

## Assignment - 1

Dhananjay Arwal

EN19CS30110

1 Explain Architecture in DBMS 2 tier, 3 tier, general?

Ans The design of a DBMS depends on its architecture.

It can be centralized or decentralized or hierarchical.

The architecture of a DBMS can be seen as either single tier or multi-tier. An n-tier architecture divides the whole system into related but independent n modules, which can be independently modified, altered, change or replaced.

1-Tier: In this architecture, the DBMS is the only entity where the user directly sits on the DBMS & uses it.

Any changes done here will directly be done on the DBMS itself. It does not provide handy tools for end-users.

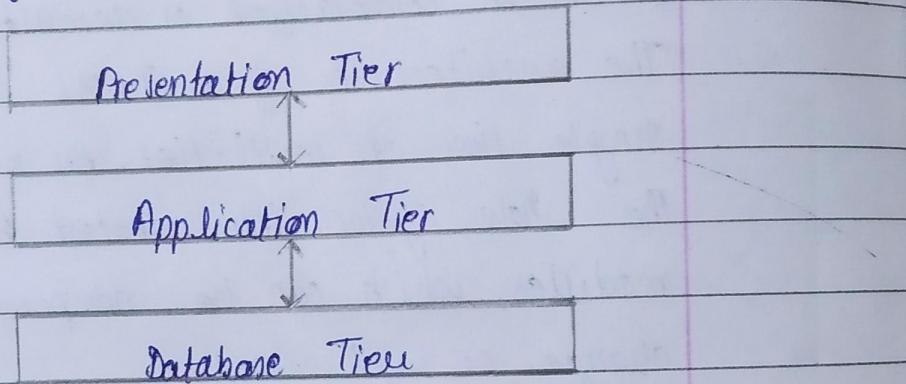
Database designers & programmers normally prefer to use single tier architecture.

2-Tier: If the architecture of DBMS is 2 tier, then

it must have an application through which the DBMS can be accessed. Programmers use 2-tier architecture where they access the DBMS by means of a application

Here the application tier is entirely independent of the database in terms of operation, design & programming.

3-Tier: It separates its tiers from each other based on the complexity of the users & how they use the data present in the database. It is the most widely used architecture to design a DBMS.



- Database tier - The database resides along with its query processing languages. We also have the relations that defines the data & their constraints at this level.
- Application (middle) tier - At this tier reside the application servers & the programs that access the database for a user, this application tier presents an abstracted view of the database. End-user are unaware of any

existence of the database beyond the application. At the other end, the database tier is not aware of any other user beyond the application tier.

- Presentation (User) tier - end user operate on this tier
  - o they know nothing about any existence of the database beyond this layer. At this layer, multiple views of the database can be provided by the application. All views are generated by application that reside in the application tier.

Multiple-tier database architecture is highly modifiable, as almost all its components are independent o can be changed independently.

2 Explain diff data models of DBMS

Ams A database model shows the logical structure of a database including the relationships o constraints that determine how data can be stored o accessed.

Types of database models.

a) Relational Model

It represents the database as a collection of relations.

A relation is nothing but a table of values. Every row in the table represents a collection of selected

data values. These stored in the table denote a real-world entity or relationship. for ex-

Student ID	First name	Last name
52 - 743965	Ram	Gupta
48 - 209689	Shyam	Porwal
14 - 204968	Hari	Mishra

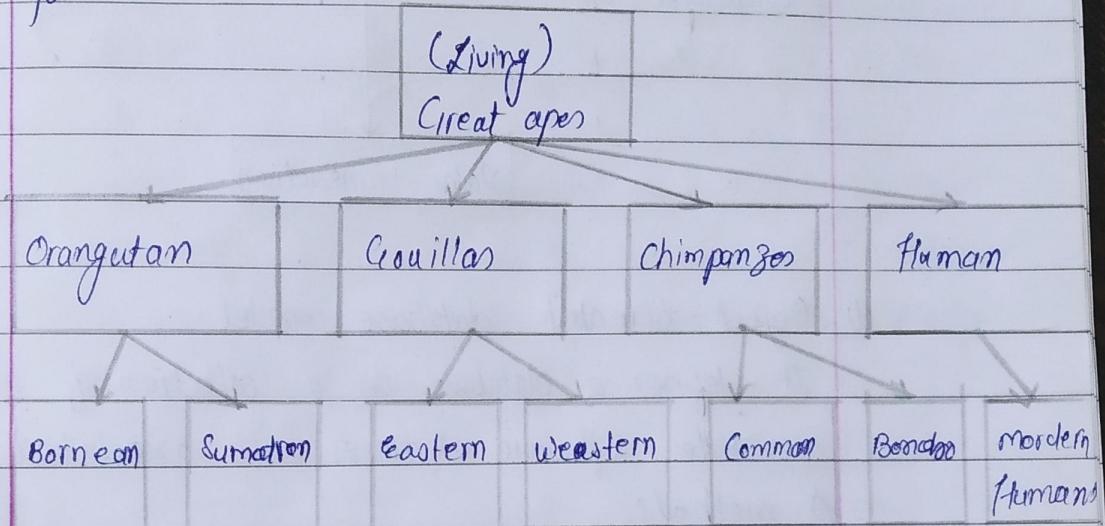
Provider ID	Provider Name
156 - 983	United Health
146 - 823	Blue Shield
447 - 784	Carefirst Inc.

Student ID	Provider ID	Type of plan	Start date
52 - 743965	156 - 983	HCA	04/02/2014
48 - 209689	146 - 823	HMO	12/01/2014
14 - 204968	447 - 784	HSA	21/04/2014

### b) Hierarchical model

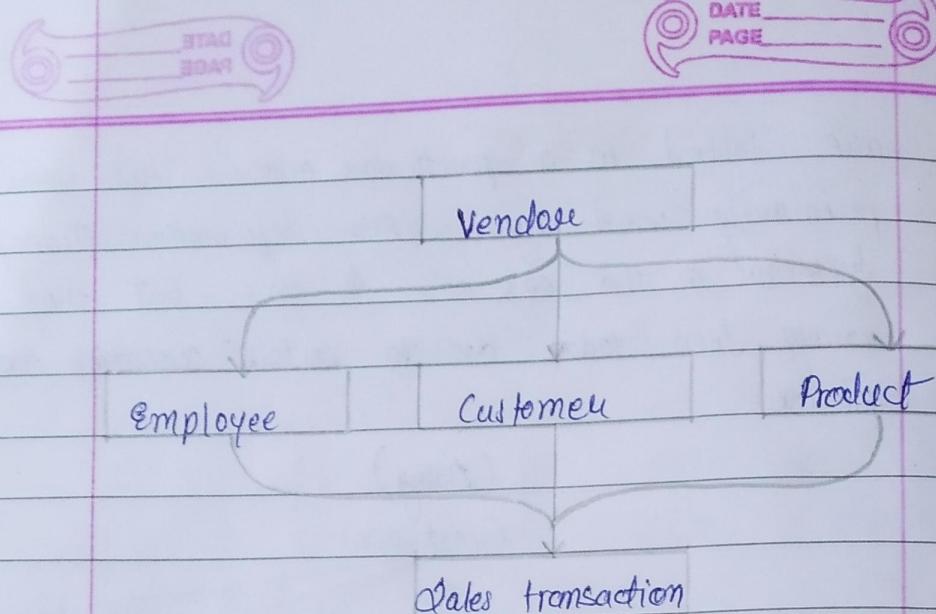
Organizes data into a tree-like st., where each record has a single parent or root. sibling records

are sorted in a particular order. This model was primarily used by IBM's Information Management Systems in the 60's & 70's, but they are scarcely seen today due to certain operation inefficiencies. for ex -



### c) Network model

This model builds on the hierarchical model by allowing many-to-many relationships b/w linked records, implying multiple parent records. Based on mathematical set theory the model is constructed with sets of related records. A record can be a member or child in multiple sets, allowing this model to convey complex relationships. for ex -



#### d) Object-oriented database model

It defines a database as a collection of objects, reusable software elements, with associated features (methods).

A hypertext database allows any object to link to any other object. It's useful for organizing lots of disparate data, but it's not ideal for numerical analysis.

object 1: Sales report

Month

object 2: Instance

01-15-14

→ Product code

54

Vendore

object 2: Sale activity 154-234

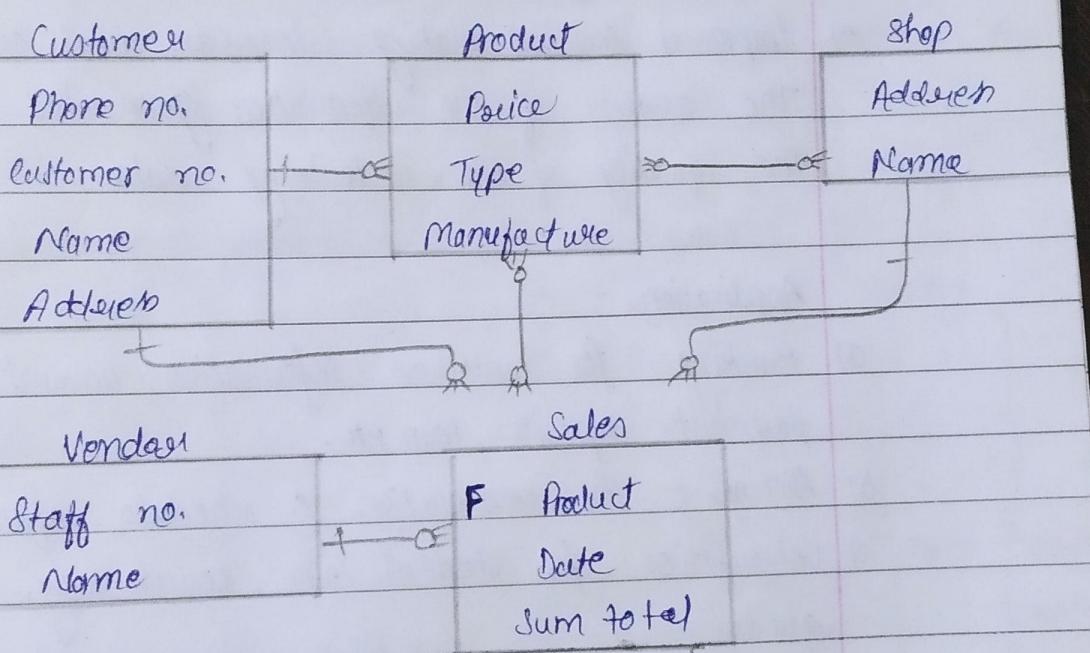
Customer

Product code

Price

### e) Entity - relationship model

This model captures the relationships b/w real world entities much like the network model, but it isn't as directly tied to the physical structure of the database. Instead, it's often used for designing a database conceptually. for ex -



### 3 Explain advantages & applications of DBMS?

Ans Advantages -

- Improved data sharing - Such access makes it possible for end users to respond quickly to changes in their environment.

- 6 DATE PAGE 9
- DATE PAGE
- b) Improved data security - The more users access the data, the greater the risks of data security breaches.
  - c) minimized data inconsistency - The probability of data inconsistency is greatly reduced in a properly designed database.
  - d) Improved data access - The DBMS makes it possible to produce quick answers to ad hoc queries.
  - e) Improved decision making - Increase end-user productivity. The quality of the information generated depends on the quality of the underlying data.

### Applications -

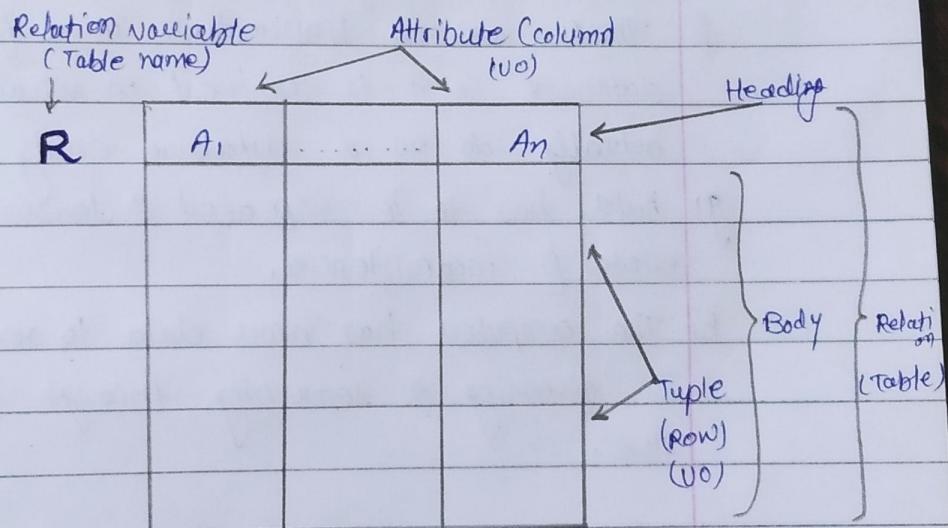
- a) Banking - for customer information, account activities, payments, deposits, loan etc.
- b) Airlines - for reservation & schedule information.
- c) Universities - for student info., course reg., colleges & grades
- d) Sales - for storing customer, product & sales info.
- e) HR management - for info. about employees, salaries, payrolls, deduction, generation of paychecks, etc

4 Explain terminology of relation data model?

Ans The relational model for database management

is an approach to managing data using a structured language consistent with first order predicate logic, first described in 1969 by English computer scientist Edgar F. Codd, where all data is represented in terms of tuples, grouped into relations.

The relational model's central idea is to describe a database as a collection of predicates over a finite set of predicate variables, describing constraints on the possible values & combination of values.



A relation consists of a heading & a body. A heading is a set of attributes. A body is a set of  $n$ -tuples. The heading of the relation is also the heading of

each of its tuples. A table is an accepted visual representation of a relation; a tuple is similar to the concept of a record.

### 5 Draw-backs of flat file system?

- Ans a) file sizes can be large due to repeated data.
- b) flat file database is harder to update.
- c) It is poor database in terms of complex queries.
- d) It increases redundancy  $\&$  inconsistency.
- e) Harder to change data format.
- f) There are no transactions in a flat file database, so it is limited in what it can actually do as a database entity.
- g) Data has to be repeated  $\&$  leads to inputting errors  $\&$  inconsistencies.
- h) The computer has more data to read through, so accessing  $\&$  searching through them can be slow.