Rows – tupils

Cols – attribute

In SQL(CASE InSenSitIve), you must define bytes occupied by each data member

SQL divided into DDL (data definition language)  
DML (data manipulation language)  
DCL (Data control language- for admin purpose-i.e. select user access)

To create table  
CREATE TABLE name(FIELD1 number(3) , FIELD2 char(10) , FIELD3 char(10) , FIELD4 number(8,2) ); [char() number() date() time() are keywords]  
 [here 2 defines number of decimals of the specified digits(if extra, truncates from right)]

**To show fields and data types:**   
DESC NAME;

**To input a value:**INSERT INTO name VALUES(101, ‘rajni’ , ‘superstar’ , 999999.9 );  
[must be VALUES]

To view full table:  
SELECT \* FROM name;

**To view selected field:**  
SELECT field1 FROM name;

**To view multiple fields:**SELECT field1,field2 FROM name;  
[order given in select is how data is shown]

**AND, OR, relation operators:**SELECT fields FROM name WHERE field2 OPERATOR value;  
[string comparison IS case sensitive ALSO, to compare it is just ‘=’, NOT ‘==’]  
eg.  
 SELECT empno,desg FROM emp WHERE salary > 3000 AND salary < 5000;

**Using BETWEEN:**  
SELECT \* FROM name WHERE field NOT BETWEEN 3000 AND 500000;  
[NOT used to change result]  
[SELECT \* FROM name WHERE field NOT IN(…) AND salary=123 NOT AND field a=5);]  
[3000 and 50000 are counted]

**Using IN:**  
SELECT \* FROM name WHERE field **IN** (‘valnam1’,’valname2’);  
[to check for values inside a given column]

**Using UPDATE:** to change data values  
UPDATE name SET field = value; // for updating whole column  
UPDATE name SET field = value WHERE field = ‘datatypename’;

**Operations:**  
UPDATE name SET field = field \* 0.1 + field;  
UPDATE name SET field1 = 1000, field2 = ‘fg’; **// multiple SET using ‘,’**

**Using LIKE:**  
SELECT field FROM name WHERE field LIKE ‘s%’;  
  
[% represents all characters before/after/between specified characters][case sensitive]  
[‘a%’ - all names starting with ‘a’][‘%a’ – all ending with a][‘%a%’ - name with a anywhere in it]  
[‘\_’ represents 1 character][‘\_ \_a%’ – third character is ‘a’][‘\_ \_ \_ \_’ means any word with 4 characters]

**Using NULL:** if assigning value is not decided… eg salary not decided  
INSERT INTO name VALUES(123,’df’,NULL,34334);  
// Check for null IS must be used  
SELECT \* FROM emp WHERE field IS NULL; **// NULL containing value is displayed**

**To add a column:**  
ALTER ADD name ADD(fieldname type(size));  
[This adds a column to the table Can check with <desc name;> command]

**To change a column Definition:**  
ALTER ADD name MODIFY(name type(size));  
  
**Aggregate functions:**

|  |  |
| --- | --- |
| FUNCTION NAME | FUNCTION |
| SUM() - SELECT SUM(salary) FROM emp; | Adds all values |
| MIN() | Displays minimum value |
| MAX() | Displays maximum value |
| AVERAGE() | Displays average of selected values |
| COUNT() | Displays number of records |

eg. SELECT SUM(salary),MAX(salary) FROM emp WHERE salary > 2000;