Day15

Database

DBeaver installation

a. DBeaver is a SQL client software application and a database administration tool. For relational databases it uses the JDBC application programming interface to interact with databases via a JDBC driver. For other databases it uses proprietary database drivers.
b. Since DBeaver was already present in my device, so I moved to another section.

Data vs Information

Data

a. Data represents raw elements or unprocessed facts, including numbers and symbols to text and images.
b. When collected and observed without interpretation, these elements remain just data—simple and unorganized.
c. When these pieces are analyzed and contextualized, they transform into something more meaningful.
d. Types of data are :
    i. Quantitative: Quantitative data refers to numerical information like weight, height, etc.
    ii. Qualitative: Qualitative data refers to non-numeric information like opinions, perceptions, etc.

Information

a. It is structured, organized, and processed data, presented within a context that makes it relevant and useful to the person who needs it.

Differences between data and information

Bits and Bytes are the measuring unit of data.	Information is measured in meaningful units like time, quantity, etc.
As tabular data, graphs, and data trees can be easily structured.	Information can also be structured as language, ideas, and thoughts.
Data does not have any specific purpose	Information carries a meaning that has been assigned by interpreting data.
It is low-level knowledge.	It is the second level of knowledge.
Data does not directly help in decision making.	Information directly helps in decision making.
Data is a collection of facts, which itself has no meaning.	Information puts those facts into context.
Example of data is student test scores.	Example of information is average score of class that is derived from given data.

a. Data is text and numeral values, whereas information is refined form of actual data.
b. Temperature readings like 72°F, 68°F, 75°F are data, but " Based on the temperature reading a weather report can be generated." is information.
c. Student grades like 85, 82, 35, etc are data, but "Based on the student grades, the average grade of class can be derived." is information
d. In each case, the raw data becomes meaningful information after being processed, analyzed, and presented in a relevant context.

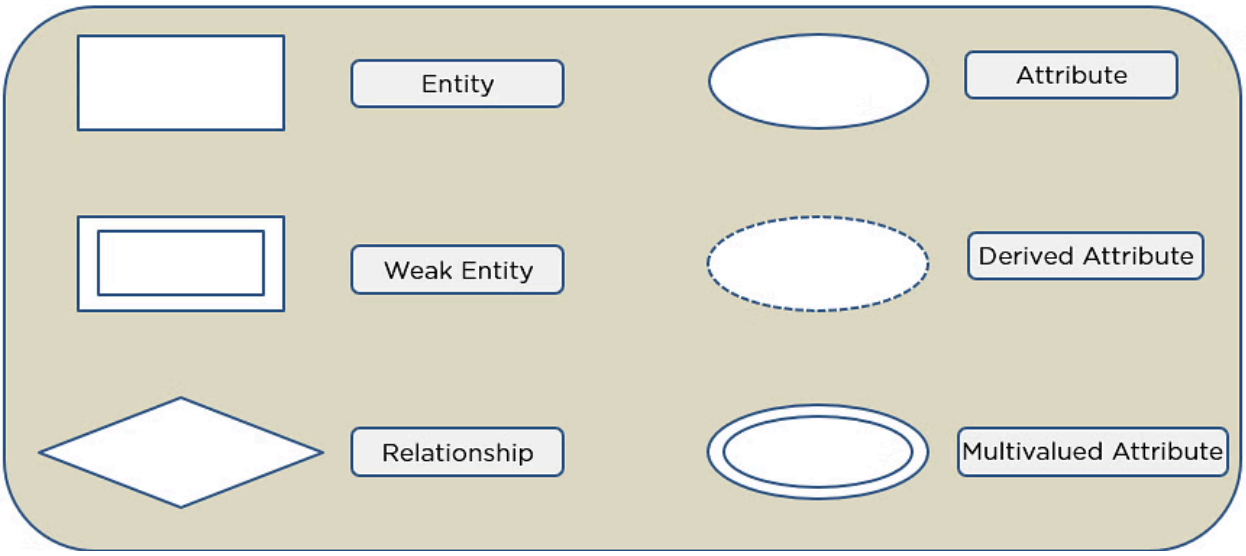
Data vs Information vs Knowledge vs Wisdom

Data is collection of facts, which itself has no meaning while Information puts those facts into context and knowledge can be defined as what we know while wisdom is Utilizing information to guide judgements and adjust to changing circumstances.

Data Modelling

a. process of creating a simplified visual diagram of a software system and the data elements it contains, using text and symbols to represent the data and how it flows.
b. provide a blueprint to businesses for designing a new database or reengineering a legacy application.
c. Mainly three levels of abstraction:
    i. Conceptual level:
        1. The conceptual level involves defining the high-level entities and relationships in the data model, often using diagrams or other visual representations.
        2. Something like ER Modelling
    ii. Logical level:
        1. The logical level involves defining the relationships and constraints between the data objects in more detail, often using data modeling languages such as SQL or ER diagrams.
        2. Something like ER Diagram
    iii. Physical level:
        1. The physical level involves defining the specific details of how the data will be stored, including data types, indexes, and other technical details.

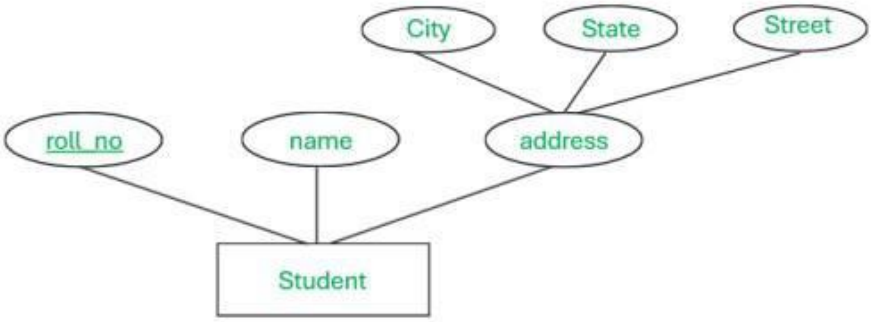
ER Diagram

a. diagram that represents relationships among entities in a database.
b. 

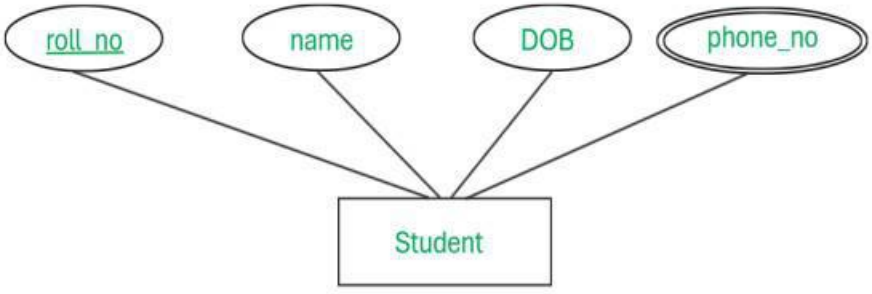
c. Weak Entity
    i. An entity that makes reliance over another entity is called a weak entity
    ii. cannot be uniquely identified by its own attributes alone and depends on a "parent" strong entity for its identification.
    iii. For eg :
        1. A customer entity is a strong entity, since each customer can be uniquely identified by CustomerID primary key.

2. But an entity called OrderItem cannot exist without entity Order. Order's PK OrderID exists as FK for Order entity.

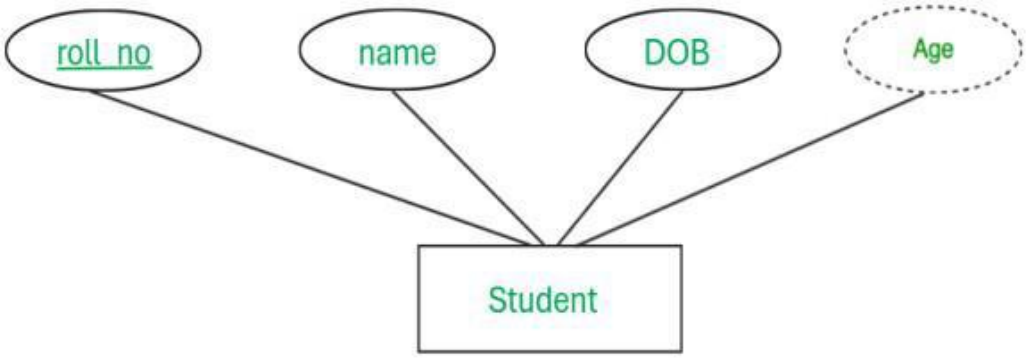
d. Attributes
   i. properties or characteristics of an entity.
   ii. Types of attributes are :
   iii. Simple (Atomic)
      1. cannot be divided further.
      2. Holds a single value, like Name
   iv. Composite
      1. that can be divided into smaller sub-parts, which represent more basic attributes with independent meanings.
      2. Eg : fullName can be divided to first and lastname, address can be divided to country, state, etc



   v. Single-Valued Attribute
      1. holds a single value for a particular entity
      2. Roll No. for a Student Entity
   vi. Multi-Valued Attribute
      1. hold multiple values for a particular entity.
      2. phoneNumbers



vii. Derived Attribute
    1. whose value can be derived from other attributes or related entities.
    2. Age can be derived from Date of Birth
    3. Stored attribute does not require any update while derived attribute require update and its value is derived from other attribute. For example stored attribute is D.O.B while age is the example of derived attribute.

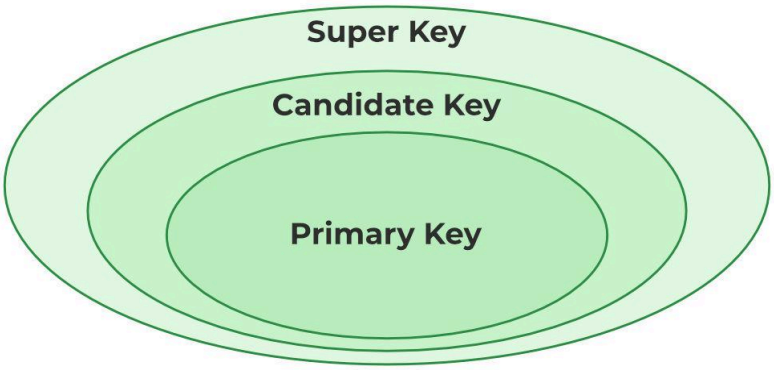


viii. Key Attribute
    1. that uniquely identifies an entity in an entity se
    2. Primary Key
    3. EmployeeID for Employee table
ix. Composite Key Attribute
    1. A combination of two or more attributes that uniquely identify an entity.
    2. For weak entity like OrderItem, combination of OrderID and ItemID can act as a composite key.
x. Stored Attribute
    1. which doesn't require any type of further update since they are stored in the database.
    2. Date of Birth

e. Keys
i. used to identify the tuples(rows) uniquely in the table.
ii. Candidate Key
    1. The minimal set of attributes that can uniquely identify a tuple is known as a candidate key
    2. It must contain unique values.
    3. It can contain NULL values.
    4. Every table must have at least a single candidate key.
    5. A table can have multiple candidate keys but only one primary key.
iii. Primary key
    1. There can be more than one candidate key in relation out of which one can be chosen as the primary key.

iv. Super Key

1. The set of attributes that can uniquely identify a tuple is known as Super Key.
2. A candidate key is a super key but vice versa is not true.
3. Super Key values may also be NULL.

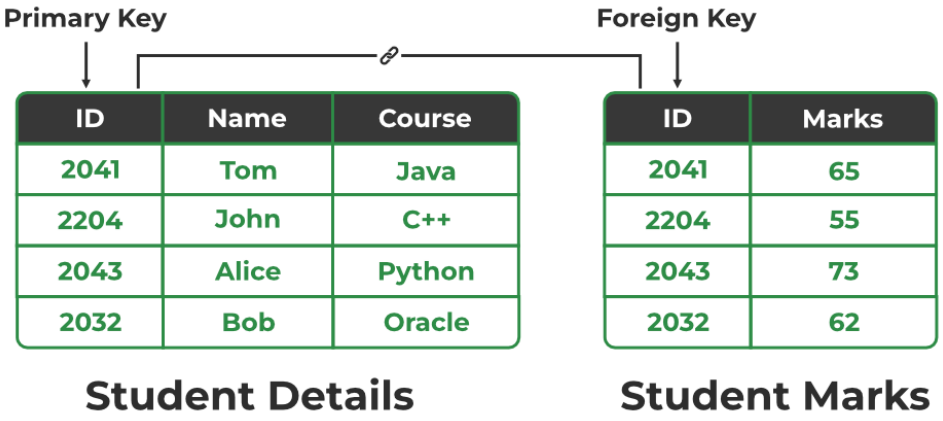


v. Alternate Key

1. The candidate key other than the primary key is called an alternate key.
2. These values are repeated.

vi. Foreign Key

1. It is a key it acts as a primary key in one table and it acts as secondary key in another table.
2. The referencing table is the table that contains the foreign key. This table references the primary key or unique key in another table to establish a relationship between the data in the two tables.
3. The referenced table is the table that contains the primary key (or unique key) being referenced by the foreign key in another table.
4. Eg : Consider a database with Orders and Customers tables. The Orders table contains a foreign key CustomerID that references the primary key CustomerID in the Customers table. So, Orders table is referencing table, whereas Customers is referenced table.



Steps for designing a database

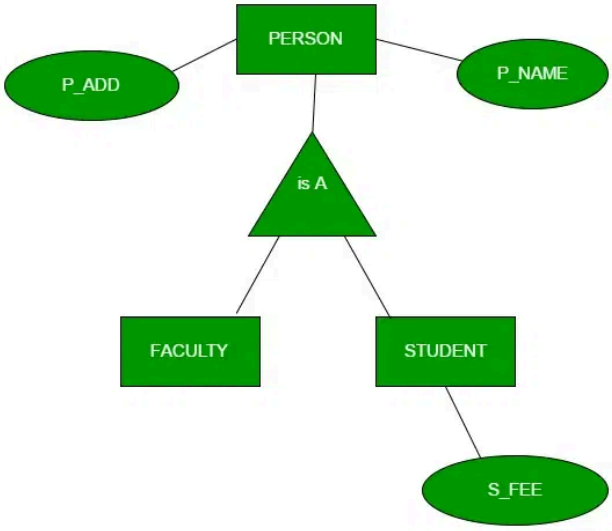
Database designs provide the blueprints of how the data is going to be stored in a system. A proper database design meets all the requirements of users.
So , steps are :

a. Requirements Analysis
    i. identify What kinds of data to store ?
    ii. identify What functionalities to support ?
    iii. identify What are most frequently ?
    iv. conducted by discussions with users, stakeholders and other nontechnical level ko people.
    v. enables db designers to understand business logic to be applied while constructing the db.
b. Conceptual Schema
    i. provides concepts close to the way how users see data
    ii. brief description of data requirements and detailed descriptions of entity types, relationships and constraints.
    iii. since they have **no implementation details**, they are easy to understand as well as they can be easily communicated with non technical users
    iv. converts requirements to a high level data model like ER model.
c. Logical Design
    i. actual implementation of database, using commercial dbms like MySQL, Oracle, etc
    ii. ER Model is converted to actual relational db schema
d. Schema Refinement
    i. Collections in relational db schema are analyzed in order to find problems in the db, such as redundancy, anomalies, etc
e. Physical Design
    i. specify internal storage structures, indexes , access paths etc
f. Security Design
    i. access rules for users' roles

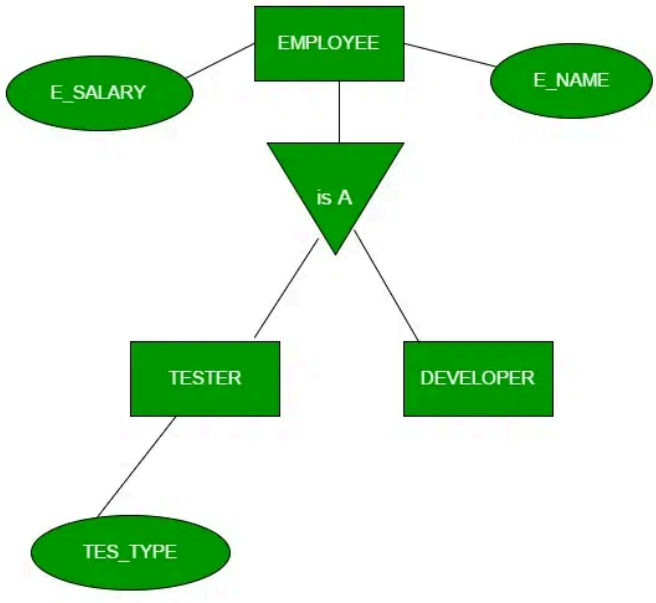
Generalization, Specialization and Aggregation in DBMS

a. Generalization
    i. Bottom-up approach
    ii. Combine lower level entities to form a higher level entity

iii. Let’s take an example, Student and Faculty(Teacher) can be generalized to Person. So, common attributes like name, age become a part of higher entity (Person), whereas specialized entity like college_fee is part of Student.

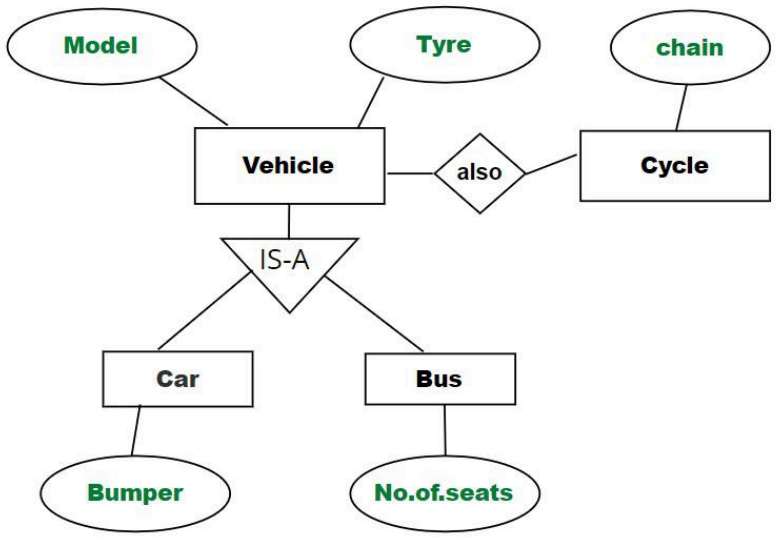


b. Specialization
   i. Top down approach
   ii. Dividing an entity into lower sub entities
   iii. For example, if there is an entity called Employee, we can break it down to sub entities like Developer and Tester. Hare, common attributes like name, salary will be part of higher entity Employee, whereas attributes like test_type can be just part of Tester entity only.



Specialization

Inheritance is an important feature of generalization and specialization. Attribute inheritance means attributes of higher level entity can be inherited by lower level entities. Also, there is participation inheritance, which means that relationships involving higher level entities are also inherited by lower level entities.



GENERALIZATION	SPECIALIZATION
Generalization works in Bottom-Up approach.	Specialization works in top-down approach.
In Generalization, size of schema gets reduced.	In Specialization, size of schema gets increased.
Generalization is normally applied to group of entities.	We can apply Specialization to a single entity.
Generalization can be defined as a process of creating groupings from various entity sets	Specialization can be defined as process of creating subgrouping within an entity set
In Generalization process, what actually happens is that it takes the union of two or more lower-level entity sets to produce a higher-level entity sets.	Specialization is reverse of Generalization. Specialization is a process of taking a subset of a higher level entity set to form a lower-level entity set.
Generalization process starts with the number of entity sets and it creates high-level entity with the help of some common features.	Specialization process starts from a single entity set and it creates a different entity set by using some different features.
In Generalization, the difference and similarities between lower entities are ignored to form a higher entity.	In Specialization, a higher entity is split to form lower entities.