Normalization | KEYS | Anomalies

Trainer: Mohammad Shabbir Taibani

AGENDA

- Different types Keys
- Normalization in DBMS
- Anomalies

CANDIDATE KEY

KEYS in DBMS is an attribute or se of attributes which helps you to identify a row (record) in a relation (table)

They allow us to find the relation between two tables

TYPES OF KEYS

There are mainly 8 different types of kyes in DBMS and each key has its different functionality:

1. Super Key
2. Primary Key
3. Foreign Key
4. Composite Key
5. Candidate Key
6. Alternate Key
7. Compound Key
8. Surrogate Key

COMMON KEYS

- **Super Key** : A Super key is a group of single or multiple keys which identifies rows in a table
- **Primary Key :** is a column or group of columns in a table that uniquely identify every row in a table
- **Foreign Key :** is a column that create relationship between two tables , the purpose of Foreign keys is to maintain data integrity and allow navigation between two different instance of an entity
- **Candidate Key** : is a set of attributes that uniquely identify tuples in a table, Candidate key is a super key with no repeated attributes
- **Composite Key**: Combination of 2 or more column

SUPER KEY

A Super key is a group of single or multiple keys which identifies rows in a table
It is most general type of key in DB

Srno	Date	EmpID	Name
1	01-01-2022	E1001	Ajay
2	01-01-2022	E1002	Vijay
3	01-01-2022	E1003	Amar
4	02-01-2022	E1004	Akbar
5	02-01-2022	E1005	Anthony
6	02-01-2022	E1006	Ajay

**PRIMARY KEY

The Primary Key constraint Uniquely identifies each record in table

Primary keys must contain UNIQUE values and cannot contain null values

A table can have only ONE primary key: and in the table , this primary can consist of single or multiple columns (fields)

EmpID	Name	Sales
E1001	Ajay Singh	704
E1002	Vijay	636
E1003	Amar	767
E1004	Akbar	454
E1005	Anthony	670
E1006	Ajay Verm	1000

Date & Emp ID	Date	EmpID	Name	Sales
010122-E1001	01-01-2022	E1001	Ajay Singh	704
010122-E1002	01-01-2022	E1002	Vijay	636
010122-E1003	01-01-2022	E1003	Amar	767
020122-E1001	02-01-2022	E1001	Ajay Singh	454
020122-E1002	02-01-2022	E1002	Vijay	670
020122-E1003	02-01-2022	E1003	Amar	1000

FOREIGN KEY

The Foreign Key constraint is used to prevent actions that would destroy links between tables
A Foreign Key is a field (collection of field) in one table , that refers to the primary key in another table
The table with the Foreign key is called the child table and the table with the primary key is call the reference or parent table
The FK constraint prevents invalid data from being inserted into FK Column , because it has to one of the values contained in the parent table

PK

Parent Table

EmpID	First Name	Last Name	Age
E1001	Ajay	Singh	26
E1002	Vijay	Verma	22
E1003	Amar	Khanna	23
E1004	Akbar	Khan	25
E1005	Anthony	Dsouza	30
E1006	Ajay	Verma	28

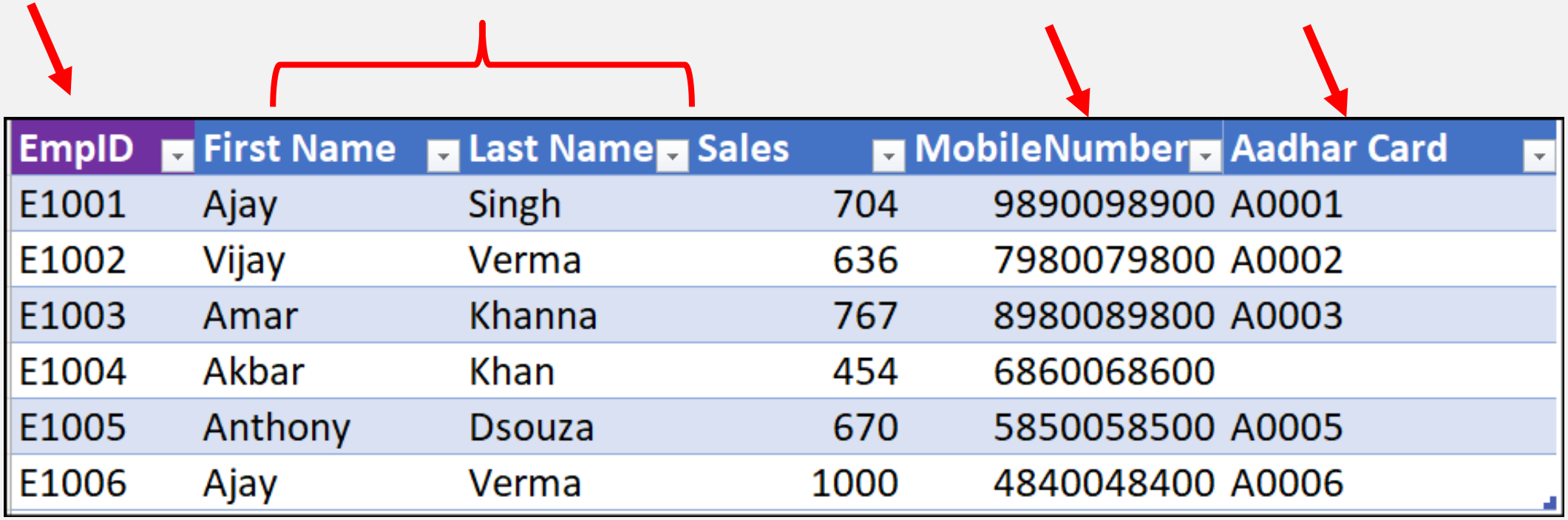
FK

Child Table

EmpID	OrderID	Order Value
E1001	O001	704
E1001	O002	636
E1002	O003	767
E1003	O004	454

CANDIDATE KEY

A Candidate is as subset of super key , A Candidate key is single field or the least combination of field that uniquely identifies each record in the table



EmpID	First Name	Last Name	Sales	MobileNumber	Aadhar Card
E1001	Ajay	Singh	704	9890098900	A0001
E1002	Vijay	Verma	636	7980079800	A0002
E1003	Amar	Khanna	767	8980089800	A0003
E1004	Akbar	Khan	454	6860068600	
E1005	Anthony	Dsouza	670	5850058500	A0005
E1006	Ajay	Verma	1000	4840048400	A0006

COMPOSITE KEY

Whenever a primary key consists of more than one attribute , it is known as composite key.
This key is also known as Concatenated key

Date & Emp ID	Date	EmpID	Name	Sales
010122-E1001	01-01-2022	E1001	Ajay Singh	704
010122-E1002	01-01-2022	E1002	Vijay	636
010122-E1003	01-01-2022	E1003	Amar	767
020122-E1001	02-01-2022	E1001	Ajay Singh	454
020122-E1002	02-01-2022	E1002	Vijay	670
020122-E1003	02-01-2022	E1003	Amar	1000

ANALYZING DATABASE DESIGN

Let’s consider a single large database have only one single relation

This Large database defined as a single relation may result in duplication of data

Can you think of the disadvantages of having a large database with repetitive data ?

ONE TABLE EXAMPLE

OrderTable

Index	Date	OrderID	CustID	Custom	Address	Product	Product	UnitPrice	Product Family	Vendor	Vendor Name	Vendor Lo	Qty
1	01-Oct-21	O001	C001	A	Pune	P001	Cinthol	₹ 10	Daily Needs	V001	ABC	Pune	25
2	03-Oct-22	O002	C002	B	Mumbai	P002	Lux	₹ 12	Daily Needs	V001	ABC	Pune	17
3	03-Oct-22	O003	C004	D	Banglore	P001	Cinthol	₹ 10	Daily Needs	V002	XYZ	Mumbai	22
4	04-Oct-22	O004	C002	B	Mumbai	P002	Lux	₹ 12	Daily Needs	V002	XYZ	Mumbai	15
5	04-Oct-22	O005	C001	A	Pune	P003	Tshirt	₹ 13	Clothing	V001	ABC	Pune	13
6	05-Oct-22	O006	C002	B	Mumbai	P004	Jeans	₹ 14	Clothing	V001	ABC	Pune	14
7	05-Oct-22	O007	C003	C	Delhi	P003	Tshirt	₹ 13	Clothing	V002	XYZ	Mumbai	19
8	05-Oct-22	O008	C004	D	Banglore	P004	Jeans	₹ 14	Clothing	V002	XYZ	Mumbai	11
9	06-Oct-22	O009	C001	A	Pune	P002	Lux	₹ 12	Daily Needs	V001	ABC	Pune	10
10	06-Oct-22	O010	C002	B	Mumbai	P003	Tshirt	₹ 13	Clothing	V002	XYZ	Mumbai	18

NORMALIZATION IN DBMS

What is Normalization?

• Normalization is a process of decomposing the relations into smaller, simpler, and well-structured relations with fewer attributes.

• It is the process of organising the data in the database.

• It is used to minimise the data redundancy from a relation or set of relations and is also used to eliminate undesirable characteristics like Insertion, Update, and Deletion Anomalies.

• Normalization consists of a series of guidelines that helps to guide you in creating a good database structure.

MULTIPLE TABLE EXAMPLE

Product Table

Product ID	Product Name	Cost
P001	Cinthol	10
P002	Lux	12
P003	Tshirt	13
P004	Jeans	14

OrderTable

Index	Date	OrderID	CustID	Product ID	Qty
1	01-Oct-21	O001	C001	P001	25
2	03-Oct-22	O002	C002	P002	17
3	03-Oct-22	O003	C004	P001	22
4	04-Oct-22	O004	C002	P002	15
5	04-Oct-22	O005	C001	P003	13
6	05-Oct-22	O006	C002	P004	14
7	05-Oct-22	O007	C003	P003	19
8	05-Oct-22	O008	C004	P004	11
9	06-Oct-22	O009	C001	P002	10
10	06-Oct-22	O010	C002	P003	18

Customer Table

CustID	CustomerName	Address
C001	A	Pune
C002	B	Mumbai
C003	C	Delhi
C004	D	Banglore

Vendor Location

VendorID	Vendor Name	Vendor Locati
V001	ABC	Pune
V002	XYZ	Mumbai

Segment Table

Product ID	Product Family
P001	Daily Needs
P002	Daily Needs
P003	Clothing
P004	Clothing

Vendor Table

VendorID	Product Family	Product Name
V001	Daily Needs	Cinthol
V001	Daily Needs	Lux
V002	Daily Needs	Cinthol
V002	Daily Needs	Lux
V001	Clothing	Tshirt
V001	Clothing	Jeans
V002	Clothing	Tshirt
V002	Clothing	Jeans

ANOMALIES IN DBMS

Data modification anomalies can be categorized into three types:

- Insertion Anomaly: Insertion Anomaly refers to when one cannot insert a new tuple into a relationship due to lack of data.

- Deletion Anomaly: The delete anomaly refers to the situation where the
- deletion of data results in the unintended loss of some other important data.

- Updatation Anomaly: The update anomaly is when an update of a singledata value requires multiple rows of data to be updated.

Let's understand anomalies with the help of example ->

TYPES OF ANOMALY

_id	e_name	e_address	e_dept
101	Rick	Delhi	D001
101	Rick	Delhi	D002
123	Maggie	Agra	D890
166	Glenn	Chennai	D900
166	Glenn	Chennai	D004

UPDATE ANOMALY : If we want to update Rick address, then it need to update 2 rows, if some how it is updating only 1 row, then Rick will have 2 different address

INSERT ANOMALY :Suppose a new employee joins the company, who is under training and currently not assigned to any department then we would not be able to insert the data into the table if the e_dept field doesn't allow nulls.

DELETE ANOMALY :Suppose, if at a point in time the company closes the department D890 then deleting the rows that are having e_dept as D890 would also delete the information of employee Maggie since she is assigned only to this department.

QUESTION N ANSWER

THANK YOU