
G
O
D
O
O
O
O
D

**Difference between file system and dbms(DBMS vs file system)**

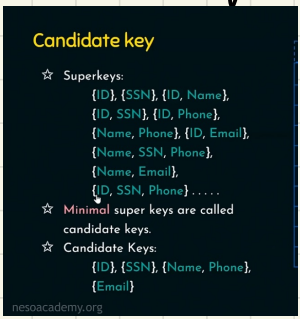
| File system | DBMS |
|---|---|
| File system is a collection of data. Any management with the file system, user has to write the procedures. | DBMS is a collection of data and user is not required to write the procedures for managing the database. |
| File system gives the details of the data representation and Storage of data. | DBMS provides an abstract view of data that hides the details. |
| In File system storing and retrieving of data cannot be done efficiently. | DBMS is efficient to use since there are wide varieties of sophisticated techniques to store and retrieve the data. |
| Concurrent access to the data in the file system has many problems like : Reading the file while other form of locking. deleting some information, updating some information | DBMS takes care of Concurrent access using some form of locking. |
| File system doesn't provide crash recovery mechanism. Eg. While we are entering some data into the file if System crashes then content of the file is lost | DBMS has crash recovery mechanism, DBMS protects user from the effects of system failures. |
| Protecting a file under file system is very difficult. | DBMS has a good protection mechanism. |


Super key → we derive other key.

Super key → Uniquely Identify

Super key → Can combine with other attribute

**Candidate key:** Minimal chosen Super key is the Candidate key.




No repetation

19. Neso Academy.

**Primary key:** Unique but no null value

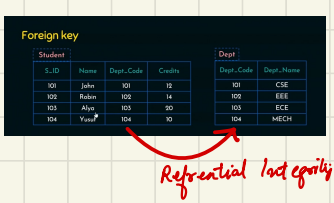
Chosen by DPA

**Alternate key:** Keys which are not primary key

**Unique key:** Can also unique but also have null value

**Composite key:** key with more than one attribute.

**Foreign key:**




Referential Integrity

# SQL Commands

53 Q Smiths.

DDL - Data Definition Language

- Create
- Alter
- Drop
- Truncate
- Rename

Structure delete
|| Data removal

```
Create table <table-name>
(
Column 1 name  datatype,
Column 2 name  datatype,
Column 3 name  datatype(7) no comma
);
desc table-name
```

```
Create table emp_detail
(
id int,
name varchar2(10)
Salary number(10)
);
desc emp;
```

TABLE EMP

| Column | Null? | Type |
|---|---|---|
| ID | - | NUMBER |
| NAME | - | VARCHAR2(10) |
| SALARY | - | NUMBER(10,0) |

Download CSV
3 rows selected.

Autocommit Display 10

```
create table employee
(
id int,
name varchar(10)
);
alter table employee add address varchar(10);
desc emp11
alter table employee drop column address;
alter table employee modify id varchar(30);
alter table employee rename column id to roll_no;
alter table employee rename to emp11;
alter table emp11 add primary key (roll_no)
```

Results Explain Describe Saved SQL History

Object Type TABLE Object EMP11

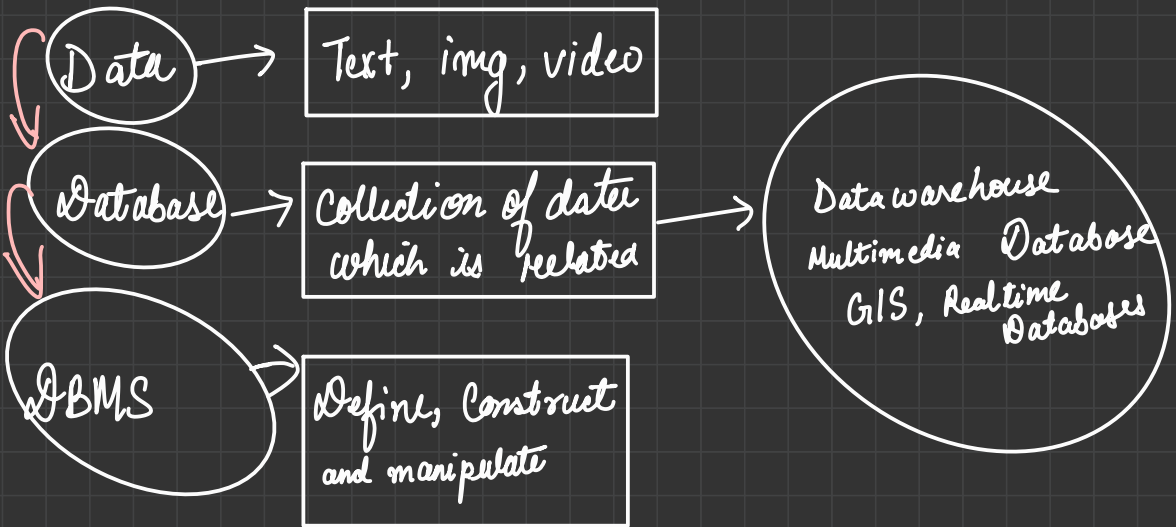
| Table | Column | Data Type | Length | Precision | Scale | Primary Key | Nullable | Default | Comment |
|---|---|---|---|---|---|---|---|---|---|
| EMP11 | ROLL_NO | Varchar2 | 30 | - | - | 1 | - | - | - |
| | NAME | Varchar2 | 10 | - | - | - | ✓ | - | - |

1 - 2

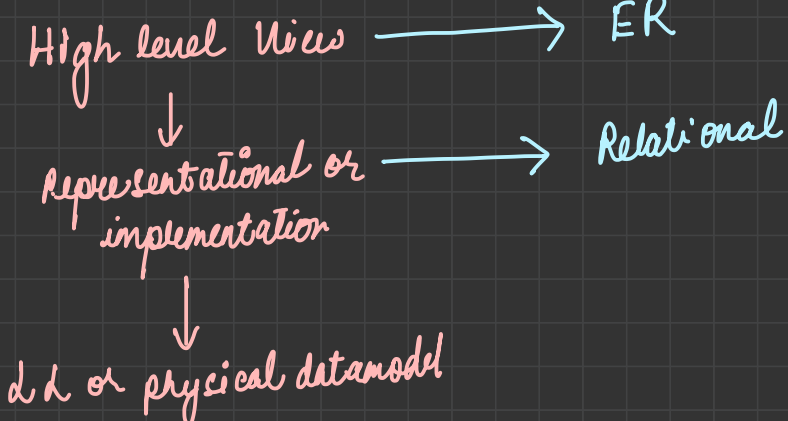
55

# Data Vs Database

RBR-1




RBR-2



# ER Model (Entity Relation Model)

RBR3

Entity → Thing → Person

Attributes → Properties → name, age

Relationship → association → works for

PERSON (Age, Name, add) → (26, Raj's, ...)

Entity type — Entity

State of Database

Rbr-4

Attributes:

Composite vs Simple attribute

(collection of simple attr.)

Name → FN, MN, LN

Single valued vs Multi valued.

Age

Ph-no.

Address

# Stored vs derived attribute

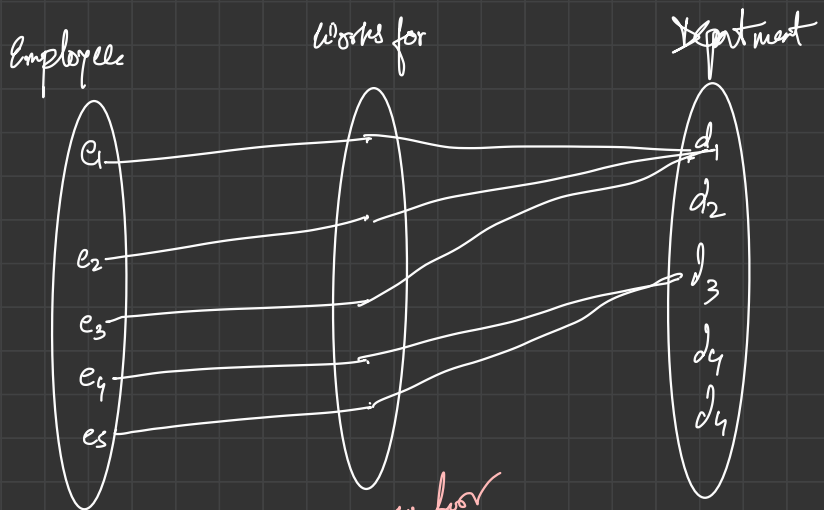
DOB

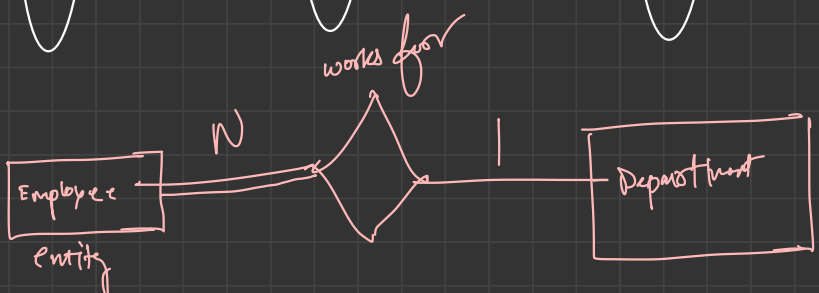
Age — can be derived from DOB

# Complex attribute.

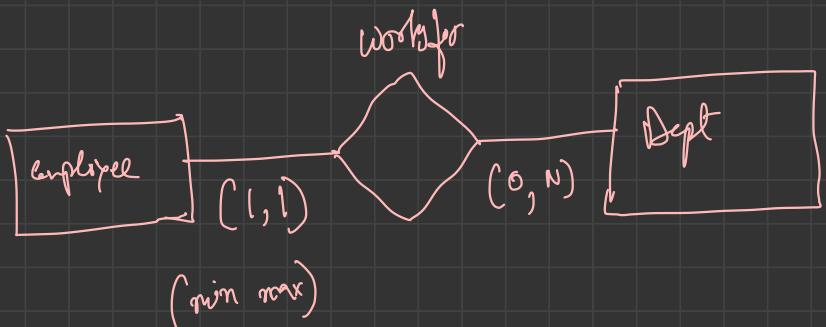
composite + multivalued.

Rbr-s



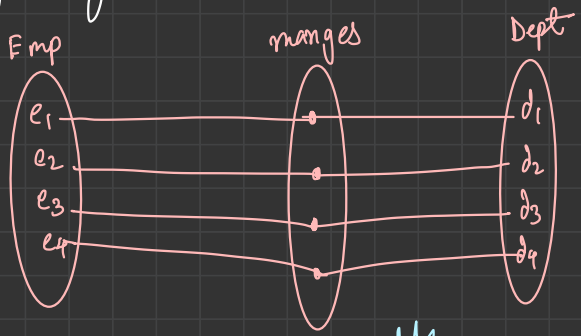



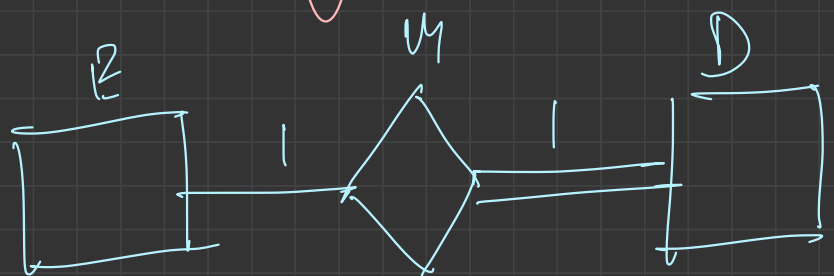

or

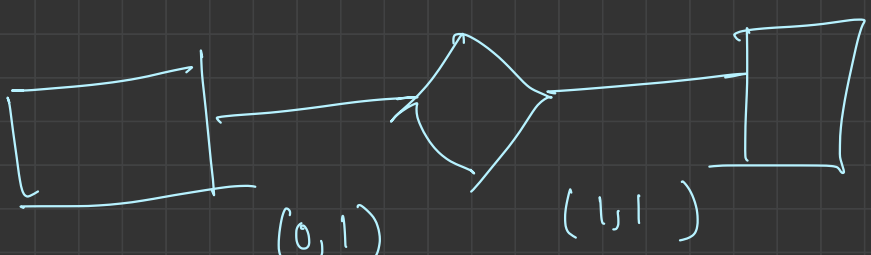


Every dept should have a manager and only one employee manages a dept and an employee can manage only one dept.

Rbr-6




There is no hard & fast rule that every employee should manage a dept.