**MySQL**

**Commands for Database**

1. To see all the databases available

* show databases;

1. To create a new database

* create database db\_name;

1. To move to any database or to use any database

* use db\_name;

1. To delete any database

* drop db\_name;

**Commands for Tables**

1. To see table in any database

* show tables;

1. To see details of any table

* desc table\_name;

1. To create a table on a database

* create table table\_name(col\_name col\_type options)

example :

create table user(id int(11) primary key, name varchar(100) not null, city varchar(50));

1. To delete table

* drop table table\_name;

1. To rename table name

* alter table old\_table\_name rename to new\_table\_name;

1. To delete data of table

* truncate table table\_name;

1. Insert into table

* insert into table\_name (col1, col3, col2) values (val1, val3, val2)

**OR**

* insert into table\_name values(val1, val2, val3)

Note: order doesn’t matter in first method but in second it does.

1. To add new column in existing table

* alter table table\_name add col\_name type;

1. To see data of table

* select \* from table\_name;

1. Update values of any column

* update table\_name set col = value ; (updates all values of col)
* update table\_name set col = value where col = value;
* update table\_name set col=val, col=val where col=val;

1. Update specific value of any col

* update table\_name set col = new\_val where col = val;

1. Delete any row of table

* delete from table\_name where col = value;

1. get any specific columns

* select col1,col2 from table\_name;
* \* - means all columns

1. AND, OR, NOT

AND – all must be right

OR – any one should be right

* Select \* from tab\_name where col=val and col=val;

1. Get rid of multiple occurrence of values

* select distince(col) from tab\_name;

1. Between

* select \* from tab\_name where col between (val and val);

1. IN

* select \* from table where col in (val1, va2,…., val n);

1. LIMIT

* select \* from table limit x;
* this will give top ‘x’ rows.

1. OFFSET

* select \* from table limit ‘x’ offset ‘y’
* will give top x row by skipping firs y rows

1. ORDER

* asc – ascending (default), desc - descending
* select \* from table order by col; - ascending or alphabetical
* select \* from table order by col desc; - descending / reverse alpha

1. LIKE

% - multiple characters (including one and zero)

\_ - one character

* To select values starting from a
* select \* from tab where col like ’a%’;
* To select values whose third char is b
* select \* from tab where col like ‘\_ \_ b%’;
* To select values whose second last is o
* select \* from tab where col like ‘%o\_’;

1. SUM(), AVG(), COUNT(), MIN(), MAX()

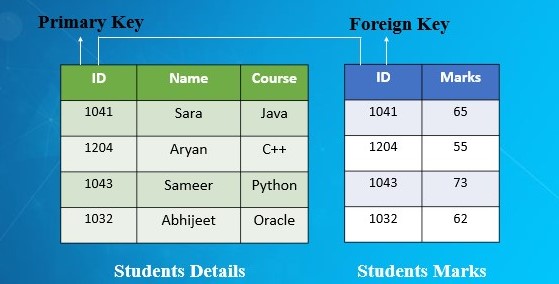
* select sum(col1) from table:

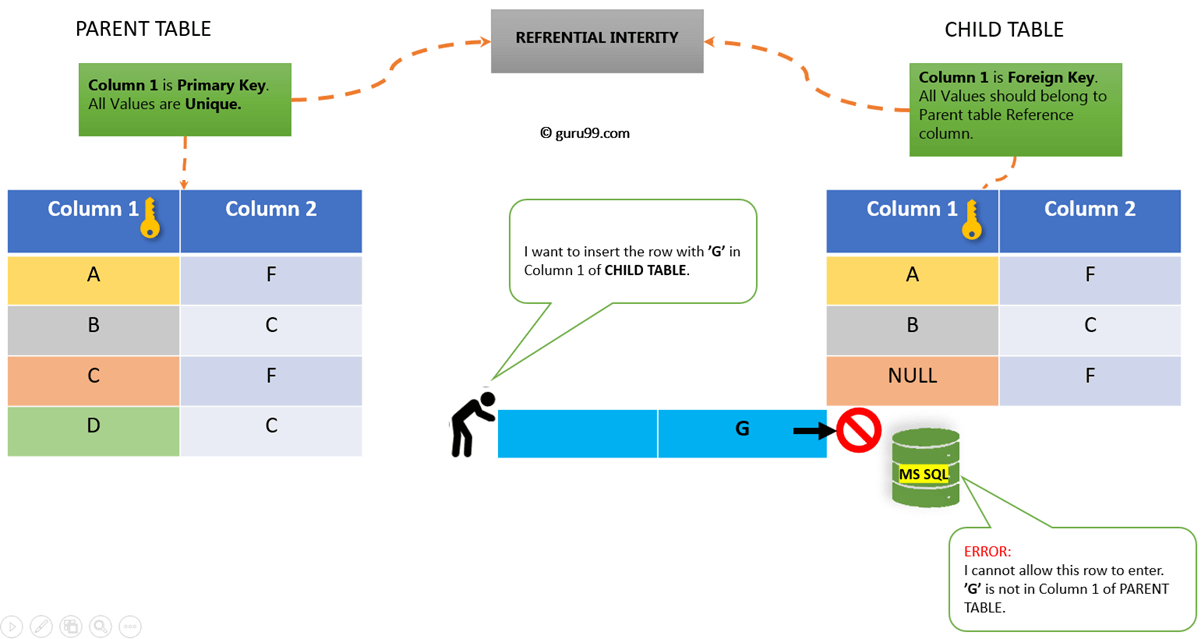
1. Nested query

* To get **name** whose **salary** is max
* select **name** from table where **salary** = (select max(**salary**) from table);

1. Foreign key and joins

Foreign key – help to connect two tables



* make new table tab2 with reference of primary key of tab1
* create table tab1(col1 type, col2 type, col3, foreign key(col3) references tab1(col\_of\_tab1);
* JOIN
* Get value using both the tables
* select tab1.col, tab2.col from tab1, tab2 where tab1.col = tab2.col;
* select tab1.col, tab2.col from tab1 inner join tab2 where tab1.col = tab2.col;