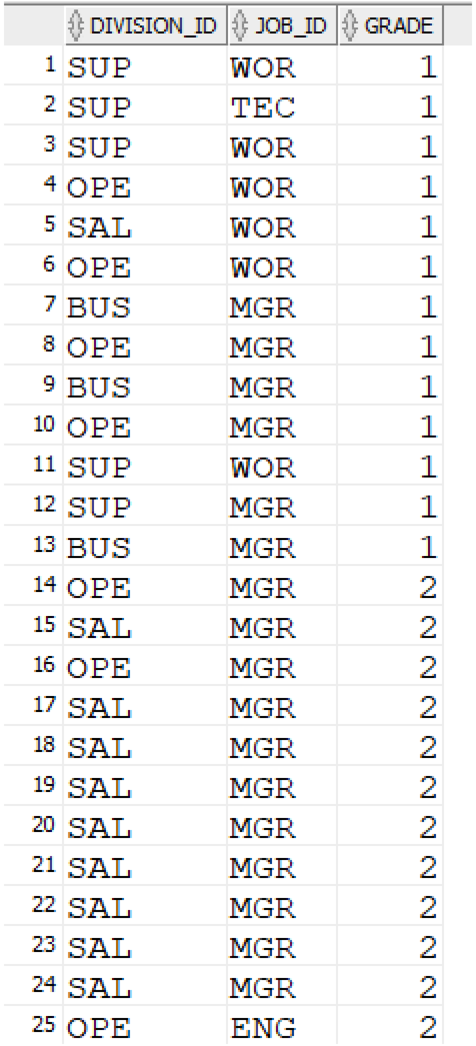
GROUPING SETS



	GRADE	LOW_SAL	HIGH_SAL
1	1	100000	200000
2	2	200001	300000
3	3	300001	400000
4	4	400001	900000

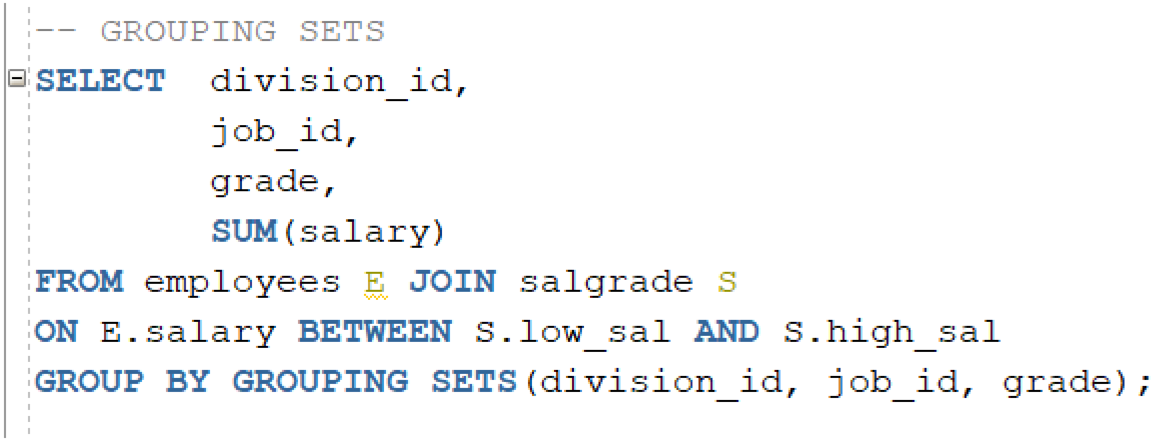


```
-- отделы, должности и уровни зарплат
select  division_id,
        job_id,
        s.grade
from employees e join salgrade s
on e.salary between s.low_sal and s.high_sal;
```

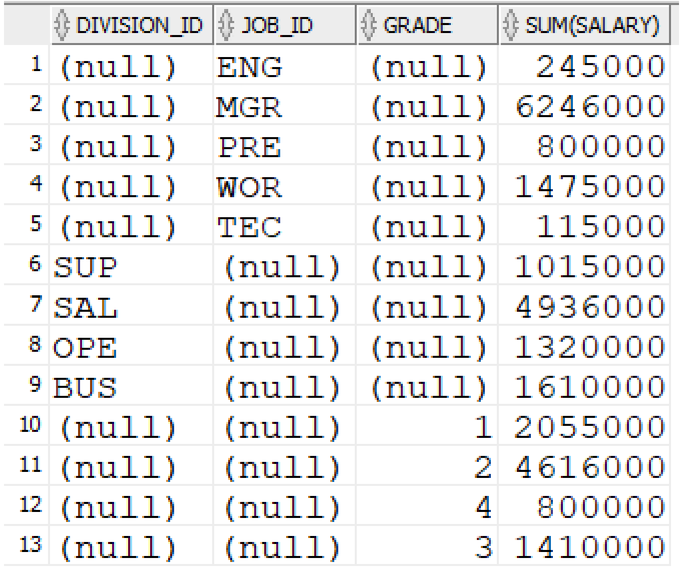


	DIVISION_ID	JOB_ID	GRADE
1	SUP	WOR	1
2	SUP	TEC	1
3	SUP	WOR	1
4	OPE	WOR	1
5	SAL	WOR	1
6	OPE	WOR	1
7	BUS	MGR	1
8	OPE	MGR	1
9	BUS	MGR	1
10	OPE	MGR	1
11	SUP	WOR	1
12	SUP	MGR	1
13	BUS	MGR	1
14	OPE	MGR	2
15	SAL	MGR	2
16	OPE	MGR	2
17	SAL	MGR	2
18	SAL	MGR	2
19	SAL	MGR	2
20	SAL	MGR	2
21	SAL	MGR	2
22	SAL	MGR	2
23	SAL	MGR	2
24	SAL	MGR	2
25	OPE	ENG	2

GROUPING SETS

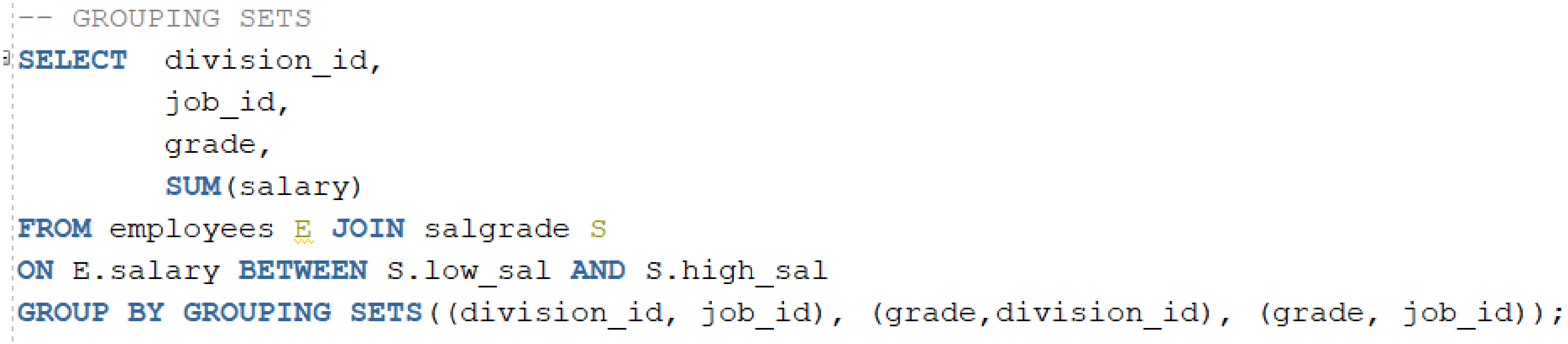


```
-- GROUPING SETS
SELECT  division_id,
        job_id,
        grade,
        SUM(salary)
FROM employees E JOIN salgrade S
ON E.salary BETWEEN S.low_sal AND S.high_sal
GROUP BY GROUPING SETS(division_id, job_id, grade);
```

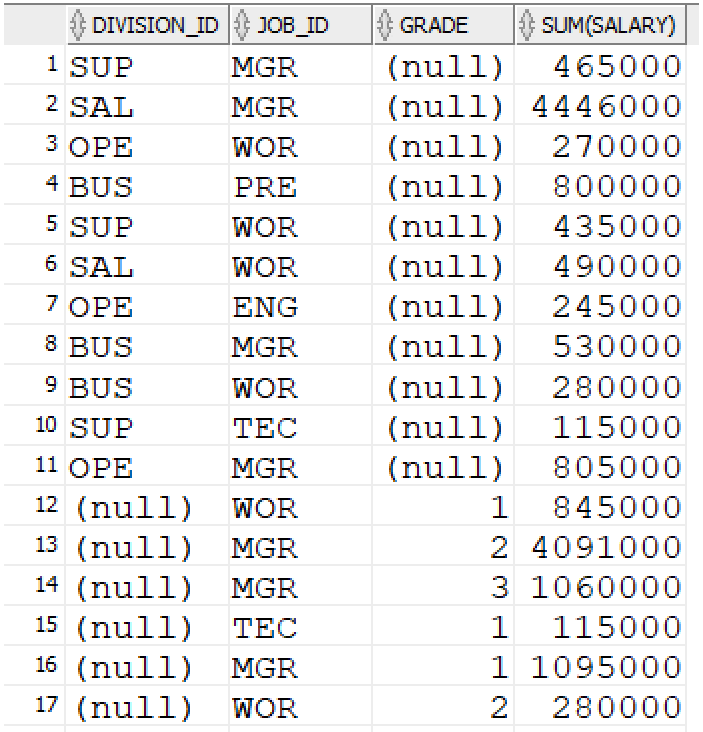


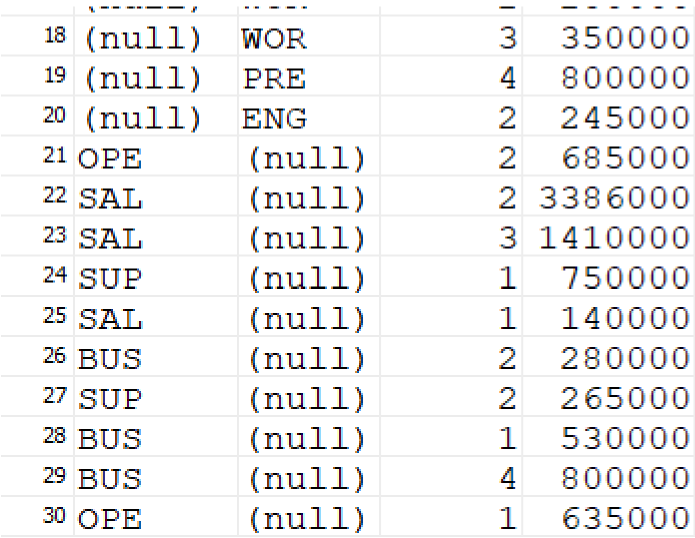
	DIVISION_ID	JOB_ID	GRADE	SUM(SALARY)
1	(null)	ENG	(null)	245000
2	(null)	MGR	(null)	6246000
3	(null)	PRE	(null)	800000
4	(null)	WOR	(null)	1475000
5	(null)	TEC	(null)	115000
6	SUP	(null)	(null)	1015000
7	SAL	(null)	(null)	4936000
8	OPE	(null)	(null)	1320000
9	BUS	(null)	(null)	1610000
10	(null)	(null)	1	2055000
11	(null)	(null)	2	4616000
12	(null)	(null)	4	800000
13	(null)	(null)	3	1410000

GROUPING SETS



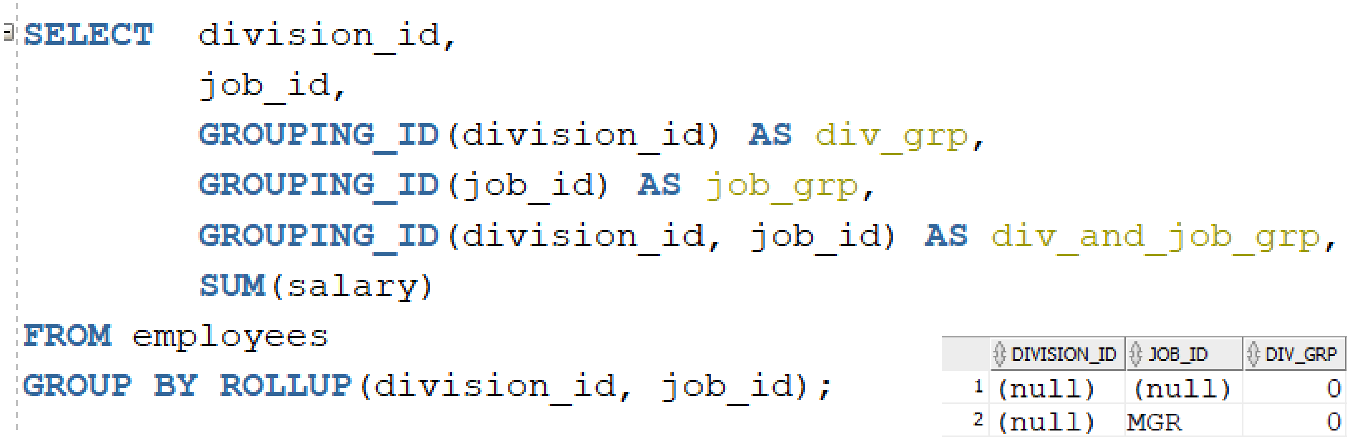
```
-- GROUPING SETS
SELECT  division_id,
        job_id,
        grade,
        SUM(salary)
FROM employees E JOIN salgrade S
ON E.salary BETWEEN S.low_sal AND S.high_sal
GROUP BY GROUPING SETS((division_id, job_id), (grade,division_id), (grade, job_id));
```



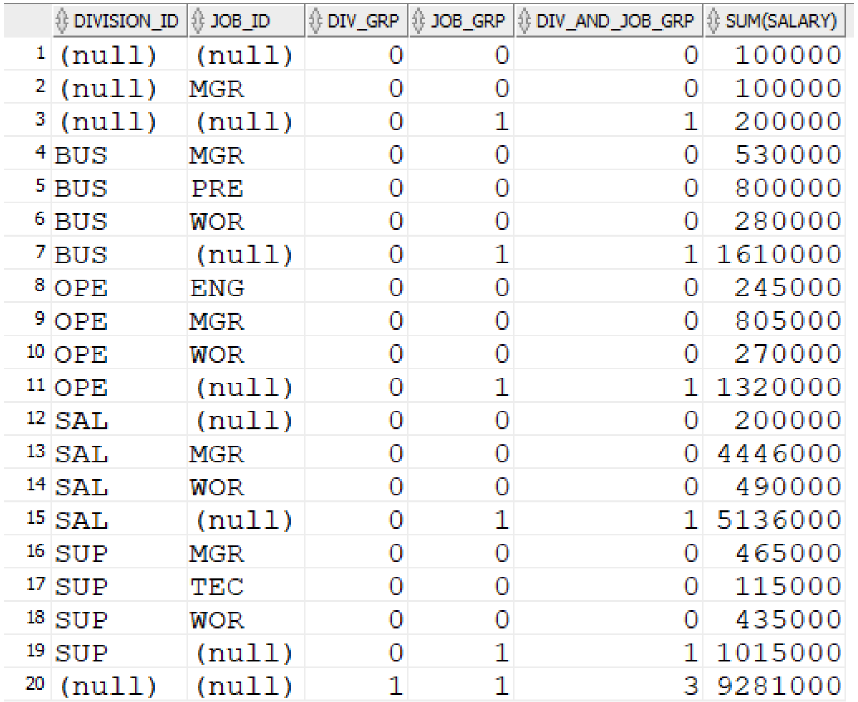


	DIVISION_ID	JOB_ID	GRADE	SUM(SALARY)
1	SUP	MGR	(null)	465000
2	SAL	MGR	(null)	4446000
3	OPE	WOR	(null)	270000
4	BUS	PRE	(null)	800000
5	SUP	WOR	(null)	435000
6	SAL	WOR	(null)	490000
7	OPE	ENG	(null)	245000
8	BUS	MGR	(null)	530000
9	BUS	WOR	(null)	280000
10	SUP	TEC	(null)	115000
11	OPE	MGR	(null)	805000
12	(null)	WOR	1	845000
13	(null)	MGR	2	4091000
14	(null)	MGR	3	1060000
15	(null)	TEC	1	115000
16	(null)	MGR	1	1095000
17	(null)	WOR	2	280000
18	(null)	WOR	3	350000
19	(null)	PRE	4	800000
20	(null)	ENG	2	245000
21	OPE	(null)	2	685000
22	SAL	(null)	2	3386000
23	SAL	(null)	3	1410000
24	SUP	(null)	1	750000
25	SAL	(null)	1	140000
26	BUS	(null)	2	280000
27	SUP	(null)	2	265000
28	BUS	(null)	1	530000
29	BUS	(null)	4	800000
30	OPE	(null)	1	635000

Функция GROUPING_ID()



```
SELECT   division_id,
         job_id,
         GROUPING_ID(division_id) AS div_grp,
         GROUPING_ID(job_id) AS job_grp,
         GROUPING_ID(division_id, job_id) AS div_and_job_grp,
         SUM(salary)
FROM employees
GROUP BY ROLLUP(division_id, job_id);
```

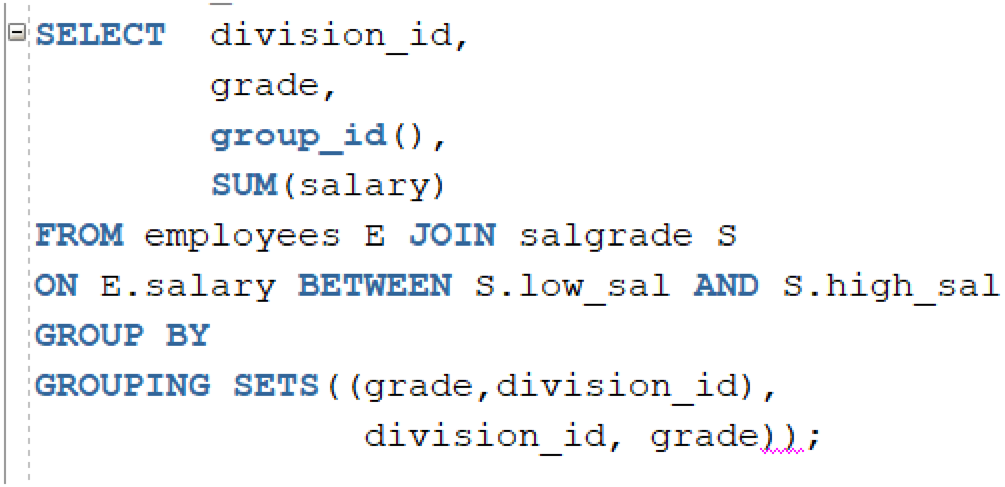


	DIVISION_ID	JOB_ID	DIV_GRP	JOB_GRP	DIV_AND_JOB_GRP	SUM(SALARY)
1	(null)	(null)	0	0	0	100000
2	(null)	MGR	0	0	0	100000
3	(null)	(null)	0	1	1	200000
4	BUS	MGR	0	0	0	530000
5	BUS	PRE	0	0	0	800000
6	BUS	WOR	0	0	0	280000
7	BUS	(null)	0	1	1	1610000
8	OPE	ENG	0	0	0	245000
9	OPE	MGR	0	0	0	805000
10	OPE	WOR	0	0	0	270000
11	OPE	(null)	0	1	1	1320000
12	SAL	(null)	0	0	0	200000
13	SAL	MGR	0	0	0	4446000
14	SAL	WOR	0	0	0	490000
15	SAL	(null)	0	1	1	5136000
16	SUP	MGR	0	0	0	465000
17	SUP	TEC	0	0	0	115000
18	SUP	WOR	0	0	0	435000
19	SUP	(null)	0	1	1	1015000
20	(null)	(null)	1	1	3	9281000

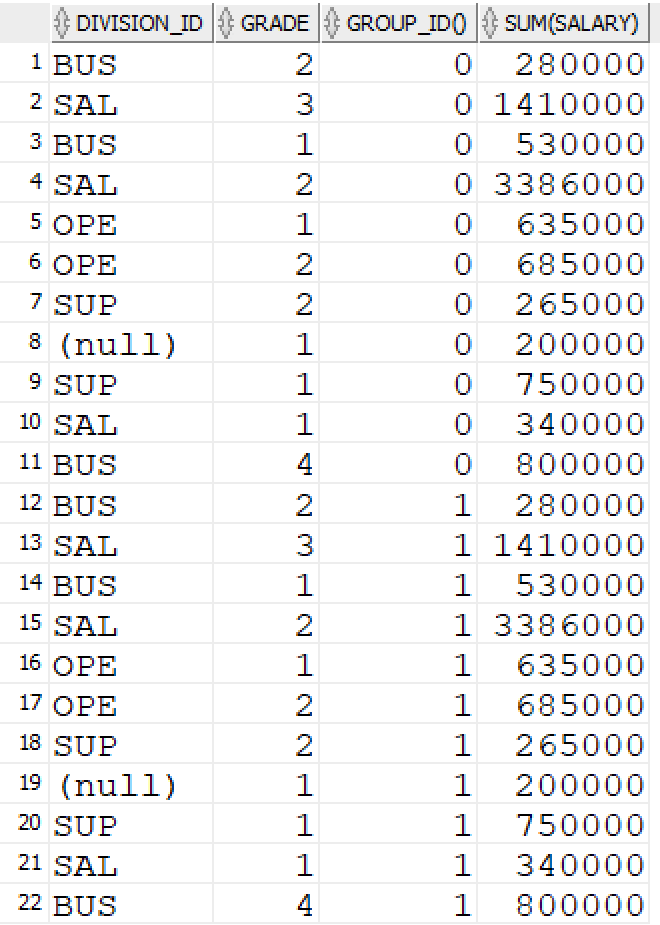
- Формирует значение битового вектора в зависимости от значений NULL в столбцах группировки

Division_id/ job_id	NOT NULL	NULL
NOT NULL	00	01
NULL	10	11

Функция GROUP_ID()



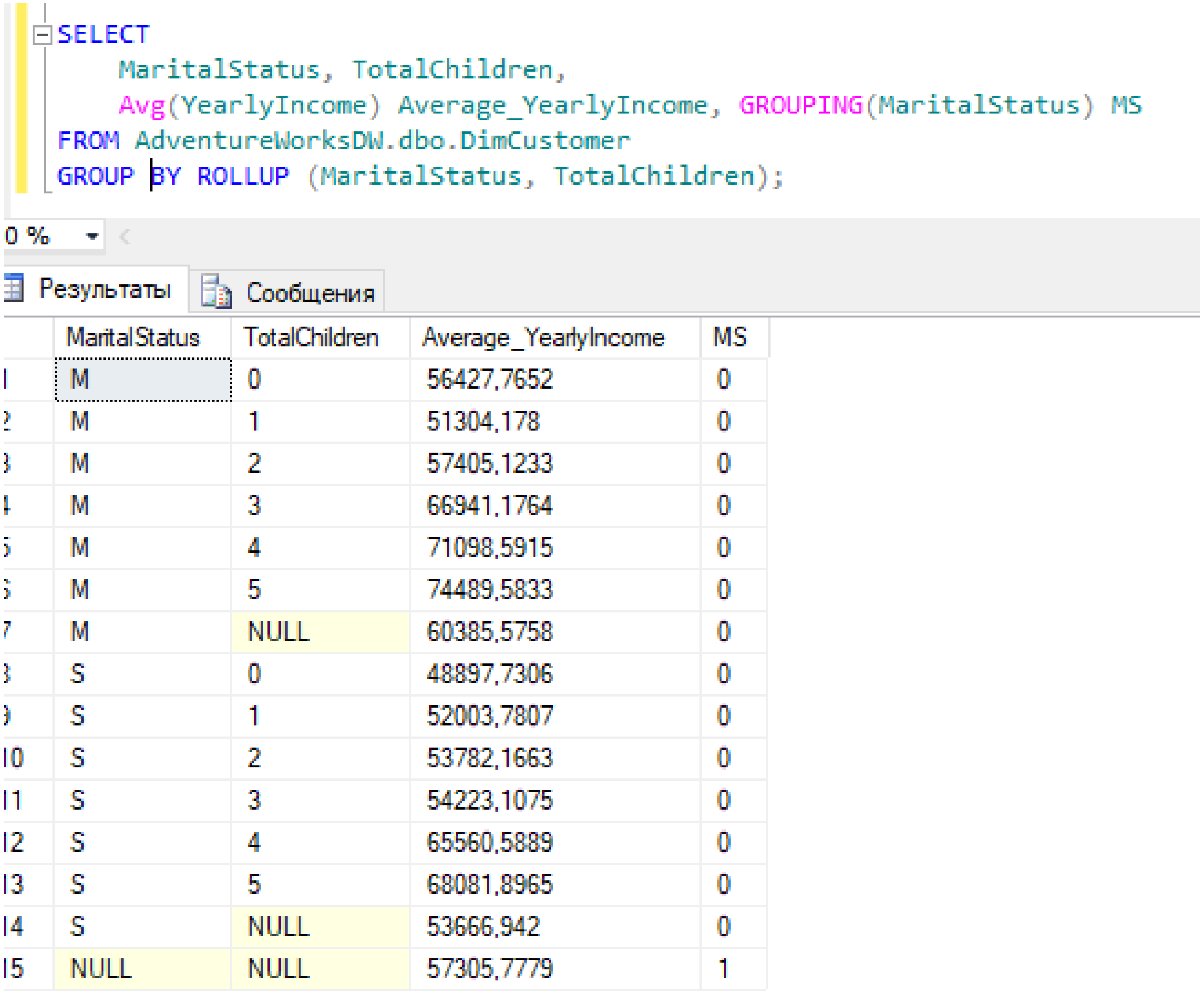
```
SELECT  division_id,
        grade,
        group_id(),
        SUM(salary)
FROM employees E JOIN salgrade S
ON E.salary BETWEEN S.low_sal AND S.high_sal
GROUP BY
GROUPING SETS((grade,division_id),
              division_id, grade));
```



	DIVISION_ID	GRADE	GROUP_ID()	SUM(SALARY)
1	BUS	2	0	280000
2	SAL	3	0	1410000
3	BUS	1	0	530000
4	SAL	2	0	3386000
5	OPE	1	0	635000
6	OPE	2	0	685000
7	SUP	2	0	265000
8	(null)	1	0	200000
9	SUP	1	0	750000
10	SAL	1	0	340000
11	BUS	4	0	800000
12	BUS	2	1	280000
13	SAL	3	1	1410000
14	BUS	1	1	530000
15	SAL	2	1	3386000
16	OPE	1	1	635000
17	OPE	2	1	685000
18	SUP	2	1	265000
19	(null)	1	1	200000
20	SUP	1	1	750000
21	SAL	1	1	340000
22	BUS	4	1	800000

- Устранение полного дублирования групп

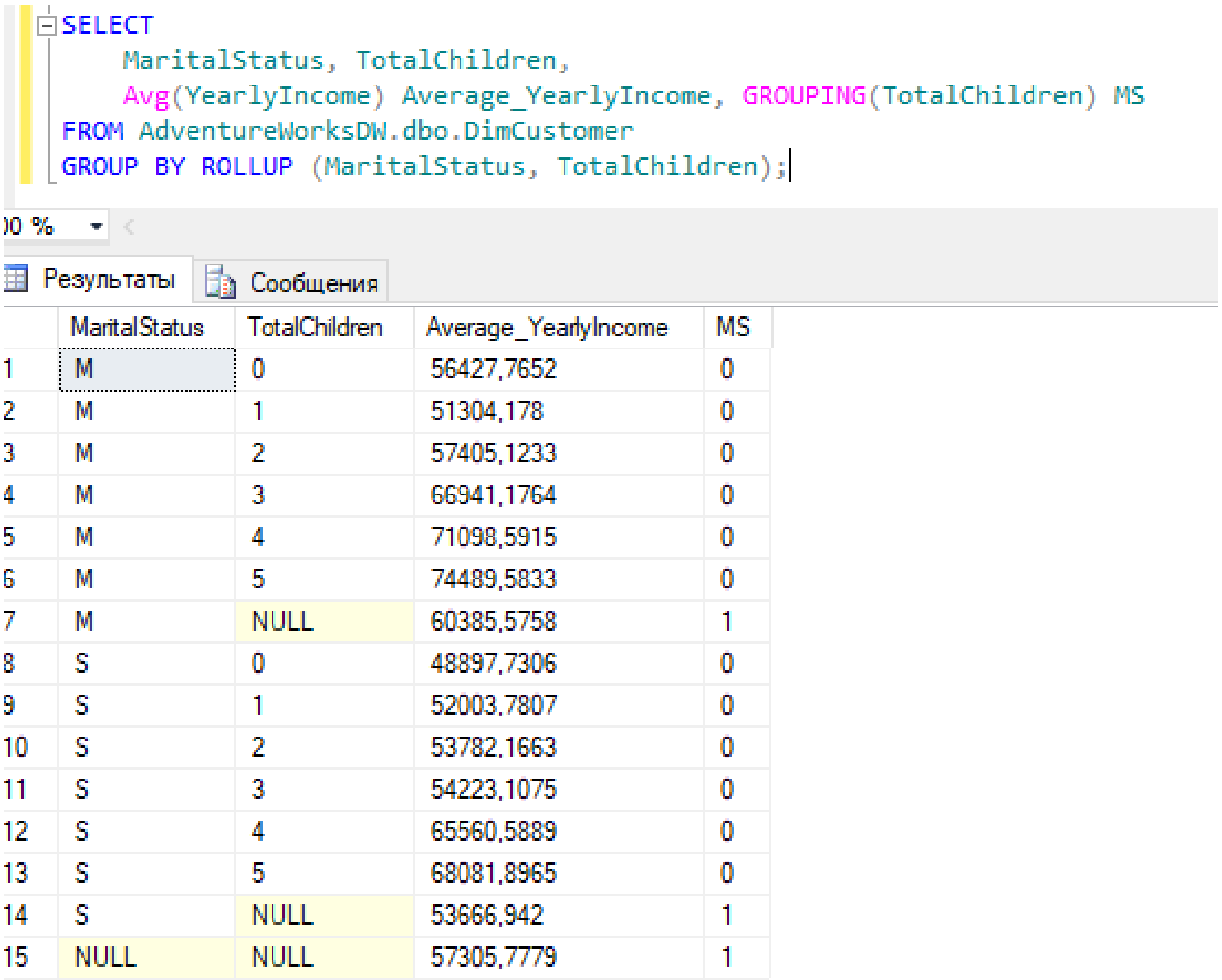
GROUPING



```
SELECT
    MaritalStatus, TotalChildren,
    Avg(YearlyIncome) Average_YearlyIncome, GROUPING(MaritalStatus) MS
FROM AdventureWorksDW.dbo.DimCustomer
GROUP BY ROLLUP (MaritalStatus, TotalChildren);
```

	MaritalStatus	TotalChildren	Average_YearlyIncome	MS
1	M	0	56427,7652	0
2	M	1	51304,178	0
3	M	2	57405,1233	0
4	M	3	66941,1764	0
5	M	4	71098,5915	0
6	M	5	74489,5833	0
7	M	NULL	60385,5758	0
8	S	0	48897,7306	0
9	S	1	52003,7807	0
10	S	2	53782,1663	0
11	S	3	54223,1075	0
12	S	4	65560,5889	0
13	S	5	68081,8965	0
14	S	NULL	53666,942	0
15	NULL	NULL	57305,7779	1

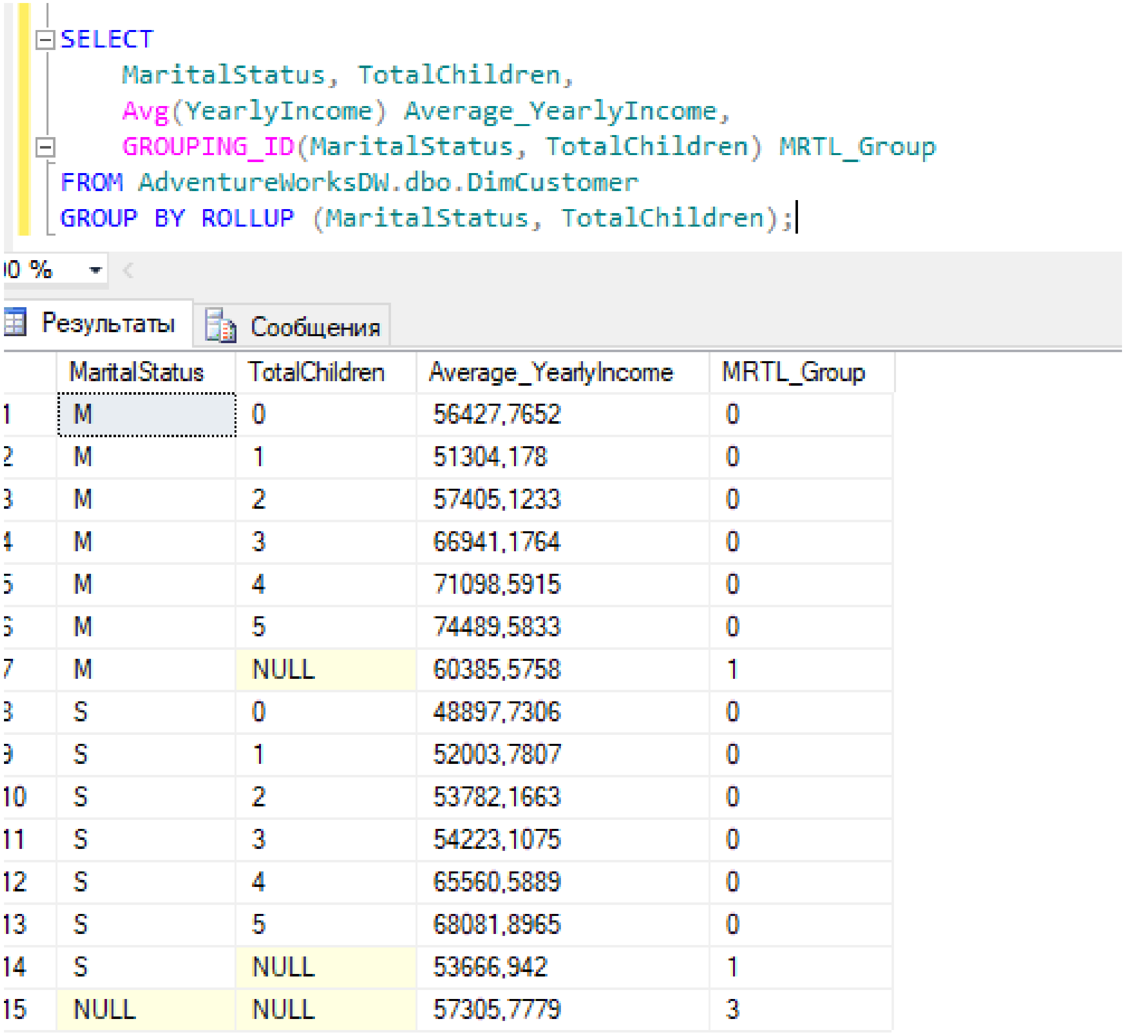
GROUPING



```
SELECT
    MaritalStatus, TotalChildren,
    Avg(YearlyIncome) Average_YearlyIncome, GROUPING(TotalChildren) MS
FROM AdventureWorksDW.dbo.DimCustomer
GROUP BY ROLLUP (MaritalStatus, TotalChildren);
```

	MaritalStatus	TotalChildren	Average_YearlyIncome	MS
1	M	0	56427,7652	0
2	M	1	51304,178	0
3	M	2	57405,1233	0
4	M	3	66941,1764	0
5	M	4	71098,5915	0
6	M	5	74489,5833	0
7	M	NULL	60385,5758	1
8	S	0	48897,7306	0
9	S	1	52003,7807	0
10	S	2	53782,1663	0
11	S	3	54223,1075	0
12	S	4	65560,5889	0
13	S	5	68081,8965	0
14	S	NULL	53666,942	1
15	NULL	NULL	57305,7779	1

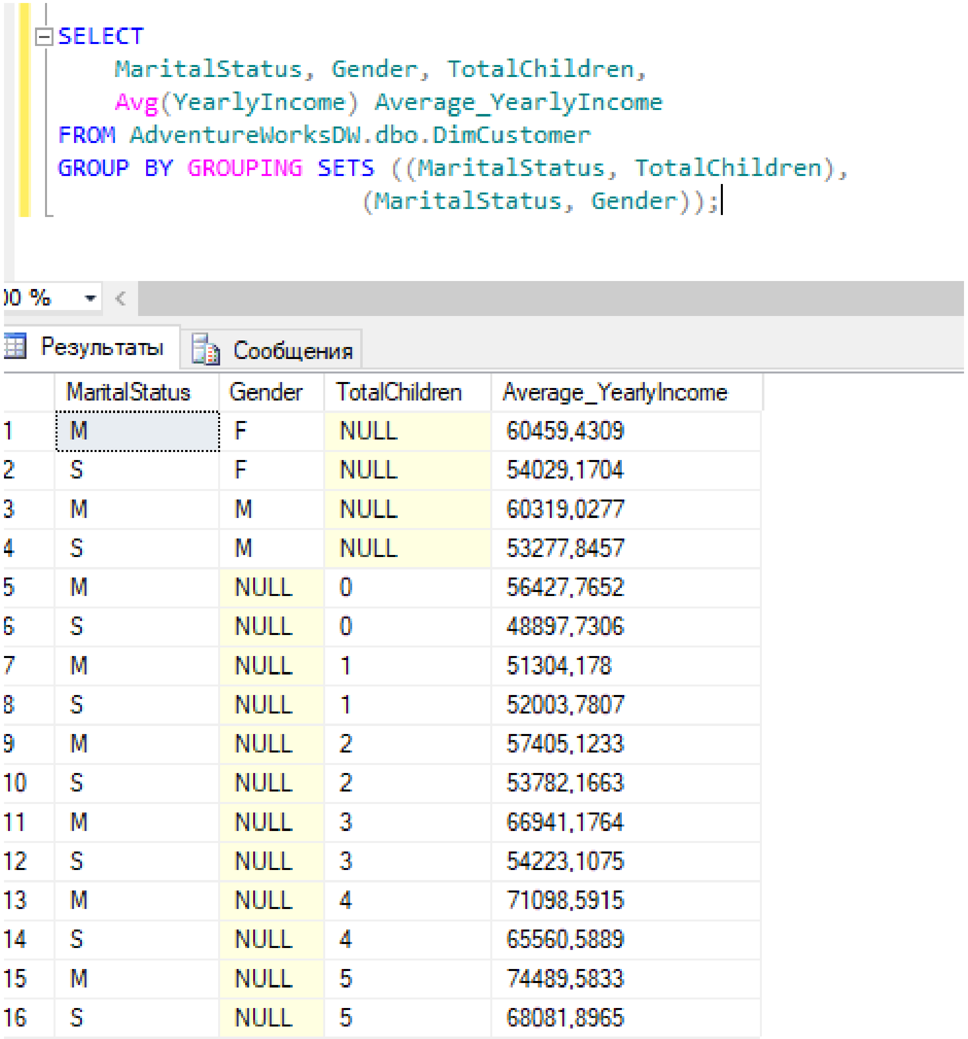
GROUPING_ID



```
SELECT
    MaritalStatus, TotalChildren,
    Avg(YearlyIncome) Average_YearlyIncome,
    GROUPING_ID(MaritalStatus, TotalChildren) MRTL_Group
FROM AdventureWorksDW.dbo.DimCustomer
GROUP BY ROLLUP (MaritalStatus, TotalChildren);
```

	MaritalStatus	TotalChildren	Average_YearlyIncome	MRTL_Group
1	M	0	56427,7652	0
2	M	1	51304,178	0
3	M	2	57405,1233	0
4	M	3	66941,1764	0
5	M	4	71098,5915	0
6	M	5	74489,5833	0
7	M	NULL	60385,5758	1
8	S	0	48897,7306	0
9	S	1	52003,7807	0
10	S	2	53782,1663	0
11	S	3	54223,1075	0
12	S	4	65560,5889	0
13	S	5	68081,8965	0
14	S	NULL	53666,942	1
15	NULL	NULL	57305,7779	3

GROUPING SETS



```
SELECT
    MaritalStatus, Gender, TotalChildren,
    Avg(YearlyIncome) Average_YearlyIncome
FROM AdventureWorksDW.dbo.DimCustomer
GROUP BY GROUPING SETS ((MaritalStatus, TotalChildren),
                        (MaritalStatus, Gender));
```

	MaritalStatus	Gender	TotalChildren	Average_YearlyIncome
1	M	F	NULL	60459,4309
2	S	F	NULL	54029,1704
3	M	M	NULL	60319,0277
4	S	M	NULL	53277,8457
5	M	NULL	0	56427,7652
6	S	NULL	0	48897,7306
7	M	NULL	1	51304,178
8	S	NULL	1	52003,7807
9	M	NULL	2	57405,1233
10	S	NULL	2	53782,1663
11	M	NULL	3	66941,1764
12	S	NULL	3	54223,1075
13	M	NULL	4	71098,5915
14	S	NULL	4	65560,5889
15	M	NULL	5	74489,5833
16	S	NULL	5	68081,8965

Вопросы?