Relational Database

By –

Rajeev Srivastava

C-DAC Mumbai

Sep, 2019

RDBMS

An Introduction

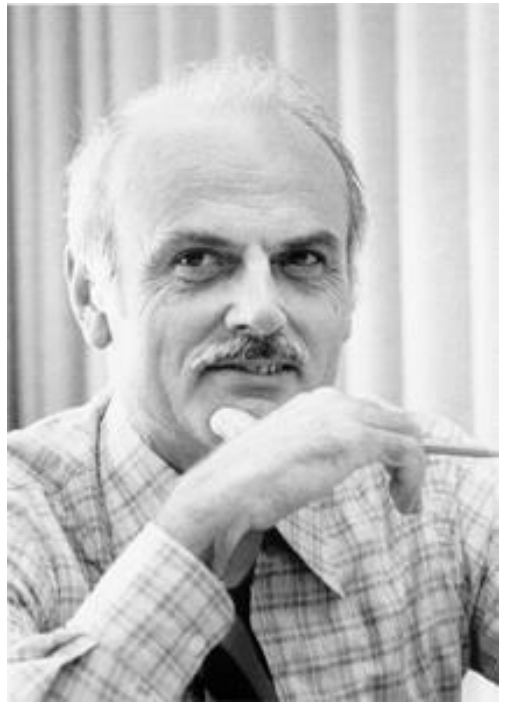
Sep, 2019

Relational Databases

- Based on Relational Model
- Stores data in form of Tables = Relations.
- Data items or information stored in a Relational Database is related to each other. Hence the term Relational.
- Follow a common query language: SQL
- Ex: PostgreSQL, MySQL, MariaDB, Oracle, MS SQL Server, IBM DB2 etc.

Relational [Database] Model

- First defined by Computer Scientist **Edgar Frank Codd** in 1969.
- Mathematical model, based on Relational Algebra and Predicate Logic.
- Very well defined and extremely mature model in terms of data storage and representation.
- All RDBMS (Relational Database Management Systems) derived from Relational model.



A Relation

- In Relational Model data is modelled in form of Relations represented by tabular structure.
- Consider the relation EMPLOYEE represented by the following table:

EmpCode	Name	Desig	Grade	JoinDate	BasicSalary	Gender	DeptCode

Tuples in a Relation

- A relation is a set of tuples; each row here is a tuple :

EmpCode	Name	Desig	Grade	JoinDate	BasicSalary	Gender	DeptCode	
								1
								2
								3
								4
								5
								6
								7

- Cardinality of a Relation: Number of Tuples in a Relation at a given time.

Attribute in a Relation

- An attribute represents a quality/information about an entity.
- A tuple consists of Attribute values.

EmpCode	Name	Desig	Grade	JoinDate	BasicSalary	Gender	DeptCode

- A degree or arity of a Relation is the number of attributes in it at a given time

Domains

- Each attribute has a domain associated with it.
- Attribute values in a relation are restricted to the values from its domain.

EmpCode	Name	Desig	Grade	JoinDate	BasicSalary	Gender	DeptCode

DESIG
PE
TO
STO

DEPT
ACCO
PURC
COUR

Consider the Employee relation as :

```
CREATE TABLE employee (
      EmpCode     integer(4),
      Name        char(30),
      Desig       char(4),
      Grade       integer(4),
      JoinDate    date,
      Basic       integer(7),
      Gender      char(1),
      DeptCode    char(4)
)
```

Domains of Attributes: Example

```
EmpCode     set of all 4-digit numbers
Name        set of all 30-alpha characters
Desig       set of all designation codes
Grade       set of all grade values
JoinDate    set of all dates  (in a given range)
Basic       set of all possible values for basic
Gender      set {'M','F‘, ‘T’}
DeptCode    set of all dept codes
```

A *Relation* may be represented as a *Table* where

Relation	Table
Tuple	Row/Record
Attribute	Column
Degree/Arity	No of Columns in the table
Cardinality	No or Rows in the table
Domain	Pool of acceptable values for a column
Primary Key	Unique Identifier

But, a Relation is not a Table, because :

- A table has an inherent order for rows; there is no concept of order for tuples in a relation.
- A relation must have a Primary Key; a table need not have an identifier/Primary Key.
- The tuples in a relation must be unique; there is no such restriction for tables

Keys in a Relation

- Keys are used to identify rows uniquely in a Relation. (Except Foreign Key)
- A **single column** or a **set of columns** can be defined as a Key.
- Major Type of Keys:
  - Candidate Key
  - Primary Key
  - Composite Key
  - Foreign Key
  - Unique Key

A Sample Table: Employee

Empid (PK)	empname	designation	salary	bonus	dept	report_to
1001	Aniket	Project Engineer	40000	5000	SE	1002
1002	Avinash	Team Lead	50000	10000	SE	1003
1003	Shweta	Project Manager	70000		SE	1005
1005	Sachin	HR Head	60000		HR	1006
1006	Vikash	CEO		10000		
1007	Sachin	Project Engineer	45000		SE	1005

A Sample Table:

EMPLOYEE_RATING

Empid (PK)	year (PK)	rating
1001	2019	A+
1001	2020	A+
1002	2020	A
1003	2020	B
1005	2019	B+
1005	2020	A
1006	2020	C
1007	2020	D

RATING_MASTER

Rating (PK)	Description
A+	Excellent
A	Very Good
B	Good
B+	Average
C	Below Average
D	Bad

Candidate Key

A **Candidate Key** must satisfy following time-independent properties:

- **Uniqueness**: No two distinct tuples have the same value for the key.
- **Minimalistic**: None of the attributes of the key can be discarded from the key without destroying the uniqueness property.

Candidate Key?

```
create table EMPLOYEE(
      EmpCode     integer(4),
      Name        char(30),
      Desig       char(4),
      Grade       integer(4),
      JoinDate    date,
      Basic       integer(7),
      Gender      char(1),
      DeptCode    char(4)
)
```

Candidate Key?

```
create table EMPLOYEE (
        EmpCode     integer(4),
        Name        char(30),
        Desig       char(4),
        Grade       integer(4),
        JoinDate    date,
        Basic       integer(7),
        Gender      char(1),
        DeptCode    char(4),
        Email       char(100),
        MobileNo    char(16)
)
```

Primary Key

Is a candidate key that have following **two qualities** -

- Uniquely identifies a tuple in a relation
- Must NOT be NULL

- **Should be selected from candidate keys such that it never changes.*
- In a relational database – all tables must have a Primary Key

Primary Key?

```
create table EMPLOYEE (
        EmpCode     integer(4),
        Name        char(30),
        Desig       char(4),
        Grade       integer(4),
        JoinDate    date,
        Basic       integer(7),
        Gender      char(1),
        DeptCode    char(4),
        Email       char(100),
        MobileNo    char(16)

)
```

Composite Key

- A candidate key with two or more attributes that uniquely identifies the tuple in a Relation.
- Also called as compound key

Composite Primary Key

- A primary key which is a composite key is called as Composite Primary Key.

Can we have more than one primary key in a table?

- No. We can NOT.

A table can have only one Primary Key.

- The Primary Key can be defined on a single column or more than one columns. If the Primary Key is defined using more than one columns, it is known as a Composite Primary Key.
- Therefore, a Composite Primary Key in a table does not mean that there are more than one Primary Keys in the table. Instead, a Composite Key uses more than one columns to define a (Single) Primary Key.

Foreign Key

- A Foreign Key is a set of attributes in one relation whose values are required to match one of the values of the primary key of the same or different relation.
- There can be more than one foreign keys in a given relation.

Identify a relation in any system/business, define its Attributes, Domain for each attribute and find out Primary, Key, Foreign Keys, Candidate Keys, Super Key in the relation.

Foreign Key(s)?

```
create table EMPLOYEE(
      EmpCode      integer(4),
      Name         char(30),
      Desig        char(4),
      Grade        integer(4),
      JoinDate     date,
      Basic        integer(7)),
      Gender       char(1),
      DeptCode     char(4)     );

create table DEPT(
      DeptCode     char(4),
      DeptName     char(30),
      Location     char(10)    );
```

Integrity Rules

- **Entity Integrity**: Implemented through Primary Key
- ***"No Attribute participating in the primary key of a relation may accept null values"***
- Guarantees that each tuple will have a unique identity.
- **Referential Integrity**: Implemented through Foreign Key, *"Values of the foreign key*

  (a) must be either null, or

  (b) if non-null, must match with the primary key value of some tuple of the `parent' relation. The reference can be to the same relation"

- *Foreign Key is also know as Reference/Referential key.