Devindu Samarasinghe

IT2040 – Data Management System

Lecture 3 – Data Schema refinement (Week 4)

Table

Description automatically generatedSchema refinement handles few anomalies

Insertion anomaly

* + Suppose a situation where one table has information of both department and employee in one row, and the table should insert a newly made department that does not have employees yet. In this situation you cannot create a dept because primary key is Eid, and Eid means employee should be present plus primary key cannot be null
  + If you add a new employee, must retype the department details and it will be redundant and repeated
* Update anomaly
  + Updating of information of dept location will result in changing everyone’s location one by one.
* Deletion anomaly
  + Deleting the last emp of the department will delete the entire department.

**Solution – Decomposing the relations to smaller relations**

* Loss – less join property
  + Property enables recovery of original relation from smaller relations without loss of data

Suppose a table,

|  |  |
| --- | --- |
| S | P |
| S1 | P1 |
| S2 | P2 |
| S3 | P1 |

|  |  |
| --- | --- |
| P | D |
| P1 | D1 |
| P2 | D2 |
| P1 | D3 |

When merging these together, we use the common column to merge these two tables

|  |  |  |
| --- | --- | --- |
| S | P | D |
| S1 | P1 | D1 |
| S1 | P1 | D3 |
| S2 | P2 | D2 |
| S3 | P1 | D1 |
| S3 | P1 | D3 |

Is this decomposition Loss- Less? It is not, because the common column is not unique, and this creates anomalies

THEREFORE, THIS Method has anomalies

* Dependency preserving property

Schema Refinement – Systematic process for analyzing relational schema to minimize redundancies and minimizing insertion, deletion and updating anomalies

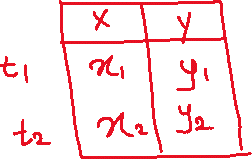
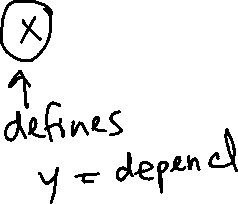
Normalization is based on functional Dependencies

* FD is when one attribute uniquely determines another attribute
* Redundancies in relations are based on functional dependencies

Activity 1

* Consider a vehicle entity. With respect to attributes, you know a vehicle holds, write three functional dependencies that can exist in a vehicle table
  + - Reg No -> Color
    - Reg No -> Owner
    - Reg No -> Model
    - Reg No -> Brand

**Mathematical Approach to show the functional dependency of attributes**



Keys and FDs

**Key constraint** – attributes in the key plays role X(determinant) and the set of all attributes play role Y(dependent)

X+ means all attributes are dependent on X

**Armstrong Axioms**

1. **Reflexivity** = X -> X
2. **Augmentation** = If X -> Y, then XZ -> YZ for any Z (Meaning Z attribute can be added to both X and Y)
3. **Transitivity** = If X -> Y and Y-> X, then X -> Z

Additional Rules

1. **Union** = If X->Y and X->Z, then X -> YZ
2. **Decomposition** = X ->YZ, then X->Y and X->Z

Example: Consider a relation R (A, B, C, D) with the following set of functional dependencies over R

F = {A -> B, B -> C, B-> D}

Attribute Closure = A+

A -> A (reflectivity)

A -> B (given)

A -> B, B -> C = A -> C (Transitivity)

A -> B, B -> D = A -> D (Transitivity)

A -> ABCD

**Therefore, A is a key**

Find,

1. B+ = {BCD}

B -> B(reflectivity)

B -> C(given)

B -> D(given)

1. C+ = {C}

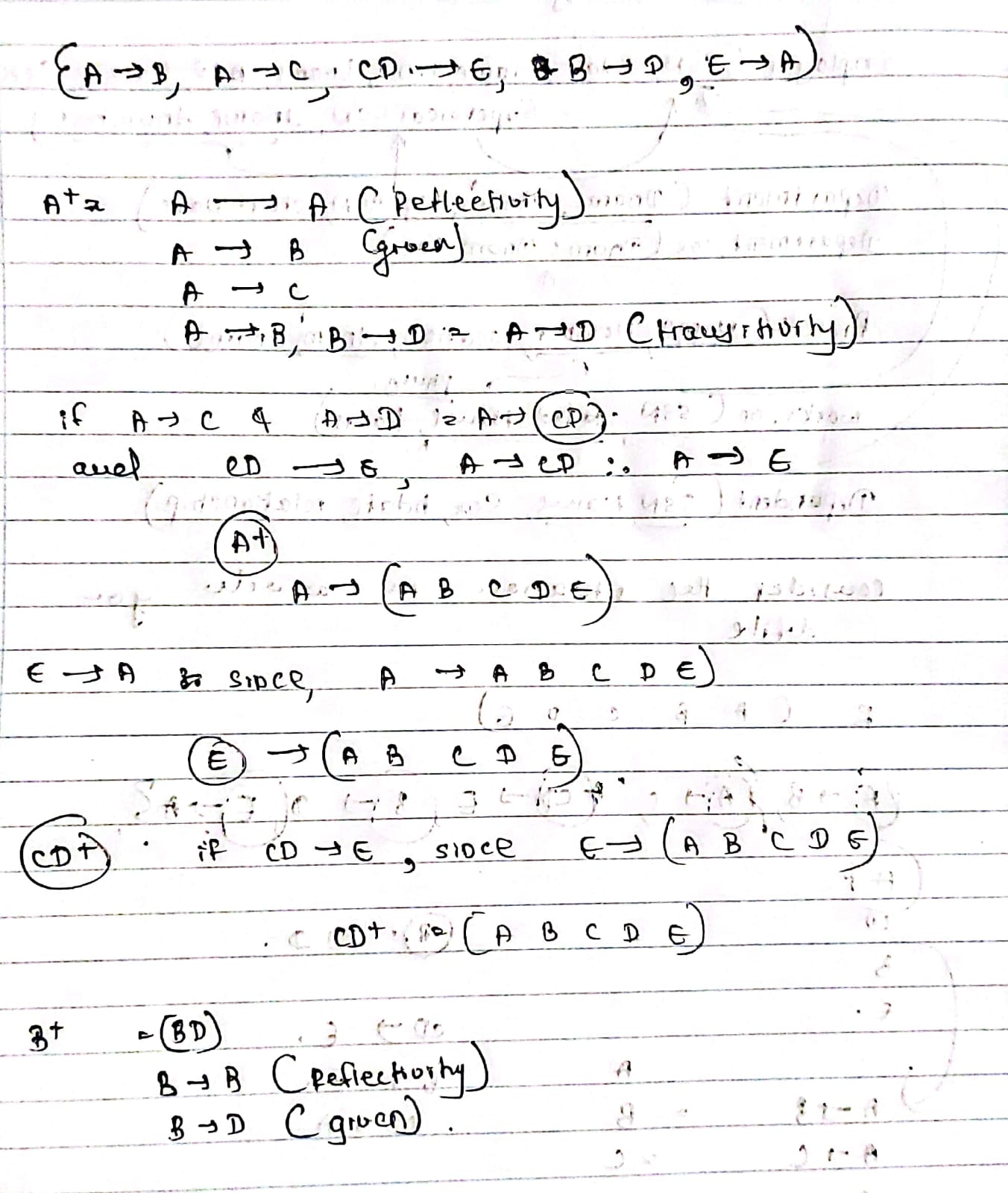
C -> C (Reflectivity)

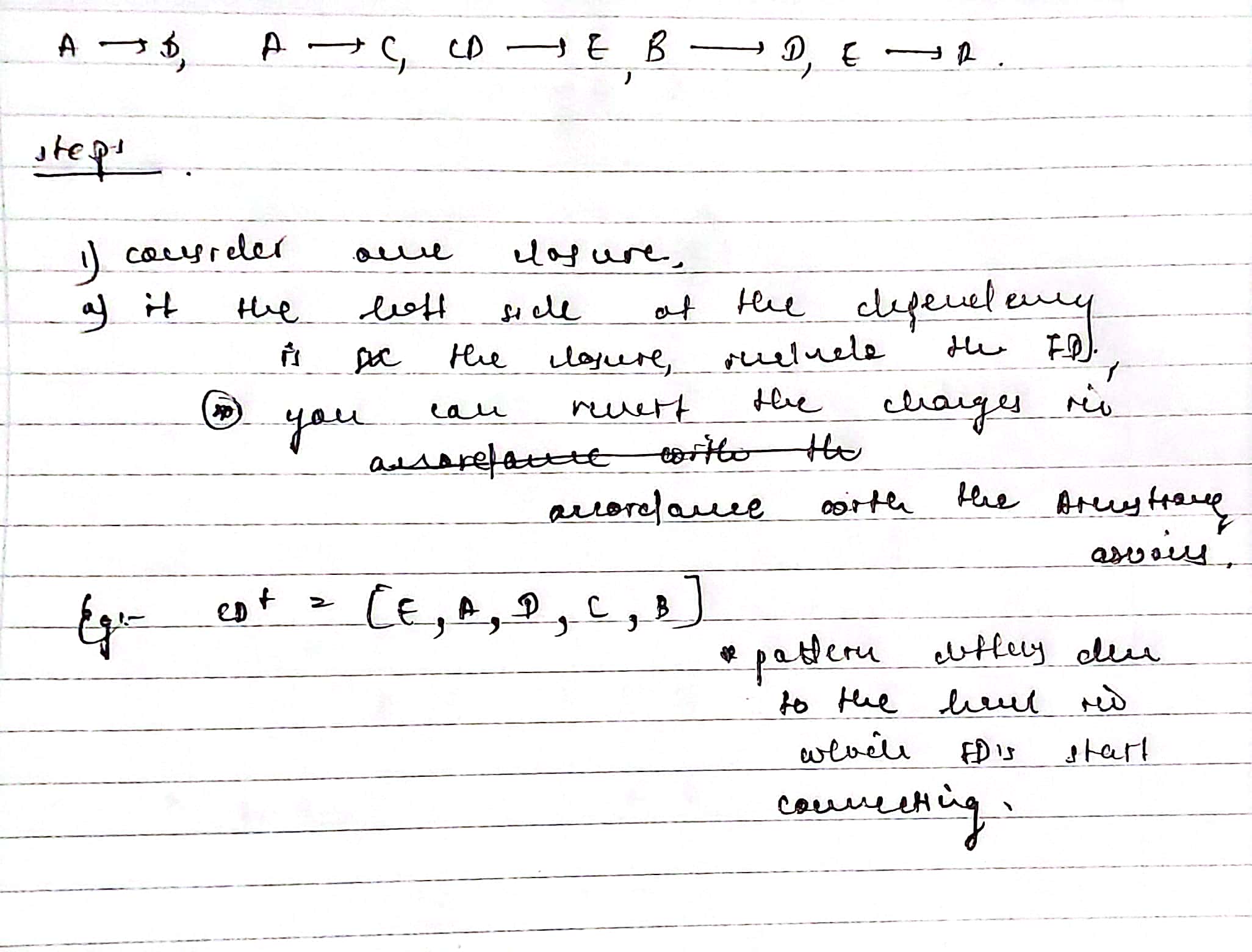
1. D+ = {D}

D -> D (reflectivity)

Consider a relation R (A B C D E) with the following set of functional dependencies over R:

{A -> B, A -> C, CD -> E, B -> D, E -> A}



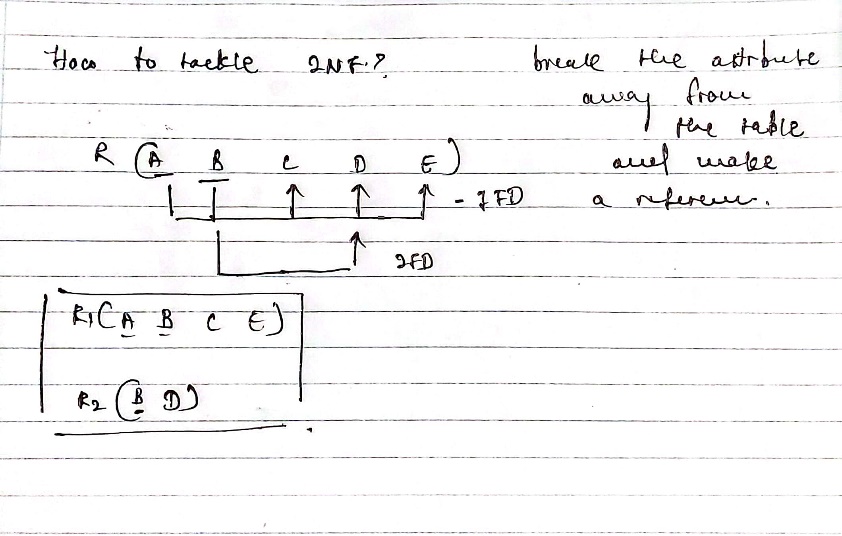


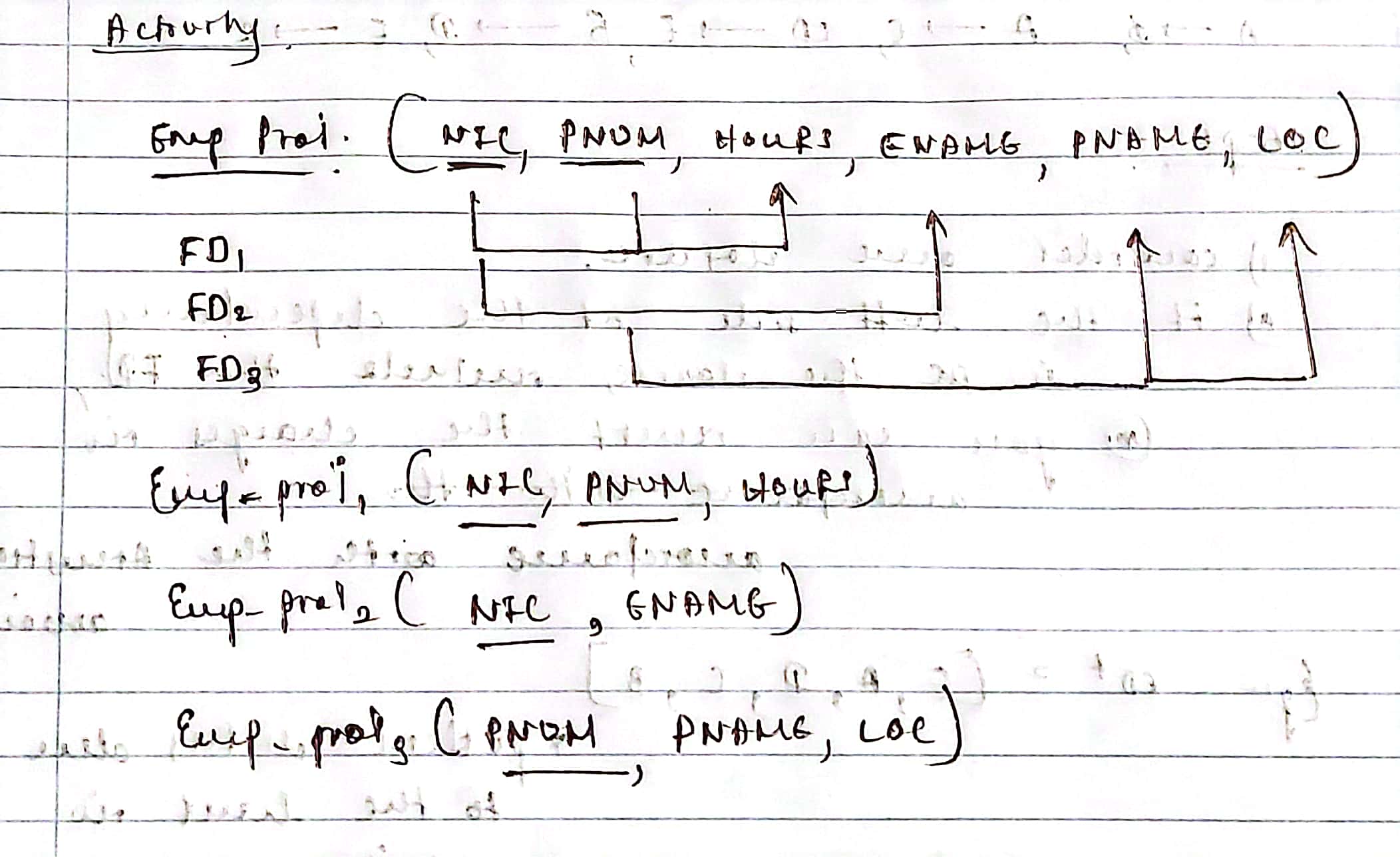
**Normal Forms**

**1NF** should eliminate Multivalued attributes, composite attributes and break them to another table

**2NF** should eliminate partial dependencies.

What is a partial dependency?

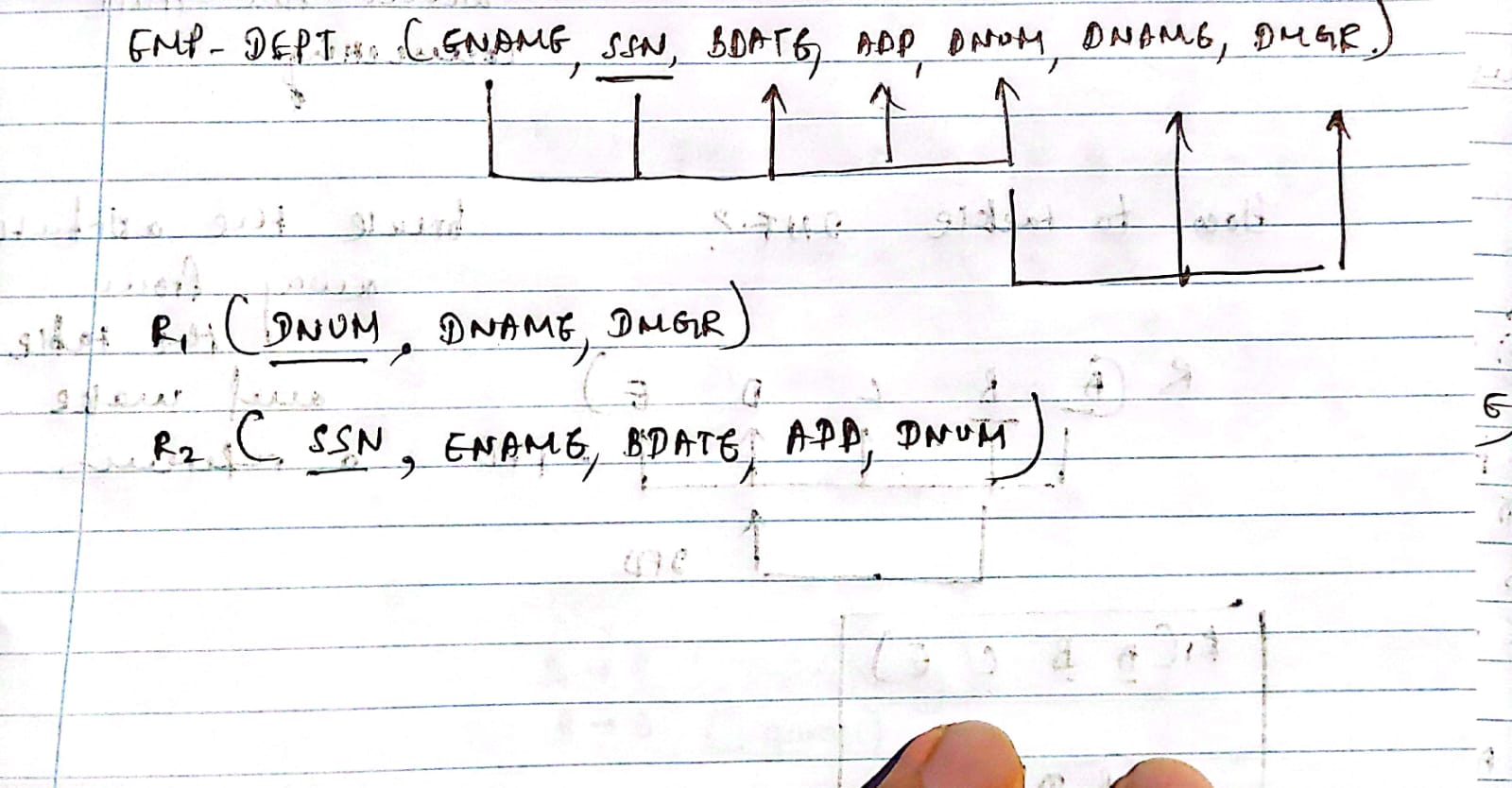
* Partial dependency is when one part of the primary key(when the pk is a composite key) is only a determinant of another attribute
* 

Activity

**3NF** is when the relation is in 2NF and has transitive dependent keys that are to be removed

Example: A B C D E R1 - A B C E



 R2 - C D



Boyce Codd Normal Form

If X -> A, then X is a superkey of R

A picture containing graphical user interface

Description automatically generated