POSTGRESQL - EXPRESSIONS

An expression is a combination of one or more values, operators, and PostgresSQL functions that evaluate to a value.

PostgreSQL EXPRESSIONs are like formulas and they are written in query language. You can also use to query the database for specific set of data.

Syntax:

Consider the basic syntax of the SELECT statement as follows:

```
SELECT column1, column2, columnN
FROM table_name
WHERE [CONTION | EXPRESSION];
```

There are different types of PostgreSQL expressions, which are mentioned below:

PostgreSQL - Boolean Expressions:

PostgreSQL Boolean Expressions fetch the data on the basis of matching single value. Following is the syntax:

```
SELECT column1, column2, columnN
FROM table_name
WHERE SINGLE VALUE MATCHTING EXPRESSION;
```

Consider the table COMPANY having records as follows:

```
testdb# select * from COMPANY;
 id | name  | age | address   | salary
----+-------+-----+-----------+--------
  1 | Paul  |  32 | California|  20000
  2 | Allen |  25 | Texas     |  15000
  3 | Teddy |  23 | Norway    |  20000
  4 | Mark  |  25 | Rich-Mond |  65000
  5 | David |  27 | Texas     |  85000
  6 | Kim   |  22 | South-Hall|  45000
  7 | James |  24 | Houston   |  10000
(7 rows)
```

Here is the simple example showing usage of PostgreSQL Boolean Expressions:

```
testdb=# SELECT * FROM COMPANY WHERE SALARY = 10000;
```

Above PostgreSQL statement will produce the following result:

```
 id | name  | age | address  | salary
----+-------+-----+----------+--------
  7 | James |  24 | Houston  |  10000
(1 row)
```

PostgreSQL - Numeric Expression:

These expressions are used to perform any mathematical operation in any query. Following is the syntax:

```
SELECT numerical_expression as  OPERATION_NAME
[FROM table_name WHERE CONDITION] ;
```

Here numerical_expression is used for mathematical expression or any formula. Following is a simple examples showing usage of SQL Numeric Expressions:

```
testdb=# SELECT (15 + 6) AS ADDITION ;
```

Above PostgreSQL statement will produce the following result:

```
 addition
----------
       21
(1 row)
```

There are several built-in functions like avg(), sum(), count(), etc., to perform what is known as aggregate data calculations against a table or a specific table column.

```
testdb=# SELECT COUNT(*) AS "RECORDS" FROM COMPANY;
```

Above PostgreSQL statement will produce the following result:

```
 RECORDS
---------
       7
(1 row)
```

PostgreSQL - Date Expressions:

Date Expressions return current system date and time values and these expressions will be used in various data manipulations.

```
testdb=#  SELECT CURRENT_TIMESTAMP;
```

Above PostgreSQL statement will produce the following result:

```
              now
-------------------------------
 2013-05-06 14:38:28.078+05:30
(1 row)
```