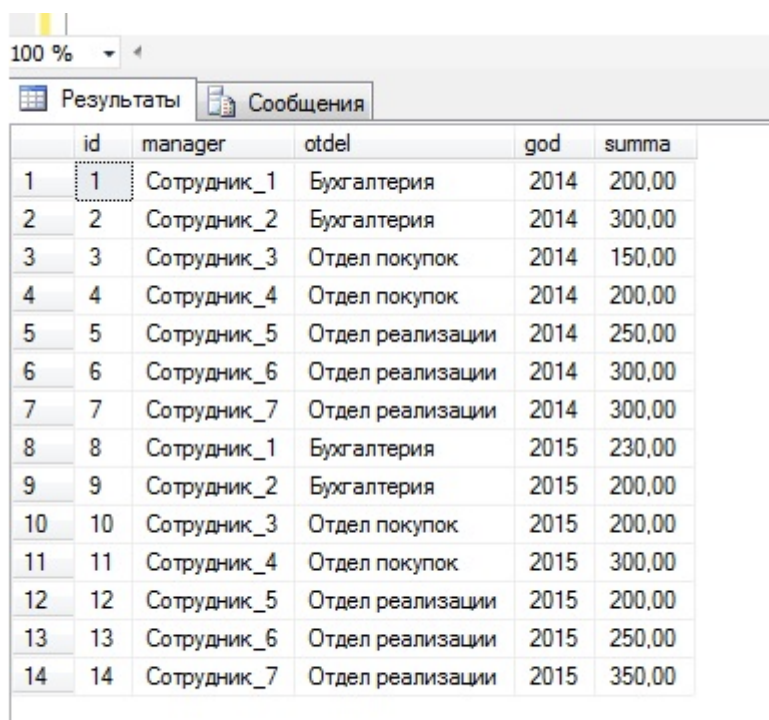


## Управляемая Самостоятельная работа студентов

### ROLLUP, CUBE и GROUPING SETS операторы Transact-SQL для подведения итогов и промежуточных итогов

#### Исходные данные для примеров

1. Создать таблицу, которая будет содержать список сотрудников с указанием отдела, в котором они работают, а также сумму их заработка по годам.
2. Ввести данные:



The screenshot shows a SQL Server Enterprise Manager window with a table named 'Employees' (implied) containing 14 rows of data. The columns are 'id', 'manager', 'otdel', 'god', and 'summa'. The data is as follows:

	id	manager	otdel	god	summa
1	1	Сотрудник_1	Бухгалтерия	2014	200,00
2	2	Сотрудник_2	Бухгалтерия	2014	300,00
3	3	Сотрудник_3	Отдел покупок	2014	150,00
4	4	Сотрудник_4	Отдел покупок	2014	200,00
5	5	Сотрудник_5	Отдел реализации	2014	250,00
6	6	Сотрудник_6	Отдел реализации	2014	300,00
7	7	Сотрудник_7	Отдел реализации	2014	300,00
8	8	Сотрудник_1	Бухгалтерия	2015	230,00
9	9	Сотрудник_2	Бухгалтерия	2015	200,00
10	10	Сотрудник_3	Отдел покупок	2015	200,00
11	11	Сотрудник_4	Отдел покупок	2015	300,00
12	12	Сотрудник_5	Отдел реализации	2015	200,00
13	13	Сотрудник_6	Отдел реализации	2015	250,00
14	14	Сотрудник_7	Отдел реализации	2015	350,00

### 3. ROLLUP

**ROLLUP – оператор Transact-SQL, который формирует промежуточные итоги для каждого указанного элемента и общий итог.**

**3.1** Получить сумму расхода на оплату труда по отделам и по годам без использования оператора ROLLUP

**3.2** Получить общий расход по каждому отделу и годам, используя оператор ROLLUP, объяснить.

**3.3.** Использовать оператор ROLLUP с группировкой по одному полю, например, группировка по отделам с общим итогом

**3.4** Использовать оператор ROLLUP с группировкой по одному полю, например, группировка по годам с общим итогом

**4. CUBE — оператор Transact-SQL, который формирует результаты для всех возможных перекрестных вычислений.**

**4.1.** Напишите такой же SQL запрос как в 3.2, только вместо ROLLUP укажите CUBE и объясните различие.

5. **GROUPING SETS** – оператор Transact-SQL, который формирует результаты нескольких группировок в один набор данных.

5.1 Напишите такой же SQL запрос как в 3.2, только вместо rollup укажите **GROUPING SETS** и объясните различие.

6. **GROUPING** – функция Transact-SQL, которая возвращает истину, если указанное выражение является статистическим, и ложь, если выражение нестатистическое. Данная функция создана для того, чтобы отличить статистические строки, которые добавил SQL сервер, от строк, которые и есть сами данные, так как когда используешь много группировок, запутаться в строках очень легко.

Выполните ниже пример, объясните результат.

```
select otdel,
       ISNULL(cast(god as varchar(30)),
              case when GROUPING(god)=1 and GROUPING(otdel)=0
                    then 'Промежуточный итог' else 'Общий итог' end) as god,
       SUM(summa) as itog,
       GROUPING(otdel) as grouping_otdel,
       GROUPING(god) as grouping_god
from dbo.test_table
group by
rollup (otdel,god)
```

100 %

	otdel	god	itog	grouping_otdel	grouping_god
1	Бухгалтерия	2014	500,00	0	0
2	Бухгалтерия	2015	430,00	0	0
3	Бухгалтерия	Промежуточный итог	930,00	0	1
4	Отдел покупок	2014	350,00	0	0
5	Отдел покупок	2015	500,00	0	0
6	Отдел покупок	Промежуточный итог	850,00	0	1
7	Отдел реализации	2014	850,00	0	0
8	Отдел реализации	2015	800,00	0	0
9	Отдел реализации	Промежуточный итог	1650,00	0	1
10	NULL	Общий итог	3430,00	1	1