

SQL

15 Structured query language which will be used to communicate with database that we have created.

Data → any useful information

20 * Table → organizing data in form of rows and columns

* Record → Entities / Entries / Rows in a table

25 * Fields → Columns

30

ER diagram

E → Entities (objects which we can feel) (rectangle)
 Attributes → Properties (oval shaped)

5

Types of commands in SQL

- DDL
- 10 → DML
- DCL
- TCL
- DQL

15 → DDL (Data Definition language)

→ used to define data in database

eg Create table, alter table, drop table,
 truncate table.

20

→ DML (Data Manipulation language)

Manipulates data

→ insert, update, delete

25

→ DCL (Data Control language)

Grant, revoke

30

TCL (Transaction control language)

- Commit to store changes in database permanently
- Rollback to undo the commit
- Savepoint to bookmark the commit

5

DQL (Data Query language)

- Select command to retrieve data from DB

10

- Schema → defines structure of database
that every object of some type
will follow.

15

We can create schema using
DDL commands

20

TYPES OF KEYS

- Super key

- Candidate key

25

- Primary key

- Alternate key

30

Data types in SQL

* Number

↳ int (1, 2, 3...)

5 ↳ float (decimal)

↳ Bool (0, 1)

* Character

↳ Char (n) → Static (not more not less)

10 ↳ Varchar (n) → dynamic less limit

* Date and time

↳ Date YYYY MM DD

↳ Year YY / YYYY

15 ↳ Datetime YYYY MM DD HH MM SS

Creating table

20 create table employee (EntryNo int, FirstName varchar(50), LastName varchar(50))

Employee int)

FirstName varchar(50),

LastName varchar(50),

);

25

Inserting values

30 INSERT INTO employee (EntryNo, FirstName, LastName)

VALUES (123, 'Rajesh', 'Ahuja')

Select

5
SELECT *
FROM employee;

Update

10
UPDATE employee
SET EntryNo = 150
WHERE FirstName = 'Rajesh';

Constraints

15
Conditions we apply on columns

eg create table table 1 (

20 ID int NOT NULL, constraint

ID column will not accept null values

GR

In already created table,

25 ALTER TABLE employee

ALTER COLUMN EntryNo int NOT NULL;

Primary key → To uniquely identify the record table

Properties

5

- ↳ It is not null
 - ↳ It should be unique
- eg Roll num

10

UNIQUE

15

ID int UNIQUE

- ↳ It will not allow any duplicate value.

20

Foreign key

- ↳ Constraint → used to limit some values
- ↳ used to prevent actions that can destroy link b/w tables

eg Create table customers(

25 c_id int NOT NULL Primary key;
 c_name varchar(100)
);

Create table orders(

amount,

30 c_id int foreign key references customers(c_id
 foreign);
 key

Check

↳ To make validation

Eg. age wit CHECK (age >= 18)

5

default

↳ To set default value if user has
not given any

10

Eg. city varchar(100) Default 'Delhi'

15

Like from

select *

from student

where name like 'a%';

20

search for name with 'a' as
first char

'%.2'

↳ z with last char

25

'%.b'

↳ 2nd char 'b'

(Top)

30

↳ Select top 2 *

↳ will return 2 rows

delete

delete from Student
where name = 'Ab'

5

↳ remove row from
club

Scalar Query

- 10 → Same as nested query
- helps in limit resultant set
- Return only 1 value
- If values returned are more than 1, error will come
- 15 → If no values are returned, then null will be returned

e.g. count, max, min, avg, sum

20

Row-NUMBER

25 select *, ROW_NUMBER () OVER (ORDER BY ~~product~~)
row_num
from product;



30 creates a extra column 'row_num', giving ~~date~~
S.no. to columns sorted by ~~product~~
by $\text{mod } 10$

Exists

↳ used to check whether the result of a correlated nested query is empty or not,

5 result of EXIST is a boolean value true / false

Correlated Subquery

10 → Interrelated queries

→ each subquery will be executed only once

Inner query is driven by outer query

15 In nested, inner query has more privileges

e.g. EXIST

Arithmetic operators

20

Select 5+5; 110

Select 10-8; 2

Select 10/5; 2

Select 10%5; 0

25

Comparisons
more than >

less than <

not equal <> or !=

Logical

30

AND, OR

most basic BETWEEN 10 AND 30

Wildcard operators

% , _

5 b% b%
 List starts with b Ends with b

? . b%

 List return all names with 'b' in it

10 e% ?

 ↓
 2nd char e

15

Aggregate function

	Marks	No of rows
Count()	10	Count (Marks) = 4
max ()	20	Max (Marks) = 50 ↑ max value of
20 min ()	30	Column
Sum ()	40	min (Marks) → 10 Sum (marks) → 100

25

Order by

Select *

from table

order by Marks DESC;

30

↓
 Descending order

For aggregate function having function used

HAVING count (marks) > 1;

5

When - Then function

when

case when marks > 20 then 'pass'

10.

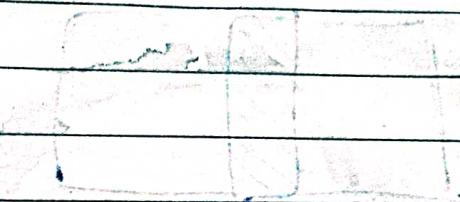
when marks < 20 then 'fail'

else 'back log'

end as 'Result'

from std:

15



20

empidont exists with

function argument, then

25

function exists with

an empty function body

30

Joins

↳ Used to combine data or rows from 2 or more tables based on common fields between them.

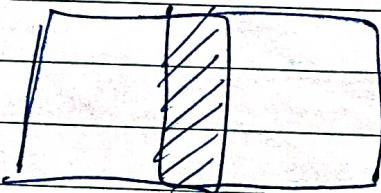
5

default

left join, right join, inner join, outer join
guaranteed to get value of 1st table guaranteed to get value of 2nd table.

* 10 Select table1.column1, table1.column2, table2.column
From table1 join table2
ON table1.matching column = table2.matching column

15



Alter command

20

↳ To add other columns after creating a table.

Eg - alter table tablename
add phoneno varchar(10);

25

alter table tablename
drop column phoneno;

To delete column

30

Truncate / Drop

DATE: _____
PAGE: _____

To delete all table values

truncate table tablename; → to make table empty
drop " " ;

5 It will also remove entire schema | to delete table

Transaction

10 Unit of work

ACID properties

15 A → Atomicty → Either transaction should execute or not at all

C → Consistency → State before transaction is same as

B → Consistency → State after transaction

20 I → Isolation → transaction should run independent of each other

D → Durability → After transaction, changes to data persist.

25

Clone of table

Create table tablename like original table.

30

To insert same values as table original

create table tablenew as Select *
from original - table

5

OR

Insert into tablenew Select *
from original Table

10

DCL commands

Data control language command

15

Grant

↳ I want privileges / grant permission to
only few users for security

20

Syntax

↳ GRANT privileges ON object TO user;

Revoke

25

↳ To take privileges / permission from
user

REVOKE privileges ON object FROM user

30

Eg GRANT ALL ON EMPLOYEE TO RIVA

↳ Receiving all permissions of Employee table
Riva

TCL commands

Transaction control language

Commands

Syntax

- Commit
- Rollback
- Savepoint

Commit ;

Rollback ;

Savepoint & name ;

→ Commit is like saving changes

→ Rollback is like undo changes

→ Savepoint is like bookmarking changes
(temporary save)

Substring

starting index

select substr('Hello', 1, 2)

↓
He

length of
substring

ASCII

String format

select ASCII(A); select left('Geeks', 3)

↓ 65

select char(65);

↓ A

Geeks

lowercase

uppercase

select concat ('A', 'For', 'Apple')

↓ Apple

select len('Geeks') → 5

Views

↳ Virtual table based on the result set of an SQL statement

5

e.g. table Table 1 (C.name, C.ID)

table Table 2 (P.name, P.ID)

10

View (C.name, P.name) or definition

only shows these

Advantages

15

→ Simplicity

→ Security

20

CRUD operations

C → Create

R → Read

U → Update (Replace)

D → Delete (Drop)

25

e.g. Create view view_name AS

Select column - - - - -

From table

30

Where condition

Regex

Patterns

[a b c] → a or b or c

[^a b c] → exclude a, b, c

[A-Z] → A to Z

5 [a-z A-Z] → a to z or A to Z

[]? → 0 or 1

[]+ → 1 or more

[]* → 0 or more

10 []{n} → n times

[]{n, x} → range times

Cg phone no.

[8 9] [0-9] {9}

15 ↓
first 2 digits blue 0-9
digit blue 0-9

Trigger

20 When ever special event occurs in database,
a procedure occurs

create trigger [trigger name]

25 before / after

[insert / update / delete]

on table

(for each row)

Set — — —