



→ Database Concept

→ File based System

→ Prior to database days data was stored in file based system

→ A file based system stores all the data in 1 big file

→ Important points

→ Store all data in 1 big file

→ There is no structure associated with the data

→ Common data may be repeated many number of times

Eg: If there are multiple departments within one organisation, then the data may be stored in multiple files for different departments even if some data is common across some departments it will be repeated and stored in different departments

→ Limitations & disadvantages

(i) → Difficulty in access: There is no structure in file based system All the data is simply stored in the file in order to access such type of data specific type of application programme have to be used written.

(ii) → Data Redundancy (duplicated data): refers to storage of same data multiple times.
: wastage of storage, it is a bad practice



(ii) ↳ Data inconsistency (caused by data redundancy):

Multiple copy of the same data when do not match.

(iv) ↳ Data isolation: It refers to a situation where data of one file cannot be mapped to other related file in the absence of link or common format

(v) ↳ Data Dependence: Since the data stored in file based system have no standard structure, a specific special application is needed to access the data.

The closed relationship between the data stored in file and the application programme that maintains whose file update & maintain those file are called data dependency.

(vi) ↳ Data sharing / Central issue: with file based system there is no provision to have control over data access.

: All data is available to all type of users



* Introducing Database system

(i) Database

Database is an organised collection of an interrelated data so, that it can be easily accessed, managed and updated.

(ii) DBMS (Database Management System)

It consists of a collection of interrelated data and a set of programme to access those data

* The GOAL of DBMS

If is to provide an environment that is both convenient and efficient to use in retrieving and storing database information

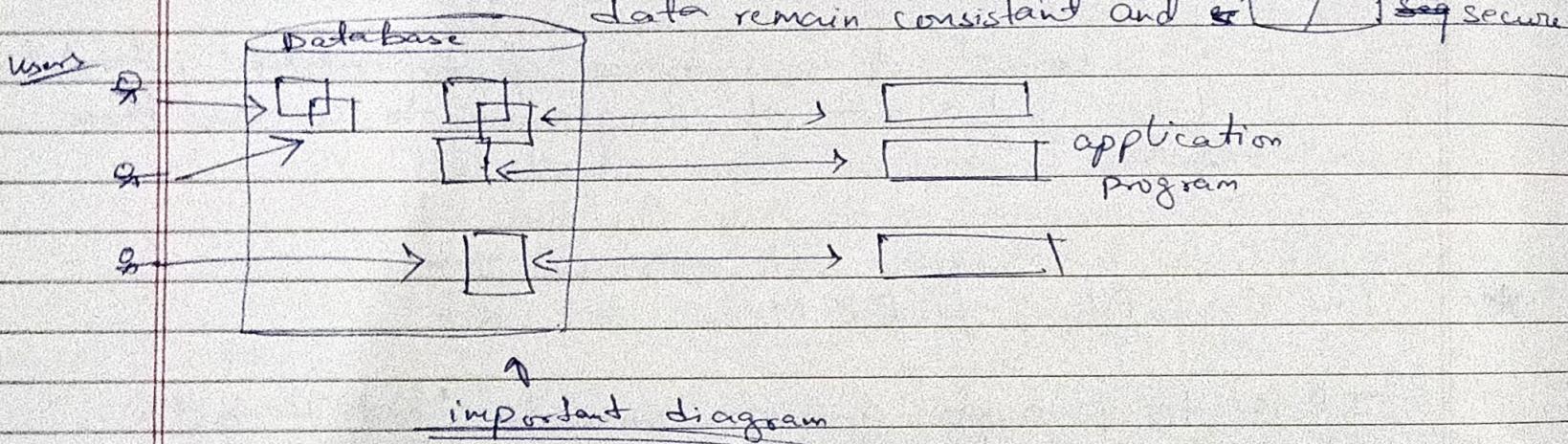


Database / DataBase Management System

Purpose of Database (Advantage)

Important question

- 1) Database reduces data redundancy \rightarrow Duplication of data is known as data redundancy
- 2) DBMS store all data at one place and all the application program can refer to the same. The centralisation ~~of data~~ makes sure that data redundancy is reduced. Data remain consistent and ~~so~~ ^{so} secure



- 3) Database can control data inconsistency by controlling redundancy. The ~~inconsistency~~ inconsistency can also be controlled. DBMS ensure that any changes made to either of the two entry is automatically made to the other.
- 4) DB facilitate sharing of data \rightarrow It means individual piece of data in the DB ^{may be} mode is shared among different users



Q) What is inconsistency? (Board PQA 2022)

→ Some information may be duplicated in several places and all copies may not be updated properly. This is known as ~~inconsistency~~ inconsistency.

5) Database enforces standards → The DBMS can ensure that all the data i.e. centrally stored follow the standard structure and format
e.g. → stored in table

6) DB can ensure data security. → only authorised user should be allowed to access the database and their Identity should be authenticated using a username and password.

→ Terms of DBMS

Def → 1) Database Schema: (Logical architecture of a database)
(Empty Design)

Q) What is DB schema? → A DB schema is a structure or design that represent the logical view of the entire Database. It define how the data is organised & how the relationship among them are associated. It is a blueprint, it actually does not have any data.

char → fixed length, char(6) → 256 BLOB & BLOB

varchar → variable length, var char(10) → max. 10.

Date float → float(6,2) → float (total len, decimal len)

(B)

Tasks

i) Create a table 'stud<roll>' with colm: roll no, RollNo, Name, Marks

./sql.sh

> create table stud03 ←
(roll_no int(3),
name varchar(15),
marks double);

(columns)
(Attributes)

roll-no	name	marks

Q) Insert 5 records in table stud 03

→

> insert into stud 03
values (101, 'aaa', 67.2);

> insert into stud 03
values (102, 'bbb', 50);

}

,

,

Q) Display all records of table stud 03

→

> select * from stud 03;

> select name from stud 03
where marks > 90;

Selena ~~set~~ ~~2020~~
→ > DESC stud23 | : header on
Date _____ | mode col } Table dictane to form



Add column to existing table

① Add age int(2) to stud23.

→ > Alter table stud23
add age int(2);

Update

> Update stud 23
set age = 16
where roll_no=101;

Φ Terms of DBMS

(2) Database Instance → The data stored in Database at a particular moment of time is called instance of Database.

→ for eg: we have a ~~similar~~ ^{single} table stud 23 in the Database. Today, the table has 5 records so, today, the instance of Database has 5 records. We are going to add another 15 records in this Table by tomorrow, so the instance of Database will have 20 ~~instance of Data~~ records in the Table.

3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20

The database instance is related term to DB schema. DB ~~schema~~ instance can change over time whereas schema is generally static in nature.

Constraint → set of rules

↳ NOT NULL, unique, check (condition)

Date _____ / _____ / _____



3) Meta Data / Data dictionary

→ Information about data

Def → Meta Data refers information about data. Data in the table have certain properties and attributes such as field, data type, length, Description, ~~constraint~~ Constraint.

Meta Data					
field	Data type	length	Description	Constraint	
roll-no	integer	3	roll no of students	1 - 500	—
Name	varchar	15	—	—	—
Marks	double	—	—	—	—

→ Meta data is stored in data dictionary

4) ~~#~~ Data Dictionary

→ Consist of 2 words.

↳ Data → simply means information

↳ Dictionary → Where this information is available

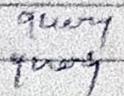
5) ~~#~~ Data Constraints

→ The rules applied on the data column of a table

Eg → NOT NULL, unique, check, primary key

→ Command prompt | Step 202 out of 202 - [open records] 

Date: / /

6) ~~#~~ query  

→ A query is a type of command that retrieves data from a database table. A query can be either select query or action query

7) ~~#~~ Database engine 

→ The underlined software component used to read, update and delete data in DB

★ 8) RDBMS (Relational Data Model)

1) Relation (Table) → A table is a collection of interrelated row & column

2) Domain → A domain is a unique set of values from which the actual values appearing in a given column are drawn

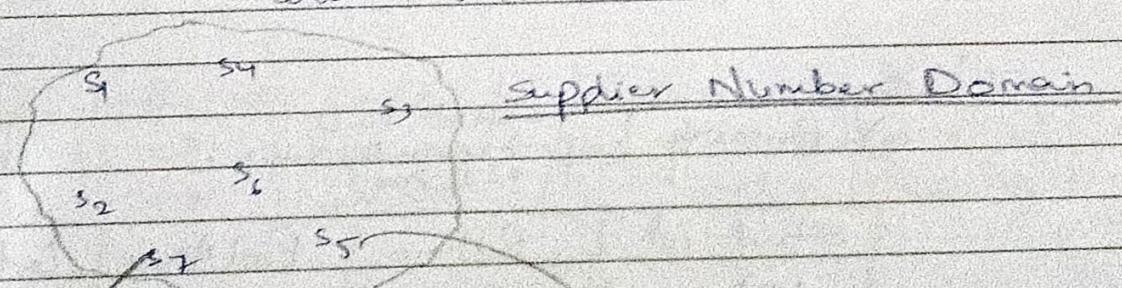


Table: shippers

Supp_no	Item_no	Qty
S3	-	-
S4	-	-
S5	-	-
S1	-	-

Table: suppliers

Supp_no	Name	City
S2	-	-
S3	-	-
S4	-	-
S5	-	-



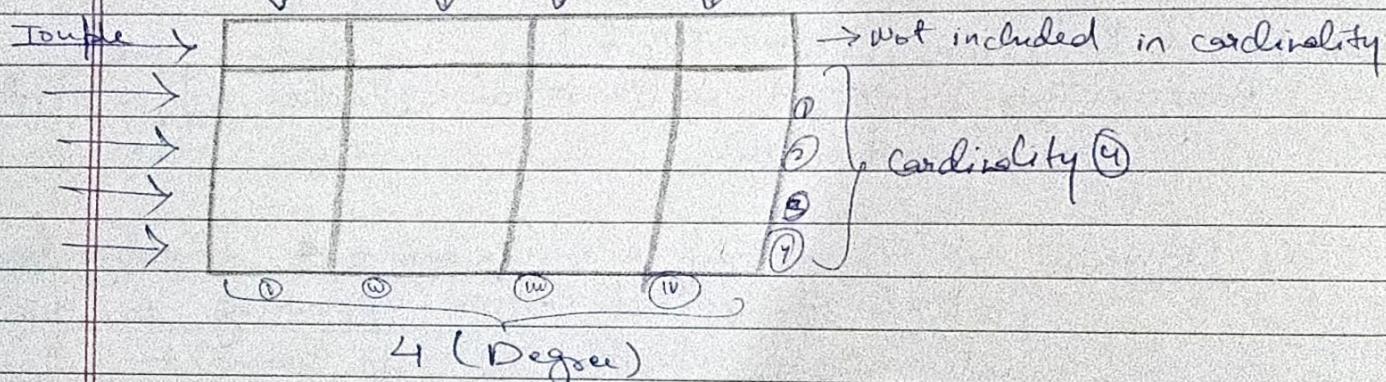
Date / /

3) Tuple (Records) → A row in a relation is known as a tuple or records

4) Attributes → A colⁿ in relation is known known as an attribute.

★ 5) Degree → The number of columns in a relation is known as its degree

★ 6) Cardinality → The no of rows in a relation is known as its cardinality



7) keys

a) Primary key → Not Null
→ Uniquely Unique
→ Primary Key

Stud id.	Roll no	First Name	Last Name	Gmail
101	11	a	x	abc@gmail.com
102	12	b	y	xyz@gmail.com
103	13	c	z	pqr@gmail.com

→ A primary key is a colⁿ group of co in a table that can uniquely identify tuple within the relation

→ NOT NULL #3rd Q-II

~~For~~ Candidate key - Primary key = Alternate key

Date / / - YYYY-mm-dd



- Cannot be duplicate i.e. same value ~~so~~ cannot appear more than ~~one~~ one time in a table
- Cannot be blank or Null

c) Write SQL Command to create stud 23 table with stud_id, class, section, gender, name, dob & Marks as attribute where ~~stud~~ stud_id is the primary key

→

```
> create table stud 23
( stud_id varchar(20) not null Primary key,
  class int (3),
  section char(1),
  gender char(1),
  name varchar(20),
  dob date,
  Marks double);
```

d) Create a table # emps to following structure.

Name structure	id int	First-name varchar	Last-name varchar	user_id varchar	salary float
-------------------	-----------	-----------------------	----------------------	--------------------	-----------------

- i) id should declare as primary key
- ii) user_id should be unique
- iii) Salary must be ~~*~~ greater than 5000
- iv) first_name must not be blank.

→ > create table emps
 (id int (3) primary key,
 first-name varchar(10) not null,
 last-name varchar(10),
 user-id varchar(5) unique,
 salary float (7,1) check (salary > 5000), check (salary > 5000))

Q) ~~Explain~~ Eliminating Redundant data (with the key word Distinct)

Syntax

> Select Distinct <col> from <table>;

With a SQL command

Q) list the eight different categories of mobile movie

movie → musical, action, horror, adventure, action, comedy

(ii) Table name - sports

col - student-no, class, game1, name, game1, grade1, game2,
 grade2
 grade1 = 2 ...

~~Q) Work SQL cmd to display name of student in grade C who have grade C in either game1 or game2 or both where~~

→ Select name from sports
 & where grade1 = 'C' or grade2 = 'C';



Date / /

b) Candidate key

Candidate key

Primary key

Stud_id	Roll_no	First Name	Last Name	Email

Alternate key

Def → All those attributes of a table that can act as a primary key are known as candidate key.

c) Alternate key

Def → All those candidate key which are not primary key are known as Alternate key.

d) Foreign key

stud_id	f_name	l_name	course_id	Table: abc		Table: xyz	
				Primary Key	Foreign Key	Primary Key	Foreign Key
-	-	-	1	1	1	Course_id	Course_name
-	-	-	2	2	2	2	2
-	-	-	3	3	3	3	3

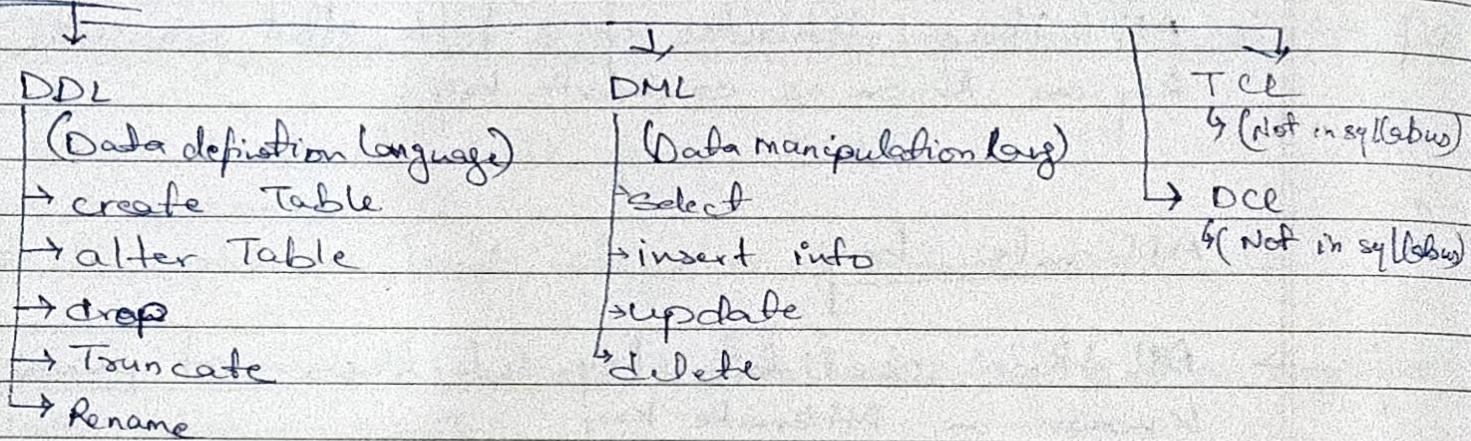
Def * → A foreign key in a table is a column whose value is derived from primary key of some other table.



8) SQL

- Stand for Structural Query Language
- A set commands that is used for accessing & manipulating SQL database

Commands



Q) Difference b/w truncate & delete command/statement

Delete

syntax
Delete from <tableName>
where <condition>

Truncate

truncate table <tableName>

① DML command

- ② There may be 'where' clause in order to filter the record
- ③ Used to delete specific row

DDL command

- There may not be used 'where' clause
- Used to ~~delete~~ all the records from the table



Differentiate b/w DDL & DML cmd.

DDL

Stand for data definition language.
Used to create DB schema &
can be used to define some
constraints.

It basically defines the col
of the data.

'Where' clause not used

DML

Stand for Data Manipulation lang.
Used to add a record, f
update a record/ table

It add / update a record from
a table

'Where' clause is used.

SQL Elements

① Literals

- ↳ character literal
- ↳ number literal

② Data Types

③ Null

④ Comments

- ① Literal → constant value
→ fixed value

→ In general, literals refer to a fixed data value
it may be ① character literal or ② number literal

Raj Kumar

19

All character literals are enclosed in single or double
quotation marks



Date _____ / _____ / _____

→ Numeric Literal can be real literal or real literal
 int
 float
 double

(ii) Data Type

→ A data type in SQL define as a type of data that any colⁿ can store.

→ Classified in 3 categories

String data type

numeric data type

date & time datatype

a) String → used to specify string

① char (size) → fixed length string that can contain number, letter, special characters

② Varchar (size) → used to specify a variable length string that can contain number, letter & special characters.

b) Numeric Data Type

① int / integer → int (size)

→ used for integer value.

→ can contain max 11 digits

② float → float (size, d)

→ used to specify a floating point number.

→ its size parameter specify total no. of digits & the no. of digits after decimal is specified by 'd' parameter

→ 8 byte memory



(iii) double (size, 8) → memory → 16 byte
→ same as float

(iv) bool (True/false)

c) date and time

(i) date : used to specify date format 'YYYY-mm-dd'
→ e.g. Dec 30th 2021 → '2021-12-30'

(ii) Date of time → used to specify date & time combination
→ 'YYYY-mm-dd HH:MM:SS'
→ e.g. '30 Dec. 2021, 3:30' → '2021-12-30 15:30:00'

-Diff

Pto
exm Q// Differentiate b/w char & varclr datatype

(iii) Null : if a column in a row has no value, then col^m is said to be null

(iv) Comment : Non executing cmd
→ A text that will not be ~~executed~~ executed it is only for documentation purpose

single line → --
multi line → /* */

ED

Date : / /



Commands

(i) Create Table cmd.

Syntax : → Create Table <table-name>
<column names> <data types>
;

→ If : used to create a new table in a database

(ii) Constraints

(iii) Data Integrity through Constraints

→ Integrity Constraints are a set of rules used to maintain quality of information.

→ Two basic type of constraints are

• Column Constraints

→ apply only to individual columns

• Table Constraints

→ apply to groups of 2 or more col^m.

(iv) Unique Constraint : This constraint ensure that no two rows have the same value in the specified col^m



(i) Primary key: This constraint declares a column as primary key of the table

→ | Not NULL |
| Primary Key |

NOTE (ii) Difference b/w Unique & Primary key

→ There can be multiple unique keys but only 1 primary key.
Primary key is used for searching.

(iii) Default Constraint

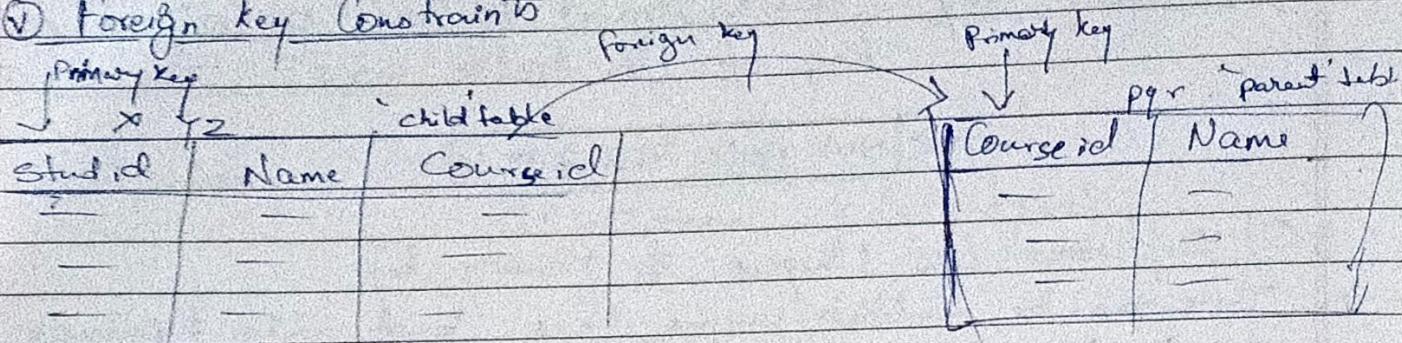
- The default constraint is used to set a default value for a column. The default value will be added to all the new records if no other value is specified.
- Eg: if no value is provided for grade, 'E' will be entered
(grade char(2) DEFAULT 'E',);

(iv) Check constraint: The check constraints are the rules that help to check the inserted data value to table based on a certain condition so that we can validate the newly inserted data value based on specified rules before accepting them to the table

Eg → gross double check (gross > 50000)



⑤ Foreign key Constraints



→ The foreign key constraint is a key used to link two tables together. The table with the foreign key is called 'child table' & table with primary key is called 'parent table'.

★ ⑥ Referential integrity: a set of rules that used to ensure that records in the related table are valid & that user do not accidentally delete or change related data

⑦ Insert data in Table

→ Insert into <table name>
values (value1, value2, ...);

⑧ Select Command

→ Select <column1>, <column2>, ...
from <table name>;

→ Used to fetch the data from a database table which returns data in form of result.



Eliminating Redundant Data (with keyword Distinct)

Syntax → Select Distinct <cols>
From <table-name>;

- The 'distinct' keyword eliminate duplicate row from the result of the select statement
- Select all the rows, use '*' or 'All' keywords

⊕ How to perform simple calculation in SQL

> Select 1 + 6

$$\rightarrow \begin{bmatrix} 1+6 \\ + \end{bmatrix}$$

Pg Q) How to display current date

- windows (oracle) → > select curdate();
- Linux (sqlite) → > select date();

⊕ Scalar Expression with selected field.

eno	ename	comm	(comm * 10)
-	-	1	10
-	-	2	20
-	-	3	30
-	-	4	40
-	-	6	60

scalar exp/

scalar

> Select ename, comm, (comm * 10)

From <table name>;

> Select comm * 10 as 'Total'
from <table name>

else

if you want to perform simple numeric computation on the data to put it in an appropriate format, SQL allows to place scalar expression on the selected field

Eg Name age aggregate
 Write a query to display name,
 Age & aggregate 15
 Label this computation as percentage
 → Select name, age, aggregate/5 as
 'percentage' from ctables

Selecting specific rows ('where clause')

The where clause in select statement specifies the criteria for selection of rows to be returned.

Syntax → select < column names >, < column names >, ...
 from < table name > where
 where < condition >;

Eg → Select name, agg
 from ctables
 where agg > 3/5;

<u>name</u>	<u>agg</u>
—	—
—	—
—	—

When a where clause is present, the database program goes through the entire table, one row at a time & examine each row to determine if the condition is true.

Q) Write a query to display name, age & marks of student whose age ≥ 16 from table student

→ $\text{select name, age, marks}$
from student
where age ≥ 16 ;

Relational operators

$=$	$>$	$<$	\geq	\leq	\neq	$/ =$
equal to	greater than	less than	more than or equal to	less than or equal to	not equal to	not equal to

age ≥ 16

Q) Write query to display member's name member city
not from delhi

→ $\text{select member, city}$
from table;
where city \neq 'Delhi';

def → A symbol used to compare two qty

Logical Operator ~~or~~ and
~~not~~

def → The logical operators 'or', 'and' & 'Not' are used to connect such condition in the 'Where' clause



Q) To display the list of employee details having grade 'E2' or 'E3'

→ Select * from employee
where grade = 'E2' or grade = 'E3';

→ Select * from employee
where grade IN ('E2', 'E3');

Q) To display the list of all employee details as 'E4' but gross < 9000 . Write the command

→ Select * from employee
where grade = 'E4' AND gross < 9000;

Q) To display the list of all employee details whose grade are other than 'G1'.

→ Select * from employee
where Not grad grade = 'G1';



Condition based on a Range (between and)

- The 'between' operator define a range of values that the column value must fall in to make the condition true.
- The range include both lower value & upper value

items

i_code	Descrip	Price	qty
-	Milk	-	20
-	Cake	-	60
-	-	-	40
-	-	-	50
-	-	-	100
-	-	-	10

a) To display the list of items whose qty fall b/w 30 to 50, both inclusive. Write the command

→ Select i_code, descrip, qty from items where qty between 30 and 50;

from

Q) Write a query to show name & aggregate (of marks) of students of those students whose aggregate not in range 380 - 425.

N
T
O

The operator 'Not between' is the reverse of 'between' operator i.e. the rows not satisfying the b/w condition are retrieved

→ Select name, aggregate from student where aggregate not between 380 and 425;

Condition based on a list
(IN)

→ To specify a list of values, IN operator is used. The IN operator selects values that match any value in a given list of values
(or values)

Q) To display a list of members from Delhi, Ranchi, Patna, Gaya

city	
-	→ Select * from members
-	where city in ('Delhi', 'Ranchi', 'Patna', 'Gaya');
-	
in	

Condition based on Pattern match Like
Not Like
 - (%, _ (underline))
 (wild card character)

→ The '%' character match any substring
 → where <col> Like 'A%';
 (→ agr 'A' is starting character)

→ The '_' character match any character
 → where <col> Like '_ _ _';
 (→ 4 letter int data to fit)

--- '%'
 (→ 4 to 4 की जटि के लिए

emp		
eno	ename	salary
1	P	1000

Order by last name |



→ Sorting result asc
desc
(order by clause)

- def → The 'order by' clause allow sorting of query result by 1 or more col → The sorting can be done either ascending or descending order (Default ascending order)
→ The data in the table is not sorted, only the result that appear on the screen are sorted.

→ `> select * from emp
order by ename asc;`

- Q) To display the list of students having marks > 400 in alphabetical order of their name

student

name	aggregate
A	400
B	500

`> select * from student
where aggregate > 400
order by name asc;`

- PY Q)

eno	ename	grade
1	P	A
2	C	A
3	B	A
4	S	B
5	L	C

`> Select * from emp
order by grade desc, ename asc;`

a) grade grade को desc तरीके से A से Z से asc करें।

1	L	C
2	G	B
3	P	A
4	O	A
5	B	A

emp			
eno	ename	grade.	Marks
Date	/	/	



∅ Modifying data with update cmd

df → Sometimes you need to change some or all of the values in an existing row, this can be done using update cmd.
The update command specify the row to be changed using 'where' clause and new data using 'set' keyword

> update emp
set Marks = 90; } → can't change it without

> update emp
set marks = marks + 5; → arithmetic

> update emp
set <col> = <val> } specific
where <condition>;

∅ Deleting Data with delete command

syntax →

> Delete from <tablename> } specific
where <condition>; → can't delete in this

> Delete from <table>; → can't do this
without

for example

Count(*) → find no of

Date _____



* Alter table Command

→ existing table schema change देते हैं।

def → Used to change definition of existing table.

→ Uses

Add a column

Changing table name

Syntax → > Alter table <table-name> {
add <column-name><data type>; } - sqlite
} Add col

> Alter table <table name>
add (<col-name> <data type>); → oracle

> Alter table <table_name> {
Rename to <new name>}; } → Rename

* Drop table command

Syntax → > Drop table <table name>;



11th Oct 2024