**sql Commands**

Create

*create table table\_name(column1 datatype, column2 datatype, column3 datatype);*

Insert

*Insert into table\_name values(value1, value2, value3);* or   
*Insert into table\_name(column1,column2) values(value1, value2);*

Select

*Select \*from table\_name;* or   
*Select column1,column2,column3 from table\_name;*

update

*updata table\_name set column\_name=value where condition;*

Alter

*alter table table\_name modify column\_name datatype;*  
*alter table table\_name drop column column\_name;*  
*alter table table\_name add column\_name datatype;*

Delete

*delete from table\_name where condition;*

Rename

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

Drop

*drop table table\_name;*

Truncate

*truncate table table\_name;*

View

Create view

*create view as select \* from student;*

Drop view

*drop view view\_name;*

Update view

*update customer\_view set age=35 where name=”raman”;*

Display view

*select \* from customer\_view;*

User

Create user

*create user Mohit identified by mohit123;*

Alter user

*alter user Mohit identified by mk123; //change password*

Drop user

*drop user Mohit;*

Grant permission to user

*grant all on student to Mohit;*

*grant insert, update, delete on student to Rohit;*

Revoke permission from user

*Revoke trigger from Mohit;*

Joins

*Select table1.column2, table2.column2, table1.column3, table2.column3,... from table1 innner join table2 on table1.column1=table2.column1*;   
*Select table1.column2, table2.column2, table1.column3, table2.column3,... from table1 left join table2 on table1.column1(+)=table2.column1*;   
*Select table1.column2, table2.column2, table1.column3, table2.column3,... from table1 right join table2 on table1.column1=(+)table2.column1*;   
*Select table1.column2, table2.column2, table1.column3, table2.column3,... from table1 full join table2 on table1.column1=table2.column1*;

Database

Create Database

*Create database MohitDB*

Drop Database

*Drop database MohitDB*

Use Database

*Use database MohitDB*

Show Databases

*Show databases*

Indexes

Create index

*create index index\_name on table\_name(class,marks);*   
*Create unique index index\_name on table\_name(rollno);* or   
*Create unique index index\_name on table\_name(rollno,admission\_no);*

Drop Index

*drop index index\_name;*

Describe table

desc student;

SQL Clauses

Group by clause

*select \*from student group by class*

LIMIT

*select \*from student LIMIT 1,7*

Order by Clause

*select \*from student order by rollno ASC/DSC*

Subquery

Display subquery

*select \* from student where Rollno in(select rollno from student2 where rollno in(127));*

Insert subquery

*insert into student(rollno, name, class) select rollno, name, class from student2 where rollno=34;*

Delete subquery

*delete from student where rollno in(select rollno from student2 where rollno=28);*

Update subquery

*update student set name=’Amandeep’ where rollno in (select rollno = 67);*

Operator in SQL

Like

*select \*from student where name like ‘moh%’;*

|  |
| --- |
| ‘%’ for one or more character  ‘\_’ only one character |

And

*select \* from student where name=’aman’ and rollno=107;*

Or

*select \* from student where name=’aman’ or rollno=107;*

In

*select \* from student where name in(‘mohit’);*

*select \*from student where name in(‘aman’, ‘rohit’, ‘mohit’);*

As

*select class as “Branch” from student;*

Between

*select \* from student where marks between 60 and 70;*

Not

*select \* from student where not marks = 79.5;*

*select \* from student where marks not in(70.9, 70.5);*

keys

Primary key

*Create table abc(eno number primary key, name varchar(30));*

Foreign key

*create table student(rollno decimal, name varchar(30), foreign key(rollno) references studentResult(rollno));*

Super key

*create unique index superKeyIndex on student(rollno, name, city);*

Cases

*select s.rollno, r.marks,*

*case*

*when marks between 60 and 70 then ‘first divition’*

*when marks between 80 to 100 then ‘Marit’*

*end as division*

*from student s, result r where s.rollno=r.rollno;*

Functions

min()

*select min(marks) from student;*