Name: Khan Aamina L033 MSC(DSAI)

EXPERIMENT: No. PRACTICAL: NO 1

Page No. 1
Date

DDL Operations on Relational Schema.

Write-up:-

I. Codd's 12 rules:

* Rule 1: The Information Rule: All information, whether it is user information or metadata, that is stored in a database must be entered as a value in a cell of a table. It said that everything within the database is organized in a table layout.

* Rule 2: The Guaranteed Access Rule: Each data element is guranteed to be accessible logically with a combination of the table name primary key (row) and attribute name (column).

* Rule 3: Systematic Treatment of Null Values: Every Null is a database must be given a systematic and uniform tratment.

* Rule 4: Active Online Catalog Rule: The database catalog, which contains metadata about the database, must be stored and accessed using the same relational database management system.

* Rule 5: The Comprehensive Data Sublanguage Rule: A Crucial component of any efficient database system is its ability to offer an easily understandable data manipulation language (DML) that facilitates defining, querying and modifying information within the database.

* Rule 6: The View Updating Rule: All views that are theoretically updatable must also be updatable by the system.

* Rule 7: High Level Insert, Update and Delete: A successful database system must possess the features of facilitating high-level insertions, updates, and deletions that can grant users the ability to conduct these operations with ease through a single query.

* Rule 8: Physical Data Independence: Application programs and activities should remain unaffected when changes are made to the physical storage structures or methods.

* Rule 9: Logical Data Independences: Application programs and activities should remain unaffected when changes are made to the logical structure of the data such as adding or modifying tables.

* Rule 10: Integrity Independences: Integrity constraints should be specified separately from application program and stored in the catalog. They automatically enforced by the database system.

* Rule 11: Distribution Independences: The distribution of data across multiple location should be invisible to users and the database system should handle the distribution transparently.

* Rule 12 : Non-Subversion Rule : If the interface of the system is providing access to low-level records, then the interface must not be able to damage the system and bypass security and integrity constraints.

II. RDBMS VS DBMS

RDBMS	DBMS
RDBMS stores data in tabular form	DBMS stores data as file
Data elements need to access individually	Multiple data elements can be accessed at the same time.
No relationship between data	Data is stored in the form of tables which are related to each other
Normalized is ~~not~~ present	Normalized is not present.
It deals with ~~small~~ large quantity of data	It deals with ~~large~~ small amount of data.
RDMS support distributed database	DBMS does not support distributed database.
Keys and indexes do not allow Data redundancy	Data redundancy is common in this model.
It is used to handle large amount of data	It is used for small organized and deal with small data.
All 12 Codd rules are statisfied	Not all codd rules are satisfied.
More security measures provided	Security is less.
Ex: MySQL, Sql Server, Oracle & Microsoft Access etc.	Ex: XML, Window Registry, Forxpro etc.

III. Types of Attributes :-

1. Simple Attribute : An attributes that cannot be further subdivide into component is a simple attribute.
eg: rollno of student, ID number, gender etc.

2. Composite Attribute: An attribute that can split into components is a composite attribute.
eg: address can be further split into house number, city, state and pin code.

3 Single- Valued Attribute: The attribute which takes up only a single value for each entity instance is a single valued attributes.
eg: The age of a student, Aadharcard number.

4. Multi-Valued Attributes: The attribute which takes up more than a single value for each entity instance is a multi-valued attribute. And it is represented by double oval shape.
eg: Phone no. of a student: landline and mobile.

5 Derived Attribute: An attribute that can be derived from other attribute is derived attributes. And it is represented by dotted oval shape.

IV * Types of Key Attribute:

1 Candidate key: The minimal set of attributes that can uniquely identify a tuple is known as a candidate key.
eg: Studo_no in Student relation.

2. Primary key: There can be more than one candidate key in relation out of which one can be chosen as the primary key primary key are not necessarily to be a single column; more than one column can also be a primary key for a table.
eg: stud_no as well as stud_phone are candidate but Stud_no can be the primary key

3 Super key: The set of attributes that can uniquely identify a tuple is known as Super key. A super key is a group of single or multiple keys that identifies rows in a table. It support null values.

4 Alternate key: The candidate key other than the primary key is called an alternate key. It is a secondary key. All key which are not primary key are called alternate key
eg: SName and addresses is Alternate keys.

5 Foreign key: If an attributes can only take the values which are present as values of some other attributes. it will be a foreign key. It is a key it acts as a primary key in one table and it acts as secondary key in another table.
eg: Dno is primary key in the Dept table and non-key in Emp

6 Composite Key : It acts as a primary key if there is no primary key in a table. Two or more attributes are used together to make a composite key.
Different combinations of attributes may give different accuracy in term of identifying the rows uniquely.

7 Unique Key : Ensure unique column values but allow Null.

V. ERD

i The Entity Relationship Model is a model for identifying entity to be represented in the database and representation of how those entities are related. The ER data model specifies enterprise schema that represents the overall logical structure of a database graphically.

ii ER diagram provide the purpose of real-world modelling of object which makes them intently useful.

iii ER diagrams represent the ER model in a database, making them easy to convert into relations.

iv ER diagram require no technical knowledge of the underlying DMDs used.

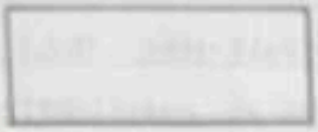
v ER model is used to model the logical view of the system from a data perspective with consists of these symbol.

a] Rectangle : Rectangle represent entities in the er model.

b] Ellipses : Ellipses represent Attributes in the Er Model.

c] Diamond : Diamonds reprsent Relationship among entites.

d] line : Lines represent attributes to entities and entity set with other relationship types.

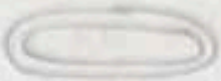


Rectangle

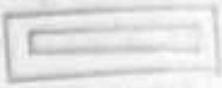
Ellipse

diamond

line



double ellipse



double Rectangle

e Double Ellipse : Double Ellipse represent Multivalued Attributes.

f Double Rectangle: Double Rectangle represent a Weak Entity.

vi ER model consists of Entities, Attributes, and Relationships among Entities in a Database System.

VI. Constraints

i Constraints are used to ~~limit~~ specify rules for the data in table

ii Constraint are used to limit the type of data that can go into a table. This ensure the accuracy and reliability of the data in the table. If there is any violation between the constraint and the data action, the action is aborted.

iii Constraints can be column level or table level. Column level constraints apply to a column, and table level constraints apply to the whole table.

iv Following constraints are commonly used in :

a] Not Null : Ensures that a column cannot have a null value

b] Unique : Ensures that all value in a column are different.

c] Primary key : A combination of a Not Null and unique. Uniquely identifies each row in a table.

d] Foreign key : Prevents actions that would destroy links between tables.

e] Check : Ensures that the values in a column satisfies a specific condition.

Teacher's Sign. :

f Default : Set a default value for a column if no value is specified.

g Create index : Used to create and retrieve data from the database very quickly

VII. DDL

i DDL actually represent Data Definition Language which is actually a set of command used to create a structure and maintain database. Those would include create, alter, drop, truncate and rename statement for creatings, changing the structure of, and dropping structure in the database, such as table.

ii DDL include following command.

i Create : This is used to create table in the relational database.

ii Alter : Alter command is used for altering the table in many form.

iii Truncate : This command removes all the records from a table but this command will not destory the the table structure.

iv Drop : This command completely remove the table from the database along with the destruction of the table.

4