

Лабораторная работа №21

Оператор MODEL предложения SELECT

Цель работы

1. Изучение основных возможностей оператора MODEL.
2. Приобретение практических навыков написания SQL запросов с использованием оператора MODEL.

Теоретические сведения

- MODEL
- DIMENSION BY
- DIMENSION и MEASURES
- RULES
- Основные правила
- FOR Loops
- UNIQUE SINGLE REFERENCE
- Функция CV()
- PARTITION BY
- Агрегатные и аналитические функции

MODEL

В данной лабораторной работе в качестве примера используется демонстрационная база данных SCOT TIGER. Данную БД можно установить в облачной системе APEX.ORACLE.COM. Также имеется возможность работы с этой БД на сайте LIVESQL.ORACLE.COM.

Конструкция MODEL появилась в версии Oracle 10g. Описывается как мощное средство, которое дает доступ к изменению любой ячейки запроса в результирующем наборе данных. И так же добавлены новые возможности, ранее доступные только через PL/SQL вызовы. MODEL – выражение позволяющее представлять результат запроса в виде многомерного куба и задавать выражения для расчета его произвольных ячеек. Например, с помощью MODEL, можно взять следующую таблицу:

```
select * from scott.emp t ;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17.12.1980	800,00		20
7499	ALLEN	SALESMAN	7698	20.02.1981	1600,00	300,00	30
7521	WARD	SALESMAN	7698	22.02.1981	1250,00	500,00	30
7566	JONES	MANAGER	7839	02.04.1981	2975,00		20
7654	MARTIN	SALESMAN	7698	28.09.1981	1250,00	1400,00	30
7698	BLAKE	MANAGER	7839	01.05.1981	2850,00		30
7782	CLARK	MANAGER	7839	09.06.1981	2450,00		10
7788	SCOTT	ANALYST	7566	09.12.1982	3000,00		20
7839	KING	PRESIDENT		17.11.1981	5000,00		10
7844	TURNER	SALESMAN	7698	08.09.1981	1500,00	0,00	30
7876	ADAMS	CLERK	7788	12.01.1983	1100,00		20
7900	JAMES	CLERK	7698	03.12.1981	950,00		30
7902	FORD	ANALYST	7566	03.12.1981	3000,00		20
7934	MILLER	CLERK	7782	23.01.1982	1300,00		10

и добавив ряд команд, создать новые строки с итогами:

```
select case when key like 'Итог%' then key end key,
        ename, job, hiredate, sal from scott.emp t
model
dimension by ( cast (empno as varchar2(40)) as key,
               ename,
               job,
               hiredate
             ) measures (sal) rules
(
  sal['Итог 1: ename like 'S%''',null,null,null]
    = sum(sal)[any,ename like 'S%',any,any],
  sal['Итог 2: salesman; сентябрь',null,null,null]
    = sum(sal)[any,any,'SALESMAN',extract(month from
hiredate)=9],
  sal['Итог 3: length(ename)=5;
декабрь',null,null,null]
    = sum(sal)[any,length(ename)=5,any,extract(month
from hiredate)=12]
)order by key NULLS FIRST, hiredate, ename;
```

KEY	ENAME	JOB	HIREDATE	SAL
	SMITH	CLERK	17.12.1980	800
	ALLEN	SALESMAN	20.02.1981	1600
	WARD	SALESMAN	22.02.1981	1250
	JONES	MANAGER	02.04.1981	2975
	BLAKE	MANAGER	01.05.1981	2850
	CLARK	MANAGER	09.06.1981	2450
	TURNER	SALESMAN	08.09.1981	1500
	MARTIN	SALESMAN	28.09.1981	1250
	KING	PRESIDENT	17.11.1981	5000
	FORD	ANALYST	03.12.1981	3000
	JAMES	CLERK	03.12.1981	950
	MILLER	CLERK	23.01.1982	1300
	SCOTT	ANALYST	09.12.1982	3000
	ADAMS	CLERK	12.01.1983	1100
Итог 1: ename like 'S%'				3800
Итог 2: salesman; сентябрь				2750
Итог 3: length(ename)=5; декабрь				4750

Можно использовать процедурные свойства MODEL для получения результата, который сложно, неэффективно или невозможно реализовать в SQL запросе (без PL/SQL).

```
select job , substr( str, 2 ) as str from scott.emp
t model
return updated rows
partition by (job)dimension by ( row_number() over
(partition by job order by job) as position)measures
(cast(sal as varchar2(200)) as str)rules upsert
```

```

        iterate( 4 )until (
presentv(str[iteration_number+2],1,0) = 0 )
        ( str[0] = str[0] || ',' ||
str[iteration_number+1]
        )order by job;

```

JOB	STR
ANALYST	3000,3000
CLERK	1300,950,800,1100
MANAGER	2850,2975,2450
PRESIDENT	5000
SALESMAN	1500,1250,1250,1600

Условия MODEL выполняются после всех условий запроса, исключая SELECT DISTINCT и ORDER BY. Далее рассмотрим возможности MODEL более подробно.

DIMENSION BY

DIMENSION BY определяет столбцы SQL запроса, с помощью которых для любой ячейки можно сопоставить единственную строку. В примерах показаны примеры использования DIMENSION BY для лучшего понимания. Посмотрим на таблицу EMP

```
select * from scott.emp t ;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17.12.1980	800,00		20
7499	ALLEN	SALESMAN	7698	20.02.1981	1600,00	300,00	30
7521	WARD	SALESMAN	7698	22.02.1981	1250,00	500,00	30
7566	JONES	MANAGER	7839	02.04.1981	2975,00		20
7654	MARTIN	SALESMAN	7698	28.09.1981	1250,00	1400,00	30
7698	BLAKE	MANAGER	7839	01.05.1981	2850,00		30
7782	CLARK	MANAGER	7839	09.06.1981	2450,00		10
7788	SCOTT	ANALYST	7566	09.12.1982	3000,00		20
7839	KING	PRESIDENT		17.11.1981	5000,00		10
7844	TURNER	SALESMAN	7698	08.09.1981	1500,00	0,00	30
7876	ADAMS	CLERK	7788	12.01.1983	1100,00		20
7900	JAMES	CLERK	7698	03.12.1981	950,00		30
7902	FORD	ANALYST	7566	03.12.1981	3000,00		20
7934	MILLER	CLERK	7782	23.01.1982	1300,00		10

В качестве уникального идентификатора используется EMPNO. Значит, это потенциальный столбец для dimension. Не обращаем пока внимание на MEASURES и RULES условия.

```

select empno, ename from scott.emp t
model
dimension by (empno)
measures (ename)
rules ( )
order by empno;

```

EMPNO	ENAME
7369	SMITH
7499	ALLEN
7521	WARD
7566	JONES
7654	MARTIN
7698	BLAKE
7782	CLARK
7788	SCOTT
7839	KING
7844	TURNER
7876	ADAMS
7900	JAMES
7902	FORD
7934	MILLER

Если необходимо, можно определять более чем одну колонку.

```
select empno, job, ename from scott.emp t
model
dimension by (empno, job)
measures (ename)
rules ()
order by empno;
```

EMPNO	JOB	ENAME
7369	CLERK	SMITH
7499	SALESMAN	ALLEN
7521	SALESMAN	WARD
7566	MANAGER	JONES
7654	SALESMAN	MARTIN
7698	MANAGER	BLAKE
7782	MANAGER	CLARK
7788	ANALYST	SCOTT
7839	PRESIDENT	KING
7844	SALESMAN	TURNER
7876	CLERK	ADAMS
7900	CLERK	JAMES
7902	ANALYST	FORD
7934	CLERK	MILLER

14 rows selected

Нельзя использовать алиасы из SELECT условия.

```
select empno as id, ename from scott.emp t
model
dimension by (id)
measures (ename)
rules ()
order by empno;
```

ORA-00904: "ID": недопустимый идентификатор

```
select empno * 10 as id, ename from scott.emp t
model
```

```

dimension by (id)
measures (ename)
rules ()
order by empno;

```

ORA-00904: "ID": недопустимый идентификатор

Можно использовать алиасы напрямую в dimension by.

```

select id, ename from scott.emp t
model
dimension by (empno * 10 as id)
measures (ename)
rules ()
order by id;

```

ID	ENAME
73690	SMITH
74990	ALLEN
75210	WARD
75660	JONES
76540	MARTIN
76980	BLAKE
77820	CLARK
77880	SCOTT
78390	KING
78440	TURNER
78760	ADAMS
79000	JAMES
79020	FORD
79340	MILLER

По умолчанию, если dimension by не дает уникального ключа для определения строки, то получаем ошибку.

```

select job, ename from scott.emp t
model
dimension by (job)
measures (ename)
rules ()
order by job;

```

ORA-32638: Неуникальная адресация в измерениях MODEL

Но можно ослабить это правило, используя **UNIQUE SINGLE REFERENCE**

```

select job, ename from scott.emp t
model UNIQUE SINGLE REFERENCE
dimension by (job)
measures (ename)
rules ()
order by job;

```

JOB	ENAME
ANALYST	SCOTT
ANALYST	FORD
CLERK	MILLER
CLERK	JAMES

CLERK	SMITH
CLERK	ADAMS
MANAGER	BLAKE
MANAGER	JONES
MANAGER	CLARK
PRESIDENT	KING
SALESMAN	TURNER
SALESMAN	MARTIN
SALESMAN	WARD
SALESMAN	ALLEN

Но данное условие, накладывает ряд ограничений по доступу к ячейкам запроса. (рассмотрим далее)

MEASURES

Условие MEASURES определяет, какие колонки из sql запроса будут использоваться для доступа к данным. Количество столбцов может быть 1 или больше.

```
select empno, ename, sal from scott.emp t
model
dimension by (empno)
measures (ename, sal)
rules ()
order by empno;
```

EMPNO	ENAME	SAL
7369	SMITH	800
7499	ALLEN	1600
7521	WARD	1250
7566	JONES	2975
7654	MARTIN	1250
7698	BLAKE	2850
7782	CLARK	2450
7788	SCOTT	3000
7839	KING	5000
7844	TURNER	1500
7876	ADAMS	1100
7900	JAMES	950
7902	FORD	3000
7934	MILLER	1300

Можно определять константы, выражения

```
select empno, ename, sal, date_now
from scott.emp t
model
dimension by (empno)
measures (ename, sal * 100 as sal, sysdate as
date_now)
rules ()
order by empno;
```

EMPNO	ENAME	SAL	DATE_NOW
-----	-----	-----	-----

7369	SMITH	80000	06.04.2010
7499	ALLEN	160000	06.04.2010
7521	WARD	125000	06.04.2010
7566	JONES	297500	06.04.2010
7654	MARTIN	125000	06.04.2010
7698	BLAKE	285000	06.04.2010
7782	CLARK	245000	06.04.2010
7788	SCOTT	300000	06.04.2010
7839	KING	500000	06.04.2010
7844	TURNER	150000	06.04.2010
7876	ADAMS	110000	06.04.2010
7900	JAMES	95000	06.04.2010
7902	FORD	300000	06.04.2010
7934	MILLER	130000	06.04.2010

DIMENSION и MEASURES

Нельзя обращаться к столбцу, если он отсутствует в dimension by и measures

```
select empno, ename, sal from scott.emp t
model
dimension by (empno)
measures (ename)
rules ()
order by empno;
```

ORA-32614: недопустимое выражение MODEL SELECT

Нельзя использовать колонку в двух условиях сразу

```
select empno, ename from scott.emp t
model
dimension by (empno)
measures (empno, ename)
rules ()
order by empno;
```

ORA-00957: дублируется имя столбца

Константа NULL не разрешена

```
select empno, ename from scott.emp t model
dimension by (empno, null as test)
measures (ename)
rules ()
order by empno;
```

ORA-01723: столбец нулевой длины недопустим

```
select empno, ename from scott.emp t
model
dimension by (empno)
measures (ename, '' as test)
rules ()
order by empno;
```

ORA-01723: столбец нулевой длины недопустим

Вместо NULL следует использовать CAST

```
select empno, ename, test from scott.emp t
```

```

        model
        dimension by (empno)
        measures (ename, CAST( NULL AS VARCHAR2(10) )
as test)
        rules ()
        order by empno;

```

EMPNO	ENAME	TEST
7369	SMITH	
7499	ALLEN	
7521	WARD	
7566	JONES	
7654	MARTIN	
7698	BLAKE	
7782	CLARK	
7788	SCOTT	
7839	KING	
7844	TURNER	
7876	ADAMS	
7900	JAMES	
7902	FORD	
7934	MILLER	

Если будет превышена допустимая длина поля, получим ошибку

```

select empno, ename, test from scott.emp t
        model
        dimension by (empno)
        measures (ename, CAST( NULL AS VARCHAR2(10) )
as test)
        rules (test[any] = '12345678910')
        order by empno;

```

ORA-25137: Нарушен диапазон значений данных

RULES

Условие rules описывает расчёт значений для блока measures.

Основные правила

Условие rules может быть пустым

```

select empno, job, ename from scott.emp t
        model
        dimension by (empno, job)
        measures (ename)
        rules() order by empno;

```

EMPNO	JOB	ENAME
7369	CLERK	SMITH
7499	SALESMAN	ALLEN
7521	SALESMAN	WARD
7566	MANAGER	JONES
7654	SALESMAN	MARTIN
7698	MANAGER	BLAKE


```

7782 MANAGER CLARK
7788 ANALYST SCOTT
7839 PRESIDENT KING
7844 SALESMAN TURNER
7876 CLERK ADAMS
7900 CLERK JAMES
7902 ANALYST FORD
7934 CLERK MILLER

```

Можно изменять любые значения колонок из measures

```

select empno, job, ename from scott.emp t
model
dimension by (empno)
measures (job, ename)
rules (
  ename[7839] = 'YOU!!!',
  ename[empno < 7788] = 'Эй ты',
  ename[empno BETWEEN 7900 and 7902] = 'O-LO-LO'
) order by empno;

```

EMPNO	JOB	ENAME
7369	CLERK	Эй ты
7499	SALESMAN	Эй ты
7521	SALESMAN	Эй ты
7566	MANAGER	Эй ты
7654	SALESMAN	Эй ты
7698	MANAGER	Эй ты
7782	MANAGER	Эй ты
7788	ANALYST	SCOTT
7839	PRESIDENT	YOU!!!
7844	SALESMAN	TURNER
7876	CLERK	ADAMS
7900	CLERK	O-LO-LO
7902	ANALYST	O-LO-LO
7934	CLERK	MILLER

Условие `ename[7839]` называется «cell reference» и определяет значение колонки, для которой значение `empno` (ключ из `dimension by`) равно 7839. Cell reference может указывать только на колонку measures. Когда cell reference используется для присвоения некоего значения, на левой стороне условия (относительно знака `=`), то используется название «cell assignment». Для правой части условия cell reference не имеет ни каких спец. названий. Часть условия заключенная в квадратные скобки (`7839, empno < 7788`) называется «dimension reference». Dimension references для всех cell references может содержать как константы, так и различные условия. Для cell assignment можно использовать ключевое слово `ANY` — любое значение.

```

select empno, job, ename from scott.emp t
model
dimension by (empno)
measures (job, ename)
rules (
  ename[any] = 'QWERTY'
) order by empno;

```

EMPNO	JOB	ENAME
7369	CLERK	QWERTY
7499	SALESMAN	QWERTY
7521	SALESMAN	QWERTY
7566	MANAGER	QWERTY
7654	SALESMAN	QWERTY
7698	MANAGER	QWERTY
7782	MANAGER	QWERTY
7788	ANALYST	QWERTY
7839	PRESIDENT	QWERTY
7844	SALESMAN	QWERTY
7876	CLERK	QWERTY
7900	CLERK	QWERTY
7902	ANALYST	QWERTY
7934	CLERK	QWERTY

Можно использовать вложенные cell references, но только для определения одной ячейки

```
select empno, job, ename from scott.emp t
model
dimension by(empno)
measures(job, ename, empno as empno2)
rules(
ename[ (empno2[7900] + 2) ] = null
)
order by empno;
```

EMPNO	JOB	ENAME
7369	CLERK	SMITH
7499	SALESMAN	ALLEN
7521	SALESMAN	WARD
7566	MANAGER	JONES
7654	SALESMAN	MARTIN
7698	MANAGER	BLAKE
7782	MANAGER	CLARK
7788	ANALYST	SCOTT
7839	PRESIDENT	KING
7844	SALESMAN	TURNER
7876	CLERK	ADAMS
7900	CLERK	JAMES
7902	ANALYST	
7934	CLERK	MILLER

Если значение dimension is null, то оно будет попадать под следующие условия • [any] • [group_1 is any] • [null] • [group_1 is null] Условие типа «[group_1 = null]» не работает

```
select
id ,
using_any ,
using_is_any ,
using_null ,
using_is_null ,
using_equals_null from
(select 'A' id from dual union all
```

```

        select null from dual) tmodel
dimension by( id )measures
(
    'N' as using_any ,
    'N' as using_is_any ,
    'N' as using_null ,
    'N' as using_is_null ,
    'N' as using_equals_null
)rules
(
    using_any          [ ANY          ] = 'Y' ,
    using_is_any       [ id IS ANY    ] = 'Y' ,
    using_null         [ NULL         ] = 'Y' ,
    using_is_null      [ id IS NULL   ] = 'Y' ,
    using_equals_null [ id = NULL    ] = 'Y'
)
;

ID USING_ANY  USING_IS_ANY  USING_NULL  USING_IS_NULL  USING_EQUALS_NULL
--
A   Y         Y         N         N         N
    Y         Y         Y         Y         N

```

Если попробовать изменить dimension столбец, то получим ошибку

```

select empno, job, ename
  from scott.emp t model
  dimension by(empno)
  measures(job, ename)
  rules(
    empno[7369] = null
  )
 order by empno;

```

ORA-00904: : недопустимый идентификатор

Если указано неверное количество значений для dimension references, то получим ошибку

```

select empno, job, ename
  from scott.emp t model
  dimension by(empno, job)
  measures(ename)
  rules(
    ename[7499] = 'QWERTY'
  )
 order by empno;

```

ORA-00947: не хватает значений для данных

Использовать dimension столбец в правой части условия для dimension references нельзя

```

select empno, job, ename
  from scott.emp t

```

```

model
dimension by(empno, job)
measures(ename)
rules(
    ename[7499,length(job) < empno ] = 'QWERTY'
)
order by empno;

```

ORA-32625: недопустимое измерение в предикате ссылки на ячейку

Если вложенные cell references определяют более чем одну ячейку получим ошибку

```

select empno, job, ename
from scott.emp t
model
dimension by(empno)
measures(job, ename)
rules(
    ename[ (job[any]) ] = null
)
order by empno;

```

ORA-32622: недопустимая ссылка на несколько ячеек

FOR Loops

Для dimension references возможно использование цикла.

Запрос без использования цикла.

```

select deptno, dname, loc from scott.dept t
model
dimension by(deptno)
measures(dname, loc)
rules(
    loc[20] = 'SPB',
    loc[30] = 'SPB',
    loc[40] = 'SPB'
) order by deptno;

```

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	SPB
30	SALES	SPB
40	OPERATIONS	SPB

С циклом

```

select deptno, dname, loc from scott.dept t
model
dimension by(deptno)
measures(dname, loc)
rules(

```

```

loc[FOR deptno FROM 20 TO 40 INCREMENT 10] =
'SPB'
)order by deptno;

```

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	SPB
30	SALES	SPB
40	OPERATIONS	SPB

ИЛИ

```

select deptno, dname, locfrom scott.dept t
model
dimension by(deptno)
measures(dname, loc)
rules(
loc[FOR deptno in (20,30,40) ] = 'SPB'
)order by deptno;

```

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	SPB
30	SALES	SPB
40	OPERATIONS	SPB

Можно использовать комбинирование строки со значением из цикла

```

with t as
(select 'W-'||rownum as id, ename, jobfrom
scott.emp)
select * from t
model
dimension by(id)
measures(ename, job)
rules(
job[FOR id LIKE 'W-%' FROM 2 TO 14 INCREMENT 2 ]
= null
)

```

ID	ENAME	JOB
W-1	SMITH	CLERK
W-2	ALLEN	
W-3	WARD	SALESMAN
W-4	JONES	
W-5	MARTIN	SALESMAN
W-6	BLAKE	
W-7	CLARK	MANAGER
W-8	SCOTT	
W-9	KING	PRESIDENT
W-10	TURNER	
W-11	ADAMS	CLERK

W-12	JAMES	
W-13	FORD	ANALYST
W-14	MILLER	

UNIQUE SINGLE REFERENCE

Данное свойство позволяет иметь дубликаты ключевых значений в запросе.

```
select * from scott.emp t
  model
  dimension by (job)
  measures (ename, sal)
  rules ()
  order by job
;
```

ORA-32638: Неуникальная адресация в измерениях MODEL

Указываем **UNIQUE SINGLE REFERENCE**

```
select * from scott.emp t
  model UNIQUE SINGLE REFERENCE
  dimension by (job)
  measures (ename, sal)
  rules ()
  order by job;
```

JOB	ENAME	SAL
ANALYST	SCOTT	3000
ANALYST	FORD	3000
CLERK	MILLER	1300
CLERK	JAMES	950
CLERK	SMITH	800
CLERK	ADAMS	1100
MANAGER	BLAKE	2850
MANAGER	JONES	2975
MANAGER	CLARK	2450
PRESIDENT	KING	5000
SALESMAN	TURNER	1500
SALESMAN	MARTIN	1250
SALESMAN	WARD	1250
SALESMAN	ALLEN	1600

Можно задавать правила для все строк

```
select * from scott.emp t
  model UNIQUE SINGLE REFERENCE
  dimension by (job)
  measures (ename, sal)
  rules (sal[any]=9999)
  order by job;
```

JOB	ENAME	SAL
ANALYST	SCOTT	9999
ANALYST	FORD	9999
CLERK	MILLER	9999
CLERK	JAMES	9999

CLERK	SMITH	9999
CLERK	ADAMS	9999
MANAGER	BLAKE	9999
MANAGER	JONES	9999
MANAGER	CLARK	9999
PRESIDENT	KING	9999
SALESMAN	TURNER	9999
SALESMAN	MARTIN	9999
SALESMAN	WARD	9999
SALESMAN	ALLEN	9999

Можно ограничивать левое условие rules

```
select * from scott.emp t
      model UNIQUE SINGLE REFERENCE
      dimension by (job)
      measures (ename, sal)
      rules (sal['CLERK']=9999)
      order by job;
```

JOB	ENAME	SAL
ANALYST	SCOTT	3000
ANALYST	FORD	3000
CLERK	ADAMS	9999
CLERK	MILLER	9999
CLERK	SMITH	9999
CLERK	JAMES	9999
MANAGER	JONES	2975
MANAGER	BLAKE	2850
MANAGER	CLARK	2450
PRESIDENT	KING	5000
SALESMAN	ALLEN	1600
SALESMAN	MARTIN	1250
SALESMAN	WARD	1250
SALESMAN	TURNER	1500

Но нельзя указывать в правой части условия rules ссылку на неуникальную ячейку

```
select * from scott.emp t
      model UNIQUE SINGLE REFERENCE
      dimension by (job)
      measures (ename, sal)
      rules (sal[any]= sal['MANAGER'])
      order by job;
```

ORA-32638: Неуникальная адресация в измерениях MODEL

Функция CV()

Если необходимо получить значение ячейки из dimension, то использовать cell references нельзя. Но можно использовать функцию CV() , которая дает доступ к текущему значению dimension.

```
select * from scott.emp t
      model
      dimension by (empno)
      measures (job,ename, 0 sub_empno)
      rules (
```

```
sub_empno[any] = cv(empno) * 10
) order by empno;
```

EMPNO	JOB	ENAME	SUB_EMPNO
7369	CLERK	SMITH	73690
7499	SALESMAN	ALLEN	74990
7521	SALESMAN	WARD	75210
7566	MANAGER	JONES	75660
7654	SALESMAN	MARTIN	76540
7698	MANAGER	BLAKE	76980
7782	MANAGER	CLARK	77820
7788	ANALYST	SCOTT	77880
7839	PRESIDENT	KING	78390
7844	SALESMAN	TURNER	78440
7876	CLERK	ADAMS	78760
7900	CLERK	JAMES	79000
7902	ANALYST	FORD	79020
7934	CLERK	MILLER	79340

Можно использовать функцию CV() без аргумента для возврата текущего значения dimension относительно позиции в cell reference.

```
select * from scott.emp t
model
dimension by (empno, job)
measures (ename, empno as sub_empno)
rules (
sub_empno[any,any] = sub_empno[cv(),cv()]] * 10
)
order by empno;
```

EMPNO	JOB	ENAME	SUB_EMPNO
7369	CLERK	SMITH	73690
7499	SALESMAN	ALLEN	74990
7521	SALESMAN	WARD	75210
7566	MANAGER	JONES	75660
7654	SALESMAN	MARTIN	76540
7698	MANAGER	BLAKE	76980
7782	MANAGER	CLARK	77820
7788	ANALYST	SCOTT	77880
7839	PRESIDENT	KING	78390
7844	SALESMAN	TURNER	78440
7876	CLERK	ADAMS	78760
7900	CLERK	JAMES	79000
7902	ANALYST	FORD	79020
7934	CLERK	MILLER	79340

Для функции CV() нельзя использовать значение measures

```
select * from scott.emp t
model
dimension by (empno)
measures (job,ename, empno as sub_empno)
rules (
sub_empno[any] = cv(sub_empno) * 10
)
```



```
order by empno;
```

ORA-00904: : недопустимый идентификатор

Нельзя использовать вне dimension references функцию CV() без аргумента

```
select * from scott.emp t
model
dimension by (empno)
measures (job,ename, 0 sub_empno)
rules (
sub_empno[any] = cv() * 10
)
order by empno;
```

ORA-32611: некорректное использование оператора MODEL CV

PARTITION BY

Позволяет ускорить обработку данных на многопроцессорных системах и в некоторых случаях сокращает код в dimension references.

```
select * from scott.emp t
model
partition by (deptno)
dimension by (job,ename)
measures (sal, sal as new_sal)
rules (
new_sal['CLERK' , any] = sal[cv(), cv()] * 2,
new_sal['MANAGER' , any] = sal[cv(), cv()] * 3,
new_sal['SALESMAN', any] = sal[cv(), cv()] * 4
)
order by deptno,job,ename;
```

DEPTNO	JOB	ENAME	SAL	NEW_SAL
10	CLERK	MILLER	1300	2600
10	MANAGER	CLARK	2450	7350
10	PRESIDENT	KING	5000	5000
20	ANALYST	FORD	3000	3000
20	ANALYST	SCOTT	3000	3000
20	CLERK	ADAMS	1100	2200
20	CLERK	SMITH	800	1600
20	MANAGER	JONES	2975	8925
30	CLERK	JAMES	950	1900
30	MANAGER	BLAKE	2850	8550
30	SALESMAN	ALLEN	1600	6400
30	SALESMAN	MARTIN	1250	5000
30	SALESMAN	TURNER	1500	6000
30	SALESMAN	WARD	1250	5000

Агрегатные и аналитические функции

При использовании условия model нельзя определять агрегатные или аналитические функции в SELECT условии.

```
select empno, max(job), max(ename), max(sal)
from scott.emp t
group by empno
model
dimension by (empno)
measures (job, ename, sal)
rules ()
order by job;
```

ORA-00979: выражение не является выражением GROUP BY

Но, можно использовать в PARTITION BY, DIMENSION BY, MEASURES или RULES условиях.

```
select empno, job, sal from scott.emp t
group by empno
model
dimension by (empno, max(job) as job)
measures (sum(sal) as sal)
rules ( sal[any,'PRESIDENT'] =
sal[cv(),'PRESIDENT']/2 )
order by job;
```

EMPNO	JOB	SAL
7788	ANALYST	3000
7902	ANALYST	3000
7876	CLERK	1100
7369	CLERK	800
7900	CLERK	950
7934	CLERK	1300
7698	MANAGER	2850
7566	MANAGER	2975
7782	MANAGER	2450
7839	PRESIDENT	2500
7654	SALESMAN	1250
7521	SALESMAN	1250
7499	SALESMAN	1600
7844	SALESMAN	1500

Агрегатные и аналитические функции можно использовать в правой части условия rules, в левой части использовать нельзя.

```
select empno, job, sal from scott.emp t
model
dimension by (empno, job)
measures (sal)
rules ( sal[any,'PRESIDENT'] =
sum(sal)[any,'CLERK'] )
order by job;
```

EMPNO	JOB	SAL
7788	ANALYST	3000

7902	ANALYST	3000
7876	CLERK	1100
7934	CLERK	1300
7900	CLERK	950
7369	CLERK	800
7698	MANAGER	2850
7782	MANAGER	2450
7566	MANAGER	2975
7839	PRESIDENT	4150
7521	SALESMAN	1250
7499	SALESMAN	1600
7654	SALESMAN	1250
7844	SALESMAN	1500

Порядок выполнения работы

1. Составьте запрос, который возвращает условное изображение шахматной доски. В заголовках столбцов должны быть латинские буквы a,b,c,d,e,f,h,g. Каждая строка должна начинаться с цифры 1,2,3,4,5,6,7,8. Клетки белого цвета помечаются 0, клетки черного цвета – 1. Для формирования запроса использовать оператор MODEL.
2. Составьте запрос, который возвращает список домов из таблицы TB_ELEKTROSTAL_2018. Считается, что к одному дому относятся все абоненты, у которых совпадает адрес, за исключением номера квартиры. Список домов сгруппировать по улицам. Для каждого дома указать количество квартир, Для каждой улицы добавить итоговую строку с указанием количества домов на улице. Также для каждой улицы добавить строку с указанием количества многоэтажных домов. Многоэтажным домом будем считать дом, в котором количество квартир превышает число 90. Указание: в запросе использовать оператор MODEL.
3. Составьте запрос к таблице EMPLOYEES базы данных HR. Запрос должен выводить список сотрудников упорядоченных по должностям. В результат включить три колонки – должность, имя и фамилию сотрудника – одним выражением и сумму его заработной платы. Для каждой должности включить

дополнительную строку с указанием суммарной заработной платы, работников, работающих в этой должности.

Содержание отчета.

1. Название работы
2. Цель работы
3. Листинги запросов
4. Скриншоты результатов выполнения запросов.
5. Выводы

Контрольные вопросы

1. Объяснить назначение оператора MODEL.
2. Объяснить особенности использования раздела DIMENSION BY в операторе MODEL.
3. Объяснить особенности использования раздела MEASURES в операторе MODEL.
4. Объяснить особенности использования раздела PARTITION в операторе MODEL.
5. Объяснить особенности использования раздела RULES в операторе MODEL.
6. Охарактеризовать структуру оператора MODEL.
7. Описать назначение функции CV() в операторе MODEL.
8. Описать особенности использование циклов в разделе RULES оператора MODEL.