



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

CAT – I

CSE 2004 – Database Management Systems

Course Name: B.Tech

Slot: D2

Max. Marks: 50

Answer ALL Questions (5 * 10 = 50)

1 a. Advantages of Database Management System

- ❑ Controlling Redundancy
- ❑ Restricting Un-authorized Access
- ❑ Providing persistent storage for pgm objects and data structures
- ❑ Providing multiple user interfaces
- ❑ Representing complex relationships among data
- ❑ Enforcing integrity constraints
- ❑ Providing Backup and recovery

SEARCH VIT QUESTION PAPERS
ON TELEGRAM YO JOIN



1 b. Three levels of data abstraction with a neat diagram

There are three levels of data abstraction which are discussed below.

• Physical Level or Internal Level:

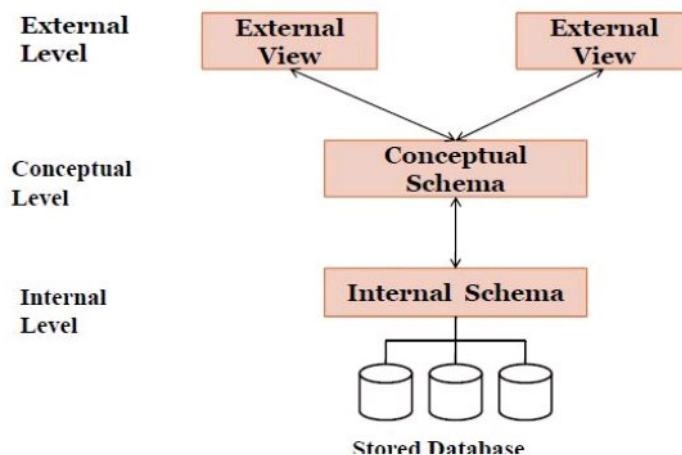
It is the lowest level of abstraction and describes how the data in the database are actually stored. This level describes complex low-level data structures in detail and is concerned with the way the data is physically stored. Data only exists at physical level.

• Logical Level or Conceptual Level:

This is the next higher level of abstraction and describes what data are stored in the database, and what relationships exist among those data. It describes the structure of whole database and hides details of physical storage structure. It concentrates on describing entities, data types, relationships, attributes and constraints. All of the views must be derivable from this conceptual schema.

• View Level or External Level:

It is the highest level of abstraction and is concerned with the way the data is seen by individual users. This level simplifies the users' interaction with the system. It includes a number of user views and hence is guided by the end user requirement. It describes only those part of the database in which the users are interested and hides rest of all from those users. Each user group refers to its own external schema.



2 a. Mapping between schema levels

- **Conceptual (high-level, semantic) data models:**
 - ▢ Provide concepts that are close to the way many users perceive data.
 - (Also called *entity-based* or *object-based* data models.)
- **Physical (low-level, internal) data models:**
 - ▢ Provide concepts that describe details of how data is stored in the computer. These are usually specified in an ad-hoc manner through DBMS design and administration manuals
- **Implementation (representational) data models:**
 - ▢ Provide concepts that fall between the above two, used by many commercial DBMS implementations (e.g. relational data models used in many commercial systems).

2 b. Functionalities

Stored Data Manager

- The database and the database catalogue are stored on disk
- Access to the disk is handled by the Operating System.
- A higher-level stored data manager controls access to DBMS information that is stored on disk, whether part of the database or the catalogue.
- The stored data manager may use basic OS services for carrying out low-level data transfer, such as handling buffers.
- Once data is in buffers, the other DBMS modules, as well as other application programs can process it.

DDL Compiler

- Processes the schema definitions and stores the descriptions (meta-data) in the catalogue.

Runtime Database Processor

- Handles database access at runtime.
- Received retrieval or update operations and carries them out on the database.
- Access to the disk goes through the stored data manager.

Query Compiler

- Handles high-level queries entered interactively.
- Parses, analyzes and interprets a query, then generates calls to the runtime processor for execution.

Precompiler

- Extracts DML commands from an application program written in a host language.
 - Commands are sent to DML compiler for compilation into code for database access. The rest is sent to the host language compiler.
1. **DML Compiler:** It translates the DML statements in a query language into an evaluation plan consisting of low-level instructions that the query evaluation understands. It also performs query optimization which actually picks up the lowest cost evaluation plan from various alternatives.
 2. **Query Evaluation Engine:** It executes the low level instruction compiled by the DML compiler.

3a. ER to relational Model

Company(CompanyID , name , address)
Staff(StaffID , dob , address , WifeID)
Child(ChildID , name , StaffID)
Wife (WifeID , name)
Phone(PhoneID , phoneNumber , StaffID)
Task (TaskID , description)
Work(WorkID , CompanyID , StaffID , since)
Perform(PerformID , StaffID , TaskID)

3b.Option 3

Explanation: Only (*cid*) and (*adhaar*) are candidate keys. Any superset of those are superkeys and any ONE of those can be chosen as primary key.

4.a Strong Vs weak Entity

Basis for Comparison	Strong Entity	Weak Entity
Basic	The Strong entity has a primary key.	The weak entity has a partial discriminator key.
Depends	The Strong entity is independent of any other entity in a schema.	Weak entity depends on the strong entity for its existence.
Denoted	Strong entity is denoted by a single rectangle.	Weak entity is denoted with the double rectangle.
Relation	The relation between two strong entities is denoted by a single diamond simply called relationship.	The relationship between a weak and a strong entity is denoted by Identifying Relationship denoted with double diamond.
Participation	Strong entity may or may not have total participation in the relationship.	Weak entity always has total participation in the identifying relationship shown by double line.

b i. Composite attribute and complex attribute.

A composite attribute consists of a group of values from more than one domain. For example, the Address attribute consists of several domains such as house number, street number, city, country, etc.

A complex attribute that is both composite and multi valued. Multi Valued Attribute: Attribute that hold multiple values.

Example1: A customer can have multiple phone numbers, email id's etc

Example2: A person may have several college degrees

ii. Entity integrity constraint and domain constraint.

Data Integrity

Data Integrity validates the data before getting stored in the columns of the table.

SQL Server supports four type of data integrity:

Entity Integrity

Entity Integrity can be enforced through indexes, UNIQUE constraints and PRIMARY KEY constraints.

Domain Integrity

Domain integrity validates data for a column of the table.

5. Types of Keys

Types of Keys

Primary Key – A primary is a column or set of columns in a table that uniquely identifies tuples (rows) in that table.

Super Key – A super key is a set of one or more columns (attributes) to uniquely identify rows in a table.

Candidate Key – A super key with no redundant attribute is known as candidate key

Alternate Key – Out of all candidate keys, only one gets selected as primary key, remaining keys are known as alternate or secondary keys.

Composite Key – A key that consists of more than one attribute to uniquely identify rows (also known as records & tuples) in a table is called composite key.

Foreign Key – Foreign keys are the columns of a table that points to the primary key of another table. They act as a cross-reference between tables.

<https://www.geeksforgeeks.org/dbms-keys-candidate-super-primary-alternate-and-foreign/>