1/8/24

Unit-2

Relational model:-

Introdvction to Relational Data model:-

It was introduced by EF-codd in 1970. This Data model defines the Data in tabular format. tables are also known as Relations. Each table must contain table name, attributes, tuples, instances & Domain. column headings are known as attributes. rows are known as Tuples. Row Data is known as instance. column data is known as Domain.

Ex:-

student

Attributes →	Id	Name	age
Tuples	1	sethu	18
	2	vardhan	18

Relation name : student

No. of Tuples : 02

No. of Attributes : 03

Instances : 1, sethu, 18
2, vardhan, 18.

Domain : Id, 1, 2

Imp

Keys:-

Key is an attribute (or) set of attributes that are used to identify unique no.of tuples in a relation. There are 4 types of keys, they are

1. primary key.
2. Super key
3. Candidate key
4. Foreign key.

1. Primary key:- It is a first key used to identify particular rows in a table. It doesn't accept duplicate values and

none values. It can't be changed in any time.

Ex:-

Student

H.T NO	name	age
2297	Sethu	18
2298	vardhan	19

In the above example H.T NO is the primary key.

2. Super key:-

It is a combination of key atti attribute & non-key attribute.

Ex:-

Student

H.T NO	Name	marks
123	Sethu	75
789	vardhan	76
456	Dileep	78

Superkeys:-

{HT.NO, name}

{HT.NO, marks}

3. Candidate key

Except primary key remaining key attributes are Candidate keys. The values in candidate key are may or may not be changed.

Ex:-

H.T.NO	Name	Marks	mobile NO	Email
215	Sethu	80	8888	Sethu06@
326	vardhan	81	8879	vardhan@

mobile NO

Emails

Candidate Keys

In the above example mobile.NO & Email are the candidate keys because a person can change their mobile no, Email Id at any time.

4. Foreign Key:-

It is used to create a relationship between two tables. It contain duplicate values

Ex:-

primary key

Person

Person Id	Name
123	setthu
431	vardhan
567	Dileep

primary key

products

Foreign key

Product Id	Name	Person Id
1288	Rice	123
1289	oil	123
1290	Sugar	431
1291	Salt	567

② Constraints:-

These are the rules that defines what data can be entered into a database. It is used to maintain quality of information. There are 4 types of constraints

1. Domain constraints
2. Entity integrity constraints
3. Referential integrity constraints
4. Key integrity constraints.

1. Domain Constraints:-

Domains are the values that are presenting attributes. Every attribute must contain valid set of values.

Ex:-

Student

Id	Name	age
1	Sethu	18
2	vardhan	A

→ This is not allowed because age contains integers not values.

2. Entity integrity Constraints:-

This rule defines primary key can't be null.

Ex:-

student

Id	Name	age
1	Sethu	18
•	vardhan	18

→ This is allowed because primary key can't be Null.

3. Referencial intergrity constraints:-

This rules defines the values in foreign key must be available in primary key.

Ex:-

person Id	Name
123	Sethu
456	vardhan
789	charan

product Id	Name	person Id
781	oil	123
652	sugar	456
352	rice	123
129	salt	789
110	wheat	520

520 is not allowed because "520" person id is not available in person t

4. Key integrity Constraints:-

This rule defines primary key doesn't contain duplicate values.

Ex:-

Person

Person Id	name
0143	charan
0231	sethu
0018	vardhan
0143	Bumrah

→ This is not allowed because primary key doesn't contain repeated values.

E.F Codd Rules:- (Edgar Frank codd)

RDBMS was implinnented by E.F Codd in 1985

Rule 0: Foundation Rule :-

The database must be in tabular format.

Rule 1: Information Rule :-

This rule defines the information must be stored in rows and columns of a table.

Rule 2: ~~Direct~~ Guranted Access Rule

This rule describes relational database contains primary key, table name & column names.

Rule 3:- Systematic treat of null values :-

This rule defines the primary key shoundn't be (missing value) null.

Rule 4 :- Active /dynamic online Catalog :-

This rule defines the every relational database must contain a logical structure .(schema).

Rule 5: Comprehensive data sublanguage :-

This rule defines relation database must supports various programming languages like C, C++ & Java

Rule 6:- View updating Rule :-

table can be theoriatically updated in whole data base so that every user ~~has~~ contain updated database

Rule 7 :- Relational level operations :-

Every relation should follow operations such as insert, update & Deleted.

Rule 8 :- physical data independence :-

This rule defines physical data independance it means when we change the multiple locations of a software the data can't be erased.

Rule 9 :- logical data independence :-

The ability to change the logical structure of the data but not affecting the application.

Rule 10 :- Integrity Independence rule :-

This rule defines the relational database must support different types of costraints. (rules)

Rule 11 :- Distribution Independence :-

It represent a database that must

work properly even if it is stored in different locations and used by different users.

Rule 12:- non-~~subertion~~ subversion rule:-

This rule difines when we change the multiple Version of SQL software, the data can't be changed.

Functional Dependency:-

The relationship between two attributes in a relation is known as functional dependency.

A → B ('A' determines 'B' & 'B' depends on 'A')

Here 'A' and 'B' are two attributes.

There are two types of functional dependengies they are... ~~Partial~~

1. Partial Functional dependency.
2. Transitive Functional dependency.

1. Partial Functional dependency:-

It occurs when a table contain two primary key attributes

Emp Id	Emp Name	Address	Dept Id	Dept name

Emp Id → Emp name

Dept Id → Dept name

Emp Id → Address

(→) determin~~ent~~es)

2. Transitive Functional dependency:-

The relationship between two non-key attributes is known as transitive functional dependency.

Ex:-

clg code	College name	Principle

① College → principle
Name Name

② DOB → Age.

Normalization :-

The process of deviding the large tables into Smaller tables is known as Normalization. It is implimented by using different normal forms. with the help of normal forms we can arrange the data in a ~~tabular~~ Standard ~~format~~ tabular format.

The types of normal forms are

1. 1 NF (first normal form) – No multiple values
2. 2 NF (second normal form) – NO partial functional dependeny
3. 3 NF (Third normal fornm) – No transitive functional dependency
4. BCNF (Boyce-Codd normal form) – similar to 3NF

First Normal Form (1NF) :-

It is also knowna as 1NF. This normal form defines each cell in a table must contain only Single value.

Ex:- Before 1NF

Id	name	mobile.NO
1	sethu	7993588 7995598
2	vardhan	8142042 3261027

After 1NF

Id	Name	mobile. No.
1	sethu	7993588
1	sethu	7995598
2	vardhan	8142042
2	vardhan	3261027

2. ~~&~~ Second Normal Form (2NF) :-

It is also known as 2NF. This normal form defines there is no partial fuctional dependency & every table must contain ~~onety~~ only one primary key.

Before 2NF

Emp Id	Emp name	Address	Dept Id	dept name
123	Sethu	Tpt	001	ABC
321	vardhan	plm	002	MNR

After 2NF

Employee

Emp Id	Emp name	Address
123	sethu	Tpt
321	vardhan	plm

Department

Dept Id	Dept Name
001	ABC
002	MNR

3. Third Normal Form (3NF) :-

It is also known as 3NF. This normal form defines every relation must contain only one primary key. & there is no transitive dependency.

college code	college name	principle name
012	ABC	Sethu
013	DEF	vardhan

⇓

college code	college name
012	ABC
013	DEF

college code	principle name
012	Sethu
013	vardhan.

4. Boyce-codd normal form:-

It is also known as BCNF. It is an advanced version of 3NF. It is mainly used in large databases. the rules are similar to 3NF.

1. Explain E.F codd rules?
2. Define key? Explain the types of keys with an example.
3. Explain the d/f types of constrains in relational database
4. Define normalization? Explain the d/f types of normal forms with an example.
5. Define functional dependency? Explain the types of functional dependencypies.