**DDL (Data Definition Language)**

Used to CREATE, ALTER, and DROP the descriptions of the database tables (relations).

**CREATE, ALTER and DROP**

table…………………………………….……relation

row……………………………………..…….tuple

column………………………………….……attribute

**DATA TYPES**

* Numeric: NUMBER, NUMBER(s,p), INTEGER, INT, FLOAT, DECIMAL
* Character: CHAR(n), VARCHAR(n), VARCHAR2(n), CHAR VARYING(n)
* Bit String: BLOB, CLOB
* Boolean: true, false, and null
* Date and Time: DATE (YYYY-MM-DD) TIME( HH:MM:SS)
* Timestamp: DATE + TIME
* USER Defined types

**CREATE TABLE**

* Specifies a new base relation by giving it a name, and specifying each of its attributes and their data types

Syntax of CREATE Command:

**CREATE TABLE <***table name>*( <Attribute*A*1> <Data Type*D*1> [< Constarints>],<Attribute *A*2> <Data Type *D*2> [< Constarints>],

<Attribute *A*n> <Data Type *D*n> [< Constarints>],

[<integrity-constraint1>, <integrity-constraint k> ] );

A constraint NOT NULL may be specified on an attribute, a constraint NOT NULL may be specified on an attribute.

* + Ex: CREATE TABLE DEPARTMENT (

DNAME VARCHAR (10) NOT NULL,

DNUMBER INTEGER NOT NULL,

MGRSSN CHAR (9), MGRSTARTDATE CHAR (9));

**DROP TABLE**

* Used to remove a relation (base table) and its definition.
* The relation can no longer be used in queries, updates, or any other commands since its description no longer exists

**Example:** DROP TABLE DEPENDENT;

**ALTER TABLE:**

* Used to add an attribute to/from one of the base relations drop constraint -- The new attribute will have NULLs in all the tuples of the relation right after the command is executed; hence, the NOT NULL constraint is *not allowed* for such an attribute.

**Example:** ALTER TABLE EMPLOYEE ADD JOB VARCHAR2 (12);

* The database users must still enter a value for the new attribute JOB for each EMPLOYEE tuple. This can be done using the UPDATE command.

**DML (Data Manipulation Language) Commands**

Select command, Insert Command, Update Command.

**Syntax of Select Command:**

SELECT <attribute list> FROM <table list> WHERE <condition>

* <attribute list> is a list of attribute names whose values are to be retrieved by the query
* <table list > is a list of the relation names required to process the query
* <condition> is a conditional (Boolean) expression that identifies the tuples to be retrieved by the query

Example: Select \* from Department; (All attributes or columns from table)

Select Dnumber, Dname from Department;

Select Dnumber from Department where Dname=”Research”;

**Syntax of Insert command**

**INSERT** INTO TABLE\_NAME (column1, column2, column3,...columnN) VALUES (value1, value2, value3,...valueN);

Here, column1, column2, column3...column are the names of the columns in the table into which you want to **insert** the data.

(or)

INSERT INTO table\_name  
VALUES (value1,value2,value3…);

(or)

Insert into tablename values (&column1, &column2, &column3,……);

here & is called as insertion operator.

Ex: Insert into department values (‘&dname’, &dnumber, ‘&mgrssn’, ‘&dmgrstartdate’);

**Update command**

The UPDATE statement is used to modify the existing records in a table.

UPDATE table\_name  
SET column1=value1,column2=value2,…  
WHERE condition;

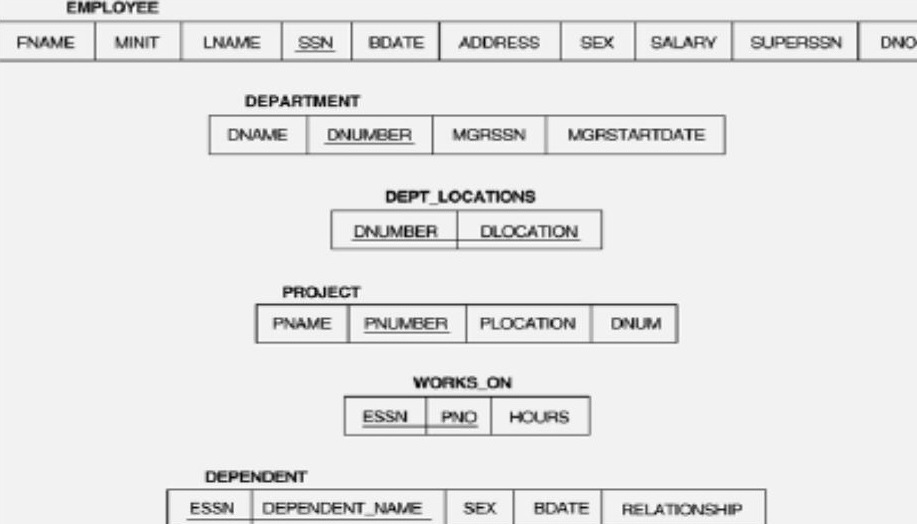
Ex: update department

set dname=’Research’ where dnumber=101;

**Consider the following Employee Database Schema which contains tables like Employee, Department, Dept\_locations, Works\_on, and Dependent.**

**Create database schema for the above mentioned tables.**

**Insert at least 5 records into each table and perform the following queries on these tables.**



1. **Retrieve the birth date and address of the employee whose name is 'John B.**

**Smith'.**

1. **Retrieve the name and address of all employees who work for the 'Research'**

**Department.**

1. **For every project located in 'Stafford', list the project number, the controlling department number, and the department manager's last name, address, and birth date.**
2. **For each employee, retrieve the employee's name, and the name of his or her immediate supervisor.**
3. **Retrieve SSN values for all employees.**
4. **Retrieve all the attribute values of EMPLOYEE s who work in department no.5**
5. **Retrieve the name and address of all employees who work for ‘Research’ Department.**