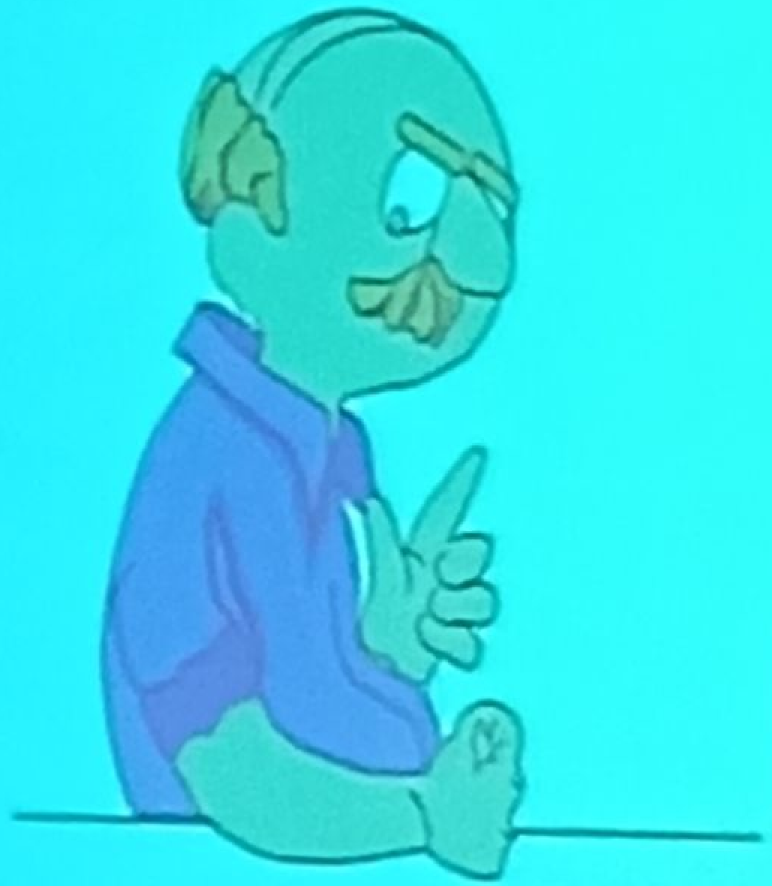

# Database Management Systems

$(A'+B)'$

Each row of data in a table is uniquely identified by a primary key (PK).

You can logically relate data from multiple tables using foreign keys (FK).

Table Name: **EMPLOYEES**

| EMPLOYEE_ID | FIRST_NAME | LAST_NAME | DEPARTMENT_ID |
|---|---|---|---|
| 174 | Ellen | Abel | 80 |
| 142 | Curtis | Davies | 50 |
| 102 | Lex | De Haan | 90 |
| 104 | Bruce | Ernst | 60 |
| 202 | Pat | Fay | 20 |
| 206 | William | Gietz | 110 |

…

Table Name: **DEPARTMENTS**

| DEPARTMENT_ID | DEPARTMENT_NAME | MANAGER_ID | LOCATION_ID |
|---|---|---|---|
| 10 | Administration | 200 | 1700 |
| 20 | Marketing | 201 | 1800 |
| 50 | Shipping | 124 | 1500 |
| 60 | IT | 103 | 1400 |
| 80 | Sales | 149 | 2500 |
| 90 | Executive | 100 | 1700 |
| 110 | Accounting | 205 | 1700 |
| 190 | Contracting | | 1700 |

Primary key        Foreign key        Primary key

EMPLOY FIRST_NAME,LAST_NAME,DEPARTMENT_ID

# Functional Dependencies

Functional Dependencies (FD):- Generalizes the concept of a key. It can also be thought of as a constraint between two sets of attributes. In other words, FD is a constraint that is to be satisfied by every desirable relation instance.

STUDENT

| | |
|---|---|
| 1 | AMIT |
| 2 | SUMIT |

| | |
|---|---|
| 1 | AMIT |
| 1 | AMIT |

| | |
|---|---|
| 1 | AMIT |
| 1 | SUMIT |

| | |
|---|---|
| 1 | AMIT |
| 2 | AMIT |

Let r be a relations schema and let X and Y be non-empty sets of attributes in r. We say that for a relation instance R of the schema r has a **Functional Dependency X → Y** if the following conditions holds for every pairs of tuples t1 and t2 in R.

If t1 [X] = t2 [X] then t1 [Y] = t2 [Y]

By considering X→ Y, we say that X functionally determines Y, or Y is functionally dependent on X.

To illustrate the concept of FD, let us consider the example. Let –

X = {Managerid, Manager Name} and Y = {ManagerAge}

Tuple t1 and t2 having same Managerid value and same Manager Name, we must have same ManagerAge. Else the data is inconsistent. Thus we can say that –

{Managerid, ManagerName} → {ManagerAge}

| A | B | C | D |
|---|---|---|---|
| a1 | b1 | c1 | d1 |
| a1 | b2 | c1 | d2 |
| a2 | b2 | c2 | d2 |
| a2 | b3 | c2 | d3 |
| a3 | b3 | c2 | d4 |

BOX

Figure: Sample relation r.

# Inference Rules for FDs

- The following are the inference rules to generate new FDs from a given set of FDs. For the rules described below, we assume X, Y and Z are the subset of attributes in the schema r.

| | |
|---|---|
| 1. Reflexive Rule | If $X \supseteq Y$, then $X \rightarrow Y$ |
| 2. Augmentation | If $X \rightarrow Y$ then $XZ \rightarrow YZ$ |
| 3. Transitive Rule | If $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$ |
| 4. Decomposition | If $X \rightarrow YZ$ then $X \rightarrow Y$, and $X \rightarrow Z$ |
| 5. Union | If $X \rightarrow Y$ and $X \rightarrow Z$, then $X \rightarrow YZ$ |