**CASE**

Наверное если вы знакомы с програмированием вам будет знакомым этот условный опратор. Так вот это функция выполняют эти же функции в SQL. Поясним синтаксис функции CASE:

**CASE expr WHEN comparison\_expr\_1 then return\_expr\_1**

**WHEN comparison\_expr\_2 then return\_expr\_2**

**……………….**

**WHEN comparison\_expr\_n then return\_expr\_n**

**ELSE else\_expr**

**END**

где

expr – это то выражение, которое сравнивается с comparison\_expr\_1 если оно совпадает, то выходит результат return\_expr\_1. Иначе идет вторая проверка и так до n-ой проверки. Если выражение не удовлетворило ни одну проверку comparison\_expr\_n, то в результате выйдет else\_expr.

К примеру с таблицы EMP вывидим список сотрудников, при этом начисляя им бонусы в зависемоси от их рода работы.

SELECT job, sal,

CASE job WHEN 'CLERK' THEN sal\*0.10

WHEN 'MANAGER' THEN sal\*0.12

WHEN 'ANALYST' THEN sal\*0.15

ELSE 0

END BONUS

FROM EMP

|  |  |  |
| --- | --- | --- |
| JOB | SAL | BONUS |
| CLERK | 800,00 | 80,00 |
| SALESMAN | 1 600,00 | 0,00 |
| SALESMAN | 1 250,00 | 0,00 |
| MANAGER | 2 975,00 | 357,00 |
| SALESMAN | 1 250,00 | 0,00 |
| MANAGER | 2 850,00 | 342,00 |
| ……… |  |  |

**DECODE**

Функция DECODE выполняет ту же функцию, что и CASE единственным отличием является ее синтаксис. Надо отметить, что исторически функция DECODE в SQL намного старьше, но ORACLE в виду более легкой читаемости CASE, ввел также данную конструкцию в свой SQL.

**DECODE (expression, search\_1, result\_1,**

**search\_2, result\_2,**

**……….**

**search\_n, result\_n,**

**default)**

где

expression – это то выражение, которое сравнивается с search1, если оно совпадает, то выходит результат result1. Иначе идет вторая проверка и так далее до n-ой проверки. В конце в случаи всех неудачных проверок в результате выйдет default.

И перепишем верхний пример CASE, но при этом уже используя DECODE. Как ниже из примера мы видим, что никакой разницы в исполнение между CASE и DECODE нет.

SELECT job, sal,

CASE job WHEN 'CLERK' THEN sal\*0.10

WHEN 'MANAGER' THEN sal\*0.12

WHEN 'ANALYST' THEN sal\*0.15

ELSE 0

END BONUS,

DECODE(job,'CLERK', sal\*0.10,

‘MANAGER',sal\*0.12,

'ANALYST',sal\*0.15,

0) BONUS\_2

FROM EMP

|  |  |  |  |
| --- | --- | --- | --- |
| JOB | SAL | BONUS | BONUS\_2 |
| CLERK | 800,00 | 80,00 | 80,00 |
| SALESMAN | 1 600,00 | 0,00 | 0,00 |
| SALESMAN | 1 250,00 | 0,00 | 0,00 |
| MANAGER | 2 975,00 | 357,00 | 357,00 |
| SALESMAN | 1 250,00 | 0,00 | 0,00 |
| MANAGER | 2 850,00 | 342,00 | 342,00 |

**Источник:**

[**http://sql.az/index.php?option=com\_content&view=article&id=75&lang=ru**](http://sql.az/index.php?option=com_content&view=article&id=75&lang=ru)

**Применение функции DECODE**

***Функция DECODE для проверки условий в запросах Oracle SQL***

**Задание:**

Напишите запрос, который возвращает информацию об имени, фамилии и должности сотрудников (столбец JOB\_ID) на основе таблицы hr.employees. При этом:

* если в столбце JOB\_ID для сотрудников находится значение SA\_REP, то должно выводиться "Торговый представитель";
* если в столбце JOB\_ID для сотрудников находится значение SA\_MAN, то должно выводиться "Менеджер по продажам";
* если в этом столбце находится любое другое значение, то должно выводиться "Другое".

Результат запроса должен быть таким, как представлено на рис. 3.6-1

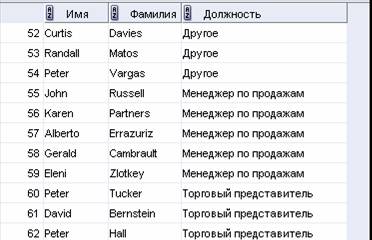


Рис. 3.6-1

**Решение:**

Код соответствующего запроса может быть таким:

SELECT first\_name AS "Имя", last\_name As "Фамилия", DECODE(JOB\_ID, 'SA\_REP', 'Торговый представитель', 'SA\_MAN', 'Менеджер по продажам', 'Другое' ) AS "Должность" FROM hr.employees