13/1/2020

DBMS

Neel Kamal
Date
Page

Q1- What is Database?

A database is a collection of related data which represents some aspect of the real world. A database system is designed to built and populated with data for a certain task.

Q2- What is Data Base Management System?

DBMS is a S/W (tool) for storing and retrieving users' data while considering appropriate security measures. It consist of a group of programs which manipulate the database.

The DBMS accepts the requests for data from an application and instructs the OS to provide the specific data.

In large ~~DBMS~~ systems, DBMS help users and third-party S/W to store and retrieve data.

Q3- Write features of DBMS

Different features of DBMS:-

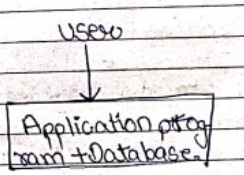
1. Relation based table
2. Isolation of data and application
3. Less Redundancy
4. Consistency
5. Security
6. Real-world Entity Model
7. Query Language

14/1/20

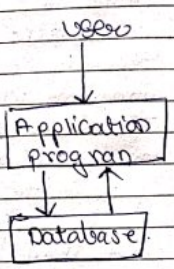
Architecture in DBMS

1. 1 Tier Architecture
2. 2 Tier Architecture
3. 3 Tier Architecture.

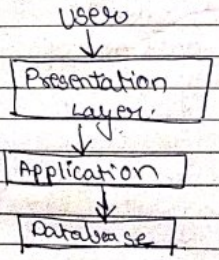
1. 1 Tier Architecture



2. 2 Tier Architecture



3. 3 Tier Architecture :-



20/2/20

Data models :-

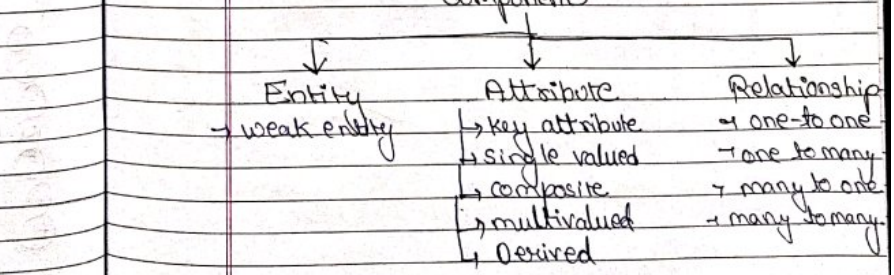
Data models is logical structure that defines no. of components & relation b/w them.
Data models defines how the logical struct of database, how they are connected, how they are processed and how they are stored.

→ ER diagram
→ Relational model.

Entity Relationship model (ER model) :-

It defines the conceptual view of database. It works around real world entity and association among them.

* It is a designing tool.
** It has 3 main component
a) entity b) attribute c) Relationship.



- Entity :- it can be real world object. It may be living or non-living.

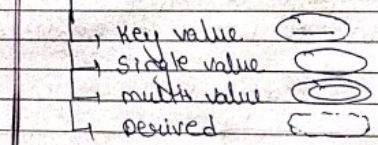
Eg. In the school data entity :- student, teacher, class, course etc.

- Attribute :- It defines properties of an entity.
- Relationship :- It defines relationship b/w two entity.

Representation

1) Entity → 

2) Attribute → 



27/1/20

Key constraints.

Types of key

① Super key
② Candidate key
③ Primary key
④ Composite key
⑤ Foreign key

Relation (Table)

st_info

D.O.B.	st_rollno	Name
1	251	~~X~~
2	252	Y
3	235	X
4	256	M.

* Course.

Course_id	Course name	St_id
101	a	1
102	b	2
103	c	1
104	d	1

Super key
{st_id, st_rollno}
{st_id}{st_idrollno}
{st_id, Name}

* Super key :- Set of one or more column to identify a row uniquely in a table

* Candidate key :- All super key without redundancy

* Primary key :- One from candidat key with least attributes.

28/1/20

① drop column
② add column etc.

SQL Query :-

① DDL :- data defination language
- ↳ create
- ↳ delete
- ↳ Alter (modifications in table)

② DML :- data manipulation language
- ↳ Insert
- ↳ Update
- ↳ select

Data types :-

① int
② char
③ varchar(n)
④ Float(n)
⑤ Smallest int
⑥ Date
⑦ Binary

Constraints :-

① Unique
② ~~NOT NULL~~ NOTNULL
③ check
④ key
⑤ ~~Au~~ Autoincrement
etc.

Basic Scheme

Create database database name;
Create table table name;
Alter table

DDL :- It is not case sensitive.

Create table tablename (a1, a2, ... an)

Create table st_info (st_name varchar(20), st_rollno char(10), st_mobile char(10), st_dob char(20), st_address varchar(100), st_id int NOTNULL);
or Unique not null

DML :-

① Insert :-
Insert into tablename (a1, a2 ... an) values (v1 ... vn);
Insert into (st_name, st_dob) values ('Rashi', '30/9/...' yyyy-mm-DD

② Select :-
select * from tablename where condition.

29/1/20

Relational Algebra:-

Query:-

Produce instance of relations as a output.
It uses operators to perform query
a operator can be unary and binary
Fundamental operations for relational algebra.

① Select (σ):-
Notation $\sigma_p(r)$

In select operation, select the tuples (values) that satisfy given predicate (conditions)

σ_p stands for the selection predicates
r stands for the relation (variable)

$\sigma_{subject = \text{"database"}}(Books)$
select values from Books where subject is database.

Condition :- Books with more than 4000 Rs

$\sigma_{subject = \text{"database"} \text{ and } price > \text{"4000"}}(Books)$

price > 230
price < 400

② Project operation (Π)

$\Pi_{A_1, A_2, A_3 \ldots A_n}(r)$

project the column that satisfy the given predicates where A_1, A_2, A_3 all are the attribute and r is the relation.

In project operation duplicates rows are automatically eliminated.

$\Pi_{subject, price}(Books) \rightarrow 230 < price > 400$"

$\Pi_{subject}(\sigma_{price > 230 \text{ and } price < 400}(Books))$

③ Union operation

$r \cup s = \{t / t \in r \text{ or } t \in s\}$

notation $r \cup s$ where r, s are relations.

r & s have same no. of attribute & compatible.
Duplicate tuple are automatically eliminated.

$\Pi_{author}(Books) \cup \Pi_{author}(articles)$

① r - s
② r ∩ s
③ r ∪ s

29/1/20

Assignment - I

① Create database for patient information system with following operation
1) Patient registration
2) Fees structure
3) Patient treatment
4) Patient Billing

② Execute following queries:- constraints:-
i) No null value insert in Patient name, DOB, fees.
ii) Define primary key in every table and each primary key value is not allowed user to insert value.

③ Perform query operation:-
a) insert query execute for every table
b) update query for single attribute in every table
c) select query - display patient name who has are the govt. servants and also categorised the department of patients.
Display the name of patient with DOB - those who are senior citizens and who are govt servants, servants and state govt servants, or they are regular patients.

31/2/20

Differ. commands and their syntax.

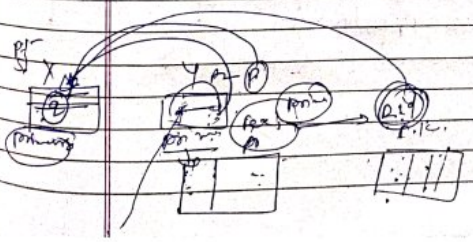
① alter table tablename add
modify
drop

② insert into tablename (columnname, ...)
values (..), (..), (..)

③ auto-increment = 100

④ update table tablename column = 'v1' where cond".

⑤ select c1, c2, From tablename where condⁿ →



05/2/2020

Q1- Create a database of for inventory management of for particular organisation which has several departments and stores with no. of departments and every department has have no of products and every department have no of employee with different designation.

Each store have facility to sale product, purchase product and return product.

Then perform following operations:-

i) Insert values

ii) Count no of employees who has same designation

Q2- Display name of product which belongs to a particular department and store.

Q3- Find out the stock of a particular product at particular store.

Q4- Find out the selling quantity of a particular product

Q5 - Find out the average price of products for a particular department.

Soln Create database inventory;

Tables :-

1) Store Table.

store id	store. Inquiry.	Location

2) Department table.

Dep_id	Depart.name	Store_id

3) product table

F.key

P_id	Product-ction	Qty	dep id	store id

4) Employe table.

Emp_id	Emp_name	Designat	Store_id

10/2/20

JOIN

- Inner join
- Outer join
  - left join
  - right join
  - Full join

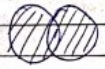
Inner join

Outer join.

left join.

right join.

Full join



Customer info (t1)

C-id	C-name	mobile no

order (t2)

O-id	C-id	Odr-rate

Select t1.a1, t1.a2, t2.a1 From t1 join t2 ON ~~cond^~~ condn.

① Select t1.a1, t1.a2, t2.a1 from t1 inner join t2 ON t1.cid = t2.cid

② Select t1.cid, t1.cname, t2.oid from t1 left join t2 ON t1.cid = t2.cid.

Diagram :-

① inner join — no null values.

O_id	C_id	C_name
1	1	XY
2	1	XY
3	1	XY

② left join — no. null values in left side.

C-id.	C-name	O-id.
1	XY	1
2	YZ	NULL

③ Right join — no. null values in right side.

O-id	C-name.	C-id.
1	XY	1
NULL	YZ	2

inner — equijoint, θ joint

11/2/20

Q1- Create all tables with key constraints then perform select operation following :-
a) List the faculty with dept name & sub-name
b) List the sub-name with dept-name
c) List all student with dept-id.

① Create table dept (dept_id, ~~dept_no~~ int not null auto_increment, dept_name varchar(30), not null, primary key (dept_id));

② Create table st_info (st_id int not null auto_increment, st_name varchar(30), not null, batch char(5), primary key (st_id), dept_id int not null, foreign key (dept_id) references dept (dept_id));

③ Create table fc_info (fc_id int not null auto_increment, ~~st_n~~ dept_id int not null, fc_name varchar(30), sub_id int not null, primary key (fc_id), foreign key (dept_id) references dept (dept_id), foreign key (sub_id) reference, sub (Sub_id));

③ Create table sub (Sub_id int not null auto_increment, Sub_name varchar(30), dept_id int not null, primary key (sub_id), foreign key (dept_id) references dept (dept_id));

① Insert into dept (dept_name) values ('CSE') ('EE') ('EEE') ('ECE') ('CE') ('ME') ('MBA');

② Select * from dept;

select dept_name from dept;
Select dept_name from dept where dept_id = 1;

③ Update dept ~~set~~ table set dept_name = 'CSE' where dept_id = '2';

12/12/20

Trigger

Create trigger triggername
[Before/After] [INSERT/Update/Delete] OF
(Column name) for each row
{ Begin trigger body }
End.

S.id	Sub-1	Sub-2	Sub-2	Total	%
1	20	~~20~~	20	60	100
2	17	18	16	51	~~85~~ 85.0

S.total = S.Sub-1 + S.Sub-2 + S.Sub-3 ;

S% = S.total * 100
60.

End.

12/12/20

Chapter-3

① Nested SQL Query.

Relation:-
(St_id, st_name, st_marks, st_department)
(dept_id, dept_name)

Ques List the toppers of all department.
→ Dept name
→ name
→ marks

S:-

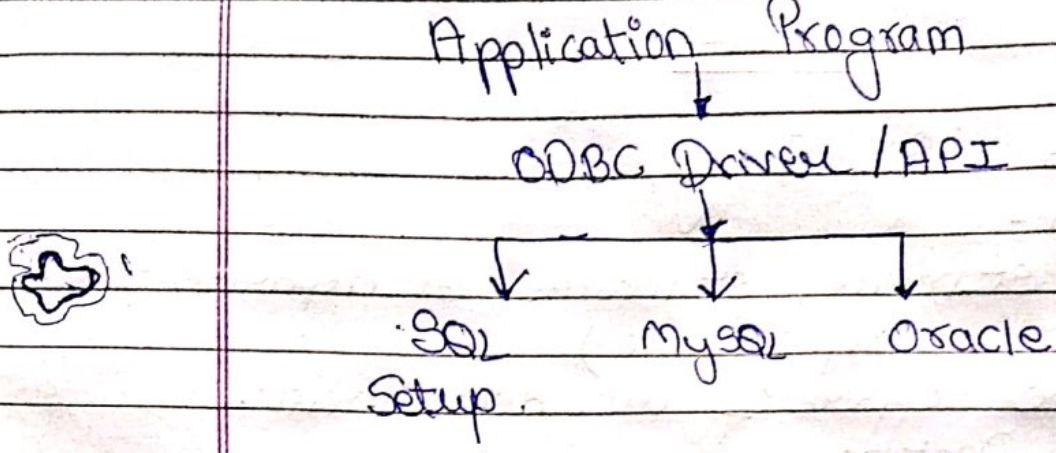
St_id	St_name	St_marks	dept_id

D:-

dept_id	dept_name

Select d.dept_name, (select S.s_name, S_marks from student where S_marks = max [S_name])
from d inner join S on d dept_id = S. dept_id;

A/b/20



ODBC driver ~~provide~~ is an API to connect and execute the SQL query with the database.

ODBC :- Open database connectivity

ODBC handle SQL request and convert it into a request that is in proper format for ~~indiv~~ individual database so database can understand the request.

ODBC can use for any programming lang.