

4.1. Написать запрос, выдающий отчёт о суммарных выплатах сотрудникам, непосредственно подчиняющимся заданному руководителю по идентификаторам должностей (поле Job\_id). Непосредственное подчинение предполагает подчинение на первом уровне. Номер каждого руководителя должен встречаться в отчете лишь дважды.

Пример результата:

Номер руководителя	Должность	Кол-во сотрудников	Выплаты	Вид выплаты
100	AD_VP	2	34000	Зарплата сотрудников в должности AD_VP
	MK_MAN	1	13000	Зарплата сотрудников в должности MK_MAN
	PU_MAN	1	11000	Зарплата сотрудников в должности PU_MAN
	SA_MAN	5	79650	Зарплата сотрудников в должности SA_MAN

4.1

```
with tab1 as
(select manager_id, job_id, count_emp, salary_man, rownum row_n
from (select manager_id, job_id, count(*) count_emp, sum(salary) salary_man
from employees
where manager_id is not null
group by rollup (manager_id, job_id)
order by manager_id)),
tab2 as
(select manager_id, max(row_n) max_row, min(row_n) min_row
from tab1
group by manager_id)
select case
when (row_n in (select min_row from tab2)) and (row_n in (select max_row
from tab2)) then ' '
when (row_n in (select min_row from tab2)) or (row_n in (select max_row from
tab2)) then to_char(manager_id)
else ' '
end "Номер руководителя",
case
when job_id is null then ' '
else job_id
end "Должность", count_emp "Кол-во сотрудников", salary_man
"Выплаты",
```

```

case
when row_n=(select max(max_row)from tab2) then 'Общий итог'
else case
when row_n in (select max_row from tab2) then 'Суммарная зарплата у
руководителя '||to_char(manager_id)
else 'Зарплата сотрудников в должности '||to_char(job_id)
end
end "Вид выплаты"
from tab1;

```

Сначала с помощью Rollup получаем отчет, где есть руководитель, каждая должность и количество сотрудников на ней, а также суммарное значение по руководителю. Далее ограничиваем первую и последнюю строку, относящуюся к данному руководителю через rownum. Впоследствии используем это, чтобы выполнить условие, по которому номер руководителя встречается в отчете лишь дважды.

#### 4.4. Вывести все даты за 2013 год и соответствующие дни недели без использования иерархических запросов и Model.

Пример результата:

Date	Day
01.01.2011	Суббота
02.01.2011	Воскресенье
03.01.2011	Понедельник
...	...
29.12.2011	Четверг
30.12.2011	Пятница
31.12.2011	Суббота

## 4.4

```

with tmp1 as (
select 'я не должен использовать иерархические запросы и Model' from dual union all
select 'я не должен использовать иерархические запросы и Model' from dual union all
select 'я не должен использовать иерархические запросы и Model' from dual union all
select 'я не должен использовать иерархические запросы и Model' from dual union all
select 'я не должен использовать иерархические запросы и Model' from dual union all
select 'я не должен использовать иерархические запросы и Model' from dual union all
select 'я не должен использовать иерархические запросы и Model' from dual union all
select 'я не должен использовать иерархические запросы и Model' from dual),
tmp2 as (select * from tmp1 cross join tmp1 cross join tmp1)
select to_char((TO_DATE('01.01.2013', 'dd.mm.yyyy') + rownum - 1), 'dd.mm.yyyy') as "Дата",
to_char((TO_DATE('01.01.2013', 'dd.mm.yyyy') + rownum - 1), 'Day', 'NLS_DATE_LANGUAGE=RUSSIAN') as "День недели"
from tmp2
where (TO_DATE('01.01.2013', 'dd.mm.yyyy') + rownum - 1) < ADD_MONTHS(TO_DATE('01.01.2013', 'dd.mm.yyyy'), 12);

```

Трижды соединяем таблицу с собой и используем rownum.

4.2.<sup>1</sup> Создать директорию Init для следующего пути:  
C:\oracleexe\app\oracle\product\11.2.0\server\config\scripts\.<sup>2</sup>

Создать внешнюю таблицу для чтения данных из текстового файла init.ora, расположенного в данном каталоге. Таблица должна содержать два столбца: Имя параметра и Значение.

Создать запрос к созданной внешней таблице, который позволит найти общее количество файлов управления (Control\_Files).

Пример результата:

4.2

Ctl_files_cnt
1

```
create table Init("Name" varchar2(250),"Value" varchar2(250))
organization external
(type oracle_loader
default directory Stud
access parameters
(records delimited by newline
nobadfile
nologfile
fields terminated by '='
missing field values are null
reject rows with all null fields)
location ('init.ora'))
reject limit unlimited;
select * from init;
drop table init;
SELECT (
LENGTH("Value")-LENGTH(REPLACE("Value", ""))
)/2 CTL_FILES_CNT
```

**FROM INit**

**WHERE LOWER("Name") = 'control\_files' AND "Value" IS NOT NULL;**

Создаем внешнюю таблицу, учитывая, что у нас названия и значения разделены знаком “=”. Далее обращаемся к таблице и находим количество двойных кавычек в строке control files, делим их количество на два, поскольку каждый файл ограничен одной парой таких кавычек.

4.3.<sup>3</sup> Создать директорию Alert для следующего пути:  
C:\oracle\xe\app\oracle\diag\rdbms\xe\xe\trace\2.<sup>4</sup>

Создать внешнюю таблицу для чтения данных из текстового файла alert\_xe.log, расположенного в данном каталоге.

Создать запрос к созданной внешней таблице, позволяющий получать информацию об ошибках Oracle за определенный день месяца.

**4.3**

**create table Alert\_Xe(Text varchar2(1000))**

**organization external**

**(type oracle\_loader**

**default directory Stud**

**access parameters**

**(records delimited by newline**

**nobadfile**

**nologfile**

**)**

**location ('alert\_xe.log'))**

**reject limit unlimited;**

**select \* from Alert\_Xe;**

**drop table Alert\_Xe;**

**Select "Date","Error\_text"**

**from (SELECT**

**TO\_CHAR(MAX(**

```
TO_DATE("Text1", 'Dy Mon DD HH24:MI:SS YYYY',  
'nls_date_language=american')
```

```
), 'DD.MM.YYYY HH24:MI:SS') "Date",
```

```
"Text2" "Error_text"
```

```
from
```

```
(select *
```

```
from ((select rownum "id1", text "Text1"
```

```
from Alert_Xe
```

```
where Text like '_____:__:____'
```

```
or Text like 'ORA-%') er_date
```

```
join
```

```
(select rownum "id2", text "Text2"
```

```
from Alert_Xe
```

```
where Text like '_____:__:____'
```

```
or Text like 'ORA-%') er_text
```

```
on (er_date."id1"<er_text."id2")) Tab1
```

```
where "Text1" like '_____:__:____'
```

```
and "Text2" like 'ORA-%'
```

```
group by "id2","Text2") Tab2
```

```
where substr(to_char("Date"),1,10)='&Dat'
```

```
order by 1;
```

Сначала получаю просто все данные едином блоком, далее разбиваю на даты и текст ошибок, далее соединяю, чтобы получить два столбца - соответствие дат и ошибок. После этого проверяю равенство введенной даты и той, которая была введена для получения результата.

## 4.5

### 4.5. Создать<sup>1</sup> директорию (объект базы данных Directory) для каталога файловой системы, содержащего текстовый файл с информацией в виде:

Номер	ФИО	Дата	Оценка	Дисциплина
1	Петров С.С.	12-05-2008	4	Математика
2	Сомов Л.Л.	4-05-2008	5	Физика
3	Амосов Д.Г.	3-05-2008	4	Физика

Создать внешнюю таблицу для чтения данных из этого файла.

```
DROP TABLE MARKS PURGE;
```

```
CREATE TABLE MARKS
```

```
(GB_NUMBER number(1),
```

```
GB_FULLNAME varchar(17),
```

```
GB_DATE DATE,
```

```
GB_GRADE number(1),
```

```
GB_SUBJECT varchar(15))
```

```
ORGANIZATION EXTERNAL
```

```
(TYPE ORACLE_LOADER
```

```
DEFAULT DIRECTORY STUD
```

```
ACCESS PARAMETERS
```

```
(RECORDS DELIMITED BY NEWLINE
```

```
SKIP 1
```

```
NOLOGFILE
```

```
NOBADFILE
```

```
FIELDS TERMINATED BY WHITESPACE
```

```
MISSING FIELD VALUES ARE NULL
```

```
REJECT ROWS WITH ALL NULL FIELDS
```

```
(GB_NUMBER position (1-1),
```

```
GB_FULLNAME position (15-30),
```

```
GB_DATE position (31-40) date "dd-mm-yyyy",
```

```
GB_GRADE position (49-52),
```

*GB\_SUBJECT position (64-74)))*

*LOCATION('marks.txt'))*

*REJECT LIMIT 0;*

Создается внешняя таблица из 5 столбцов с соответствующими типами данных, первая строка пропускается - поля разделены пробелом. Далее указаны позиции соответствующих полей, можно допустить 1 ошибку, это первая строка - не подходящая под формат.

4.6. Показать в одном отчете для каждого отдела: его номер, наименование, количество работающих сотрудников, средний оклад вместе со следующими данными по каждому сотруднику — фамилия, оклад и должность.

Пример результата:

Номер отдела	Название отдела	Кол-во сотрудников	Средний оклад	Фамилия
30	Purchasing	6	4150	
				Raphaely
				Khoo
				Baida
				Tobias
				Himuro
				Colmenares
40	HumanRe sources	1	6500	
				Mavris
50	Shipping	45	3475,56	
				Weiss
				Fripp
				Kaufling
...	...	...	...	...

## 4.6

with tab

as ((select department\_id,department\_id "Номер отдела",  
department\_name "Название отдела",count(employee\_id) "Кол-во  
сотрудников",round(avg(salary),2) "Средний оклад",null "Фамилия", 1  
Flag,null "Оклад", null "Должность"

from employees join departments using(department\_id) group by  
department\_id, department\_name

union

select department\_id,null,null,null,null,last\_name, 0 flag,salary "Оклад",  
job\_title "Должность"

```

from employees join jobs using(job_id))

order by 1,8 desc)

select nvl(to_char("Номер отдела"),' ') "Номер отдела",nvl("Название
отдела",' ') "Название отдела",

nvl(to_char("Кол-во сотрудников"),' ') "Кол-во
сотрудников",nvl(to_char("Средний оклад"),' ') "Средний оклад",

nvl("Фамилия",' ') "Фамилия",nvl(to_char("Оклад"),'
') "Оклад",nvl("Должность",' ') "Должность"

from tab

```

Можно заметить, что результат аккуратно делится на две части, в целом к отделу и должности, и к конкретному сотруднику. Найду каждую часть отдельно, потом соединю, правильную сортировку обеспечу дополнительным полем flag, которое отвечает за то, к чему относится очередная строка.



4.7. Сформировать отчёт, содержащий номер отдела, название отдела, имена и фамилии сотрудников и общее кол-во сотрудников в каждом отделе. Сортировка — по названию отдела и имени и фамилии сотрудника. Также в отчёте должен присутствовать столбец со сквозной нумерацией сотрудников, при этом строки, содержащие общее кол-во сотрудников (в т. ч. по отделам) нумероваться не должны.

Пример результата:

Номер п/п	Идентификатор отдела	Название отдела	Сотрудник
1	110	Accounting	Shelley Higgins
2	110	Accounting	William Gietz
	Кол-во сотрудников в отделе	Accounting	2
3	10	Administration	Clark Kent

33

Номер п/п	Идентификатор отдела	Название отдела	Сотрудник
4	10	Administration	Jennifer Whalen
	Кол-во сотрудников в отделе	Administration	2
...	...	...	...
110	50	Shipping	Vance Jones
111	50	Shipping	Winston Taylor
	Кол-во сотрудников в отделе	Shipping	45
	Общее кол-во сотрудников в отделах		111

4.7

```
select nvl(to_char("Номер п/п"),' ') "Номер
п/п",nvl(to_char("Идентификатор отдела"),' ') "Идентификатор отдела",
      nvl(to_char("Название отдела"),' ') "Название
отдела",to_char("Сотрудник") "Сотрудник"
from (select *
```

```

from (select * from (select rownum "Номер п/п",tab.*,0 flag
from (select to_char(department_id) "Идентификатор
отдела",department_name "Название отдела",first_name||' '||last_name
"Сотрудник"
from employees left join departments using (department_id)
order by 2,3) tab
union
select null,'Количество сотрудников в
отделе',department_name,to_char(count(employee_id)),1 flag
from employees left join departments using (department_id)
group by department_name)
union
select null,'Общее число сотрудников',null,to_char(count(employee_id)), 2
flag from employees
order by 3))

```

Разделим выходные строки на 3 типа частей: для сотрудника, для отдела и для общего - соединим через union. Планирую проделать еще раз через функцию grouping без использования доп столбца flag

---

4.8. Сформировать отчёт, содержащий номер отдела, название отдела, имена и фамилии сотрудников, а также их оклады, отсортированные по возрастанию. Отчёт должен иметь следующий вид:

FIRST_NAME	LAST_NAME	SALARY
Отдел №	10	«Administration»
Jennifer	Whalen	4400
Clark	Kent	1000
Отдел №	20	«Marketing»
Max	Smart	1000
Pat	Fay	6000
Michael	Harstein	13000
...	...	...
Отдел №	110	«Accounting»
William	Gietz	8300
Shelley	Higgins	12000

Сотрудников, не приписанных к конкретному отделу, не

4.8 выводить.

SELECT

```

CASE
WHEN GROUPING(Emp.Salary) = 1
THEN 'Отдел №'
ELSE Emp.First_Name
END AS "FIRST_NAME",
CASE
WHEN GROUPING(Emp.Salary) = 1
THEN TO_CHAR(Dep.Department_ID)
ELSE Emp.Last_Name
END AS "LAST_NAME" ,
CASE
WHEN GROUPING(Emp.Salary) = 1
THEN '«' || Dep.Department_Name || '»'
ELSE TO_CHAR(Emp.Salary)
END AS "SALARY"
FROM Employees Emp
JOIN Departments Dep
ON Emp.Department_ID = Dep.Department_ID
GROUP BY
GROUPING SETS (
(Dep.Department_ID, Dep.Department_Name),
(Dep.Department_ID, Dep.Department_Name,
Emp.Employee_ID, Emp.First_Name, Emp.Last_Name, Emp.Salary)
)
ORDER BY Dep.Department_ID, GROUPING(Emp.Salary) DESC;

```