DBMS Question Bank

# Q: Explain decomposition using multivalued dependencies.

🔑 Keywords: Decomposition, Multivalued Dependency (MVD)

📝 Meaning: How can we break down a table using multivalued dependencies to remove data redundancy?

* ✅ Answer:
* Multivalued Dependency (MVD) means one attribute determines a set of values independently of another attribute.
* If a table has MVDs, it can lead to unnecessary data repetition.
* Decomposition breaks the table into smaller relations
* Each new relation contains only the related attributes, reducing redundancy.
* Original relation can be recreated using natural joins.Example Inner, outer, left and light join
* This process ensures data consistency and avoids anomalies.

# Q: Explain in detail the design issues in the E-R model. How do extended E-R features help overcome some of these issues?

🔑 Keywords: Design issues, E-R model, Extended E-R features

 Design issues are the challenges or problems that arise during the process of creating and structuring a database.

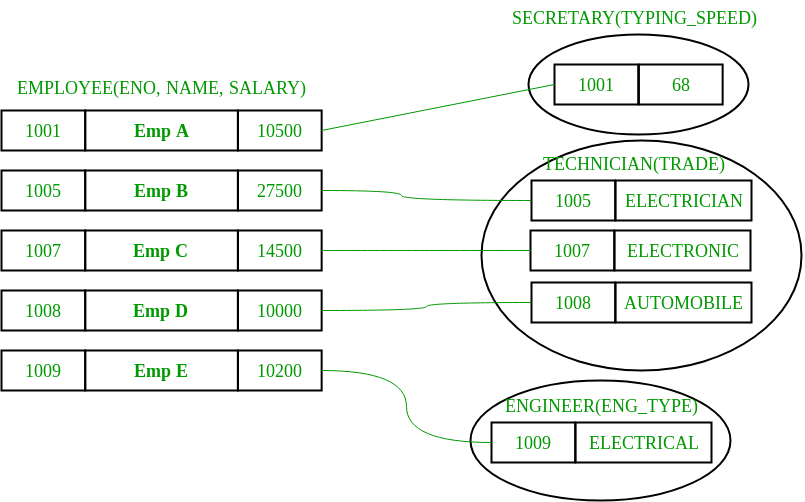
The Entity-Relationship (ER) model and Extended Entity-Relationship (EER) model are both conceptual data models used in database design.

The ER model provides a basic way to represent entities and their relationships, while the

EER model extends the ER model with more features to handle more complex data structures.

📝 Meaning: What are the problems when designing ER models, and how do advanced features help solve them?

* Answer: Design issues in database design are:
* Choosing between attributes vs entity sets can be confusing.
* Deciding the type of relationship may be unclear.
* Identifying correct primary keys is a challenge.
* Handling inheritance or classification is not always easy.
* Extended E-R features help:
* -Generalization helps group similar entities (e.g., Student and Teacher into Person).
* Specialization allows splitting a general entity into specific ones (e.g., Person into Student and Teacher).
* Aggregation models relationships between relationships.
* Categorization helps when a subclass belongs to more than one superclass.
* - These improve clarity and support complex requirements.

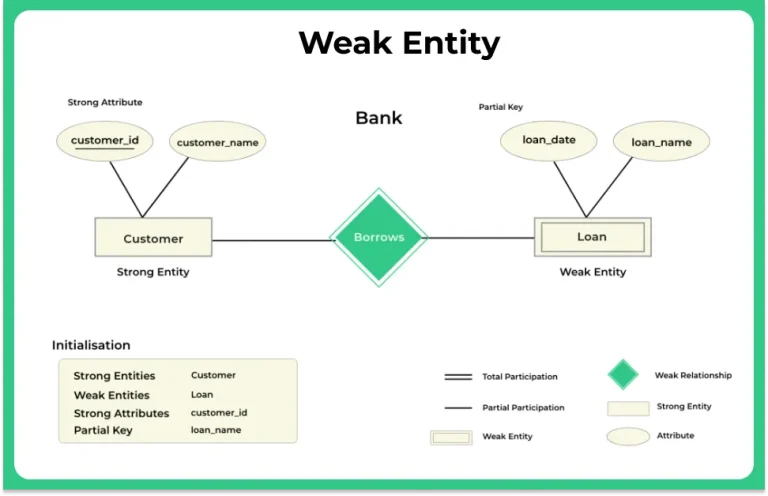


# Q: Describe weak entity sets with example and their representation in E-R diagrams.

🔑 Keywords: Weak entity, E-R diagram, Example

📝 Meaning: What are weak entities, how are they shown in ER diagrams, and give a simple example.

* ✅ Answer:
* A weak entity cannot be uniquely identified by its attributes alone.
* It depends on a strong entity for its identification.
* It has a partial key and a total participation in the identifying relationship.
* In ER diagrams:
* Weak entities are shown using double rectangles.
* Their identifying relationship is shown using a double diamond.
* Partial keys are underlined with dashed lines.
* Example: 'Dependent' is a weak entity of 'Employee'. Each dependent depends on the employee for unique identification.



NENDA DEEP: <https://prepinsta.com/dbms/weak-entity-and-strong-entity/>

# Q: Explain specialization and generalization with diagrams and examples. How are they useful in database modeling?

🔑 Keywords: Specialization, Generalization, Database modeling

📝 Meaning: What do we mean by generalization and specialization in databases, and how do they help?

* Answer:
* Generalization combines similar entities into one broad entity (e.g., Car and Truck into Vehicle).
* Specialization splits one broad entity into more specific ones (e.g., Employee into Manager and Engineer).
* These concepts support inheritance in data design.
* They simplify complex data structures.
* They help avoid redundancy by sharing common attributes in one place.

In diagrams:

* - A triangle labeled with ISA is used to show generalization/specialization.
* They improve clarity in ER modeling by showing real-world hierarchy.

# Q: Describe the features of a good relational database design.

🔑 Keywords: Relational database, Design features

📝 Meaning: What are the key features that make a relational database design effective and reliable?

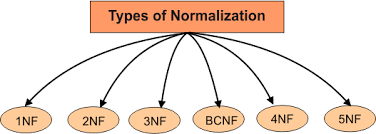
A relational database is a system that organizes and stores data in a structured manner, using tables (also called relations) to represent relationships between different data points.

* Answer:
* Minimal redundancy: No unnecessary repetition of data.
* Data integrity: Rules are enforced to maintain accuracy and consistency.
* Normalization: Tables are properly normalized to reduce anomalies.
* Efficient access: Queries can run fast with indexes and optimized schema.
* Clear primary and foreign keys: Ensures proper relationships between tables.
* Scalability: Easy to modify or expand when needed.
* Security: Data access is properly controlled.

# Q: What is normalization? Explain all its types with suitable examples.

🔑 Keywords: Normalization, 1NF, 2NF, 3NF, BCNF, 4NF, 5NF

📝 Meaning: What is normalization and how do we improve table design step by step?

* ✅ Answer:
* Normalization is the process of organizing data to reduce redundancy and improve data integrity.
* Types of Normal Forms:
* 1NF: Remove repeating groups. Each cell should have atomic values.
* 2NF: Remove partial dependencies. Every non-key attribute should depend on the whole primary key. <https://www.datacamp.com/tutorial/second-normal-form>
* 3NF: Remove transitive dependencies. Non-key attributes should depend only on the primary key. <https://www.datacamp.com/tutorial/third-normal-form>
* BCNF: A stronger version of 3NF. Every determinant must be a candidate key.
* 4NF: Remove multivalued dependencies.
* 5NF: Remove join dependencies.
* Example: Splitting a table with multiple phone numbers into separate tables with proper keys.
* 

NENDA DEE : <https://takeuforward.org/dbms/normalisation-and-its-types>

# Q: Discuss functional dependency theory and its role in relational design.

🔑 Keywords: Functional Dependency, Relational Design

📝 Meaning: What is functional dependency and how does it help in designing relational databases?

* ✅ Answer:
* Functional dependency (FD) means one attribute uniquely determines another (A → B).
* Functional dependency (FD) describes a relationship between two or more attributes (columns) within a table. It means that one attribute or a set of attributes uniquely determines the value of another attribute or set of attributes. If you know the value of the "determinant" (the attribute that determines), you can uniquely determine the value of the "dependent" (the attribute being determined).
* FDs help in identifying candidate keys and ensuring data consistency.
* They are used to detect anomalies and suggest better table structure.
* FDs are key to applying normalization rules (2NF, 3NF, BCNF).
* Helps in decomposition of relations while preserving data.
* Makes the design less redundant and more structured.

# Q: Explain decomposition using functional dependencies. How does it help in achieving normalization?

🔑 Keywords: Decomposition, Functional Dependency, Normalization

📝 Meaning: How do we use functional dependencies to split tables and make them more efficient?

* ✅ Answer:
* Decomposition breaks a relation into smaller relations using FDs.
* This helps remove partial, transitive, or multivalued dependencies.
* Ensures each relation is in a higher normal form like 3NF or BCNF.
* Reduces data anomalies and improves consistency.
* Preserves data without loss (lossless decomposition).
* Example: A relation with StudentID → Name, Course split into Student(StudentID, Name) and Enrollment(StudentID, Course).

# Q: What is multivalued dependency? How do we use it for decomposition? Give an example.

🔑 Keywords: Multivalued Dependency, Decomposition, Example

📝 Meaning: What is a multivalued dependency and how do we handle it in database design?

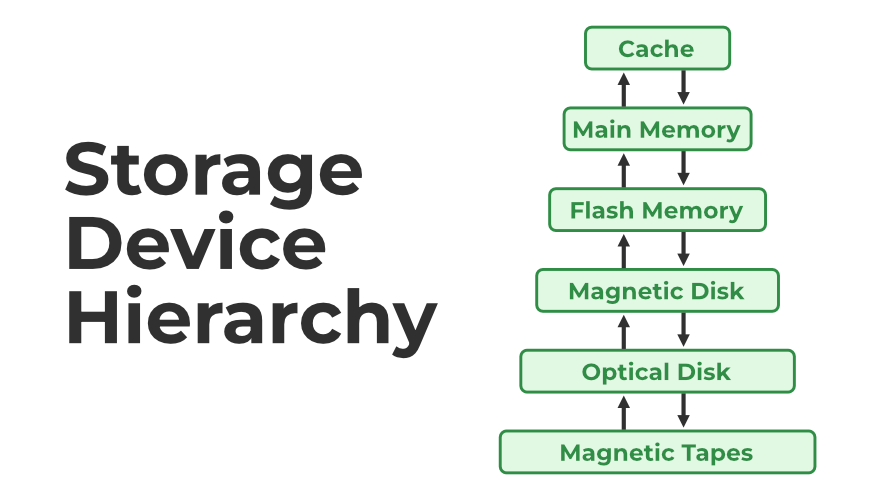
* ✅ Answer:
* MVD exists when one attribute determines multiple values of another attribute independently of other attributes.
* Occurs when a table has two or more independent multi-valued facts about an entity.
* Decomposition breaks the table into two to avoid repetition.
* Used to achieve 4NF (Fourth Normal Form).
* Preserves all data and maintains dependencies.
* Example: Student with multiple phone numbers and multiple hobbies → create two tables: StudentPhones and StudentHobbies.

# Q: Explain the different types of physical storage media used in databases. How do they vary in terms of speed, cost, and capacity?

🔑 Keywords: Physical Storage Media, Speed, Cost, Capacity

📝 Meaning: What are the main types of hardware used for storing databases and how do they compare?

* Answer:
* Physical storage media refers to the tangible materials used to store data, like hard drives, solid-state drives, or optical discs
* Cache: Very fast, very expensive, small size.
* Main Memory (RAM): Fast, costly, volatile (data lost on power off).
* Solid State Drives (SSD): Fast access, more expensive than HDDs, no moving parts.
* Hard Disk Drives (HDD): Cheaper, slower than SSD, large storage.
* Optical Discs (CD/DVD): Cheap, slower, limited storage, mostly for backup.
* Magnetic Tapes: Very cheap, very slow, used for long-term backups.



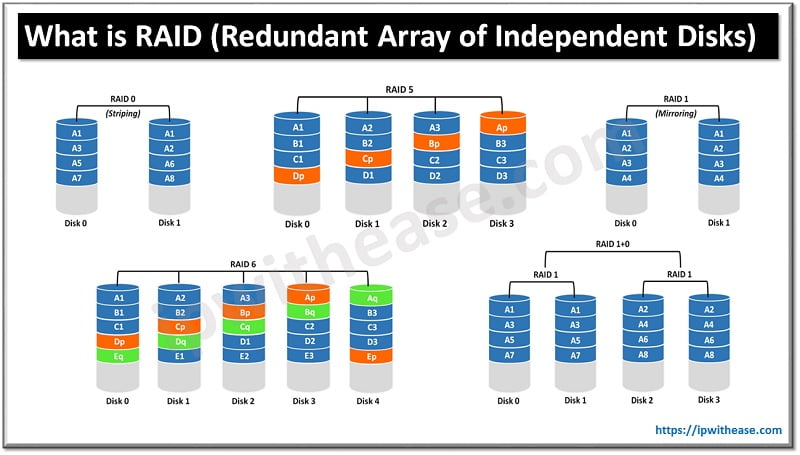
# Q: Describe the various RAID levels and their advantages and disadvantages.

🔑 Keywords: RAID, Redundancy, Performance

📝 Meaning: What are the different RAID methods for storing data and what are their pros and cons?

RAID (redundant array of independent disks) is a way of storing the same data in different places on multiple hard disks or solid-state drives (SSDs) to protect data in the case of a drive failure

* ✅ Answer:
* RAID 0: Stripes data, improves speed, no redundancy.
* RAID 1: Mirrors data, good redundancy, uses more space.
* RAID 5: Data + parity, balances speed and fault tolerance.
* RAID 6: Double parity, survives 2 disk failures, slower writes.
* RAID 10: Combines RAID 1 and 0, high performance and redundancy, costly.
* Trade-off: Higher RAID level means better fault tolerance but more cost and complexity.



# Q: Discuss the different types of file organization in databases. What are the advantages and disadvantages of each?

🔑 Keywords: File Organization, Sequential, Indexed, Hashed

📝 meaning: How is data physically stored in files and what are the pros and cons of each method?

File Organization refers to the the way in which data is stored in a file and the method(s) by which it can be accessed.

* Answer:
* Sequential: Data stored in sorted order. Easy for batch processing, slow for random access.
* Heap (Unordered): Fast insertion, slow searching.
* Hashed: Fast for exact searches, not good for range queries.
* Clustered: Related records stored close, fast access for groups.
* Indexed: Uses indexes for faster searches, needs extra space.
* Each has use cases depending on access pattern and performance goals.