DBMS

MODULE 4
Structured Query Language(SQL)

Written & Compiled by,
Prof.Priyanka Bhilare

3. Structured Query Language.

* Overview of SQL :-

SQL is a database computer language designed for the retrieval and management of data in relational database.

SQL can perform following operations :-

1) SQL can retrieve data from a database.
2) SQL can insert records in a database.
3) SQL can update records in database.
4) SQL can create new database.
5) SQL can set permissions on tables.

* SQL consists of following languages :-

i) Data Definition Language (DDL).
ii) Data Manipulation Language (DML).
iii) Data Control language (DCL).
iv) Transaction Control Language (TCL).

i) Data definition language (DDL).

Before DB can be used it must be created. DDL is used for defining a language of DB to the computer. So, SQL DDL provides commands for ~~defiining~~ defining the relation, deleting the relations and modifying existing relation schema.

Scanned with CamScanner

The most important DDL statements in SQL are :-

1) Create :- It helps to create new table.

syntax:-

Create table tablename (Columnname 1 datatype(size),
Columnname 2 datatype (size),
:
Columnname n datatype (size));

eg:-

create table student (sid int(9), sname varchar(10), smarks int(10), DOB date);

2) Drop :- A table may be dropped or destroyed using Drop table command.

syntax:-
Drop table tablename;

eg:- drop table student;

3) Alter :- It is used to modify an existing table. It manages changes like adding or dropping a column, changing a column ~~to~~ definition as well as adding or dropping a table constraint.

Scanned with CamScanner

i) To add a column :-

syntax :-
alter table tablename add columnname datatype(size);

eg:- alter table student add grade int(9);

ii) Add multiple columns :-

syntax :-
alter table tablename add columname1 datatype(size), add columnname2 datatype(size), add columnname3 datatype(size), ... add columnnameN datatype(size);

eg:-
alter table student add emailid varchar(15), add deptname varchar(20);

iii) To add column at the first :-
eg:- alter table student add branch_name varchar(9) first;

iv) To add column in between :-
eg:- alter table student add contactno int(10) after sname;

v) To change datatype of column :-
syntax :- alter table tablename ~~alter~~ modify column columnname datatype(size);

Scanned with CamScanner

OR

alter table tablename modify column columnname1 datatype (size), modify column columnname2 datatype (size);

vi) To drop column :-

Syntax:-

alter table tablename drop column columnname;

OR

alter table tablename drop column columnname 1, columnname 2, drop column

4) Rename :-

Syntax:- (for table)

rename table oldtablename to newname;

eg:-

rename table student to student_info;

Syntax (for column)

alter table tablename ~~rename~~ change column oldcolumnname newcolumnname datatype (size);

eg:- alter table student_info change column sname student_name varchar (15);

5) Describe :-

Syntax:- desc tablename;

eg:- desc student_info;

Scanned with CamScanner

2) Data Manipulation Language (DML):-

These commands are used for storing, retrieving, updating and deleting data. These commands basically deals with manipulation of data present in the DB.

1) Insert Command:-

Insert command is basically used to insert new records in the table.

Syntax:-

insert into tablename values (value1, value2, value3, ... valuen);

eg:- insert into student values (1, 'priyanka', 75, '1991-02-06');
YYYY-MM-DD

- Insert values for specified columns:-

syntax:- insert into tablename (columnname1, columnname2) values (v1, v2);

eg:- insert into student (id, name, DoB) values (2, 'Tejas', '1987-13-04');

- To insert multiple values for specified column:-

eg:- insert into student (id, name, DOB) values (3, 'Snehal', '2004-13-05'), (4, 'vipul', 1990-02-06);

Scanned with CamScanner

2) Update Command :-

Update statement is used to modify data in a table.

Syntax :- update tablename set columnname = newvalue where columnname = oldvalue;

eg :- update student set id = 6 where name = 'priyanka';

- multiple updates :-

eg :- update student set id = (7, 8) where name = ('Vipul', 'Snehal');

- update several columns in a row :-

eg :- update student set id = 7, DOB = '2005-18-12' where name = 'priyanka';

eg :- update student set id = (4, 5) where sname = ('priyanka', 'Snehal');

3) Delete Command :-

It is used to delete rows from a table.

Syntax :- delete from table where condition;

eg :- delete from student where id = 1;

Scanned with CamScanner

4) Select Command :-

It is used to retrieve data from the DB.

Syntax :- select columnname(s) from tablename where condition;

eg :- select sid, name from student where sid=1;

eg :- select * from student;

(*) Constraints in DBMS :-

Constraints are placed on columns or tables. They limit the data that can be inserted into tables.

- **Integrity constraints :-**

Integrity constraints are rules that are to be applied on DB columns to ensure the validity of data. Every time data is evaluated or checked against the constraint and only if result comes out to be true; then only data inserted into the column. Integrity constraints prevent ~~da~~ invalid data entry into the table.

Scanned with CamScanner

Integrity constraints can be divided into following

A] Domain Integrity Constraints :-

There are constraints on domain value, therefore these constraints are used to impose restrictions on some particular column.

Types of domain Integrity constraints :-

i) Not null constraint :-

By default all columns in a table allows null values. When a Not null constraint is enforced on a column or set of columns, it will not allow null values.

⊙ syntax :-

create table tablename (columnname1 datatype(size), columnname2 datatype(size) not null, columnname3 datatype(size));

eg :- create table student (sid int(4), sname varchar(10) not null, DoB date);

After creating table not null constraint can be added as,

⊙ syntax :-

Alter table tablename modify

columnname datatype (size) not null;

eg:- alter table student modify DOB date not null;

- Dropping not null constraint :-

syntax :-

Alter table tablename modify columnname datatype(size) null;

eg:- alter table student modify DOB date null

~~ii) Unique constraint :-~~

Not null constraint cannot take null values but can take duplicate values.

① insert into student values (1, 'priyanka', null);
→ error will be displaced on the screen.

② insert into student values (2, 'sneha', '2019-12-03');
— record added to the table.

③ insert into student values (3, 'Aakash', '2019-12-03');
— record added to the table ~~as~~ although we have duplicate entry for DOB.

Scanned with CamScanner

ii) Default constraint :-

Default constraint is used to provide default value for a column. Default value is added to all new records if no other value is specified.

⊙ Syntax :-

Create table tablename (columnname 1 datatype(size), columnname 2 datatype(size) default 'value');

eg:- create table student (sid int(9), sname varchar(10) default 'priyanka', D.B date);

① insert into student values (1, 'Rajesh', '1991-12-07');
— record inserted in the table

② insert into student (sid, dob) values (2, '1987-13-02');
— record inserted with default student name as priyanka.

⊙ After creating table default constraint can be added as:

Syntax :-
alter table tablename alter columname set default 'value';

Scanned with CamScanner

eg:- alter table student alter ~~columnname~~ sname set default 'priyanka';

⊙ Droping default

syntax:- alter table tablename alter column columnname drop default;

eg:- alter table student alter column sname drop default;

iii) Check constraint:-

Check constraints define a condition that each column must satisfy.

⊙ syntax:-

create table tablename (columnname 1 datatype(size), columnname 2 datatype (size), columnname 3 datatype(size), check (condition));

eg:- create table student (id int(9), age int(10), marks int(9), check (age >= 18));

① insert into student values (1, 20, 85);
— record inserted successfully.

② insert into student values (2, 15, 87);
— check constraint violeted error as age is not >= 18.

Scanned with CamScanner

show index from tablename; use select constraint_name from ... where table_name = tablename

- After creating table check constraint can be added as follows:-

Syntax:-
alter table tablename add constraint constraintname check (condition);

eg:-
alter table student add constraint chk1 check (age >=18);

- To drop check constraint:-

syntax:- alter table tablename drop constraint constraintname;

eg:- alter table student drop constraint chk1;

B] Entity - Integrity Constraints :-

As the name implies, it is applied on each entity i.e. on individual row. It specifies that each row should have uniqueness.

There are two types of Entity Integrity constraints:-

Scanned with CamScanner

1) Unique constraint:-

Unique key constraint does not allow duplicate values but it allows null values.

⊙ Syntax:-

create table tablename (columnname1 datatype(size), columnname2 datatype(size), columnname3 datatype(size) unique);

eg:- create table student (sid int(9), sname varchar(9), smarks int(9) unique);

① Insert into student values (1, 'priyanka', 75);
— record inserted successfully.

② insert into student values (2, 'Aakash', 75);
— Error : duplicate value for 'smarks.

③ insert into student values (2, 'Aakash', null);
— record inserted successfully.

⊙ After table created if you want to add unique constraint then the syntax is given below:-

syntax:- alter table tablename add constraint constraintname unique (columnname);

eg:- alter table student add constraint u1 unique (smarks);

Scanned with CamScanner

Date
Page

⊙ Syntax to drop unique constraint :-

alter table tablename drop index constraintname

eg:- alter table student drop index u1;

Note:- While creating table if unique constraint is added then name of the unique constraintname is not specified, in this case if you want to drop a unique constraint then use following syntax to see unique constraintname ~~as~~ and drop the unique constraint accordingly.

show index from tablename;

2) Primary key constraint :-

Primary key is used to prevent the duplication of values within rows of specified column or set of columns in a given table. It doesnot allow null values.

Primary key = unique + not null.

⊙ Syntax:- create table tablename (columnname1 datatype(size), columnname2 datatype(size) primary key, columnname3 datatype(size));

eg:- create table student (sid int(9) primary key, sname varchar(15), smarks int(10));

Scanned with CamScanner

- After creating table primary key is added as below:-

Syntax:- alter table tablename add constraint constraintname primary key(columnname);

eg:- alter table student add constraint pk1 primary key (sid);

- To drop primary key syntax is,

Syntax:- alter table tablename drop primary key;

eg:- alter table student drop primary key;

1. insert into student values (1, 'priyanka', 75);
→ record added successfully.

2. insert into student values (1, 'Riya', 90);
→ record doesnot added as duplicate value for sid.

3. insert into student values(null, 'Atul', 80);
→ error; primary key cannot be null.

Scanned with CamScanner

c] Referential Integrity constraint :-

It is used to establish parent-child relationship (primary - foreign key relationship) between two tables. Primary key is defined in parent table and foreign key is defined in child table.

P.K. F.K.

sid	name	marks	deptid
1	Riya	70	10
2	Uttraj	75	20
3	Shweta	80	10
4	Atul	90	30

child table (student table)

deptid	deptname
10	Comp
20	I.T.
30	EXTC

P.K.

Parent table (department table)

⊙ **syntax :-**

⇒ Create parent table first.

create table department (deptid int(3) primary key, deptname varchar(20));

⇒ Create child table

create table student (sid int(10) primary key, name varchar(15), marks int(20), deptid

int(10), constraint fk1 foreign key(deptid) references department(deptid));

- insert into student (5, 'Tejas', 95, 30);
record added successfully.

- insert into student (6, 'Vedika', 87, 40);
record doesnot added as deptid 40 is not present in department table.

⊙ After creating tables foreign key can be added as follows:-

alter table student add constraint fk1 foreign key (deptid) references department (deptid);

⊙ To drop foreign key

- alter table student drop foreign key fk1;
- desc student;
after dropping still mul index is there therefore use foll. syntax.
- alter table student drop index ~~foreign key~~ fk1;

Scanned with CamScanner