**DATA MANIPULATING LANGUAGE**

Data Manipulating Language: This is the 2nd sub language in SQL,

which is used to manipulate the data within database. This Language contains 4 commands.

1. Insert

2. Update

3. Delete

4. Select

**1. INSERT:**

* Using this command we can Insert the records into the existing table
* We can insert the records into the table in two methods

Explicit method

Implicit method

**Explicit method:**

* In this method user has to enter all the values into all the columns without

anything omitting (or) left any column data

* Syntax: INSERT INTO <TABLE NAME> VALUES <VAL1, VAL2,

….VALN>;

* Ex: INSERT INTO EMP VALUES (101,’RAJ’,9500);

1 Row(s) affected

**Implicit method:**

* In this method we can enter the values into the required columns in the table,

so that user can omit (or) left some columns data while he enters the records into the table

NOTE :- If the user omit any column data in the table then it automatically takes NULL

* Syntax: INSERT INTO <TABLE NAME> (COL1, COL2….COLN)

VALUES (VAL1, VAL2,… VALN);

* Ex: INSERT INTO EMP (EID, SAL) VALUES (106,9999);

**2. UPDATE:**

* This command is used to modify the data in the existing table
* By using this command we can modify all the records in the table & also specific records in the table (Using ‘where’ clause)
* Syntax: UPDATE <TABLE NAME> SET COL=VALUE;
* Ex: UPDATE EMP SET SAL=10000;

Syntax change for more than one data simultaneously

* Syntax: UPDATE <TABLE NAME> SET COL1=VALUE,

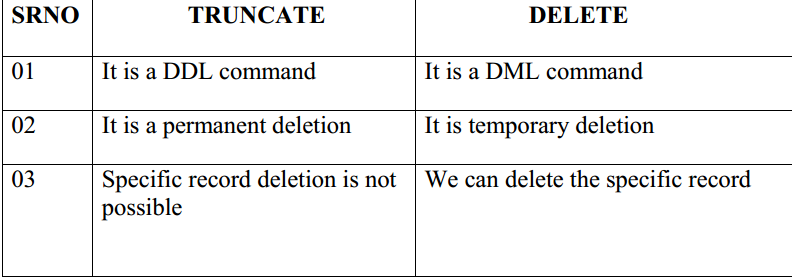
COL2=VALUE………COLN=VALUE;

* Ex: UPDATE EMP SET EID=007,SAL=10000;

**3. DELETE:**

* This command is used to delete the records from existing table
* Using this command we can delete all the records and also to delete specific record (by using ‘where’ clause)
* Syntax: DELETE FROM <TABLE NAME>
* Ex: DELETE FROM EMP;

10 row(s) affected



**4. SELECT:**

* This command is used to retrieve the data from existing table.
* Using this command we can retrieve all the records & also specific records from existing table (by using ‘where’ clause)
* Using this command we can retrieve the data from the table in 3 ways

1. Projection

2. Selection

3. Joins

* Syntax: SELECT \* FROM <TABLE NAME>
* Ex: SELECT \* FROM EMP;

NOTE -: \* represents all columns

**Projection:**

* Retrieving the data from specific columns is known as Projection
* Syntax: SELECT COL1,COL2……..COLN FROM <TABLE NAME>
* Ex: SELECT EID,ENAME FROM EMP;

**Selection:**

* Retrieving the data based on some condition is called selection
* In SQL, whenever we need to check a condition, we need to use a special clause called ‘where’
* Syntax: SELECT \* FROM <TABLENAME> WHERE (CONDITION);
* Ex: SELECT \* FROM EMP WHERE EID=101;

**WHERE CLAUSE:**

* This clause is used to check the condition based on the Condition, we can retrieve, update, delete specific records in the table
* So we can apply the where clause only in select, update & delete

**Select Command With Where clause:**

* Syntax: SELECT \* FROM <TABLE NAME> WHERE <CONDITION>
* Ex: SELECT \* FROM EMP WHERE EID=102;

**Update Command With Where clause:**

* Syntax: UPDATE <TABLE NAME> SET <COLUMN NAME>=VALUE

WHERE (CONDITION);

* Ex: UPDATE EMP SET ENAME=”sai” WHERE EID=102;

**Delete Command With Where clause:**

* Syntax: DELETE FROM <TABLE NAME>WHERE <CONDITION>
* Ex: DELETE FROM EMP WHERE EID=102;

**ALIAS:**

* ALIAS is a duplicate name (or) alternate name for the original column name (or) Table name (or) an expression name.
* Column level Alias:
* Syntax: SELECT COLUMN NAME AS “ALIAS NAME”,

COLUMN NAME AS “ALIAS NAME”,

:

:

COLUMN NAME AS “ALIAS NAME” FROM <TABLE NAME>;

* EX: SELECT EID AS “EMPLOYEE ID”, ENAME AS “EMPLOYEE

NAME”, SAL AS “SALARY” FROM EMP;

* NOTE: In the above example the keyword ‘as’ is optional
* EX: SELECT EID “EMPLOYEE ID”, ENAME “EMPLOYEE NAME”,

SAL “SALARY” FROM EMP;

* NOTE: In the above example quotations is also optional but there should not be space between column name
* EX: SELECT EID EMPLOYEEID, ENAME EMPLOYEENAME, SAL

SALARY FROM EMP;

* Ex: SELECT EID EMPLOYEEID, ENAME EMPLOYEENAME, SAL

SALARY, SAL\*12 ANNUALSALARY FROM EMP;

* EX: SELECT EID EMPLOYEEID, ENAME EMPLOYEENAME, SAL

SALARY FROM EMP WHERE ANNUALSALARY > 115000

* In the above example returns the runtime error message invalid column name ‘annual salary’ because we cannot check the conditions on Alias name

**IDENTITY:**

It is use to generate unique values in sequential order without

user interaction. The default value of identity is Identity (1, 1).

* Syntax: Identity (seed, increment)
* Ex: CREATE TABLE EMP (EID INT IDENTITY (100, 1), ENAME VARCHAR (50));