

11-9-25

dependencies upto BCNF

Task-8: Normalization databases using functional dependencies
upon relational tables created in task-2, perform normalization upto BCNF based on given dependencies as following:

Employee Database:

① Identify employee attributes: Employee-ID, Name, Department, Job-Title, Manager-ID, Hire-Date, Salary

② Define relational schema: Employee

③ Determine Functional dependencies (FDs) between attributes

- Employee-ID \rightarrow Name, Department, Job-Title, Manager-ID, Hire-Date, Salary
- Department \rightarrow Manager-ID
- Manager-ID \rightarrow Name

Step 2: Convert to 1NF

- ① Eliminate repeating groups or arrays
- ② Create separate tables for each repeating group

Step 3: Convert to 2NF

- ① Ensure each non-key attribute depends on the entire prime key
- ② Move non-key attributes to separate tables if they depend on only part of the primary key

- Create Department table: Department(Department-ID, Manager-ID, Name)
- Create Employee table: Employee(Employee-ID, Name, Dept-ID, Job-Title, Salary, Hire-Date)

Step 4: Convert to 3NF

- ① Ensure there are no transitive dependencies
- ② Move non-key attributes to separate tables if they depend on another non-key attribute
- Create Manager table: Manager(Manager-ID, Name)
- Update Department table: Department(Department-ID, Manager-ID)

Step 5: Convert to BCNF

- ① Ensure every determinant is a candidate key.
- ② Check for overlapping candidate keys
- ③ Decompose relations to eliminate redundancy
- No further decomposition needed

using Griffith tool

- ① Input relational schema and functional dependencies.
- ② Griffith tool generates a dependency graph.
- ③ Analyze the graph to identify normalization issues.
- ④ Apply normalization rules to transform the schema
- ⑤ Verify the resulting schema meets BCNF criteria.

Griffith tool Steps

- ① Create a new project in Griffith
- ② Define the relational schema and FDs
- ③ Run the "Dependency Graph" tool.
- ④ Analyze the graph for normalization issues.
- ⑤ Apply transformations using the "Normalize" tool.
- ⑥ Verify BCNF compliance using the "BCNF check" tool.

11-9-25

Task-8

FUNCTIONAL DEPENDENCY :

Attributes in Table

① Separate attributes using a comma (,)

employee_id, name, department, job_title, manager_id, hire_date, salary

Functional Dependencies

employee_id *



name * department *

Delete

job_title * hire_date *

manager_id * salary *

Add Another Dependency

NORMAL FORM :

Check Normal Form



2NF

The table is in 2NF



3NF

The table is in 3NF



BCNF

The table is in BCNF

Show Steps



2NF

find all candidate keys. The candidates keys are { employee_id }. The set of key attributes are: { employee_id }
 for each non-trivial FD, check whether the LHS is a proper subset of some candidate key or the RHS are not
 all key attributes
 checking FD: employee_id → name, department, job_title, hire_date, manager_id, salary

3NF

find all candidate keys. The candidates keys are { employee_id }. The set of key attributes are: { employee_id }
 for each FD, check whether the LHS is superkey or the RHS are all key attributes
 checking functional dependency employee_id → name, department, job_title, hire_date, manager_id, salary

BCNF

A table is in BCNF if and only if for every non-trivial FD, the LHS is a superkey.

CONVERT 2NF :

Normalize to 2NF

Attributes

employee_id name department job_title manager_id hire_date salary

Functional Dependencies

employee_id name department job_title hire_date manager_id salary

Show Steps

First, find the minimal cover of the FDs, which includes the FDs
employee_id \rightarrow name
employee_id \rightarrow department
employee_id \rightarrow job_title
employee_id \rightarrow hire_date
employee_id \rightarrow manager_id
employee_id \rightarrow salary

Initially rel[1] is the original table

Round1: checking table rel[1]

**** The table is in 2NF already, send it to output ****

CONVERT 3NF :

1NF to 3NF

Attributes

employee_id name department job_title manager_id hire_date salary

Functional Dependencies

employee_id name

employee_id department

employee_id job_title

employee_id hire_date

employee_id manager_id

employee_id salary

Show Steps

Table already in 3NF

CONVERT BCNF :

Normalize to BCNF

Attributes

employee_id name department job_title manager_id hire_date salary

Functional Dependencies

employee_id → name department job_title hire_date manager_id salary

Show Steps

Take already in 3NF, convert to 3NF

VEL TECH	
PERFORMANCE (5)	5
REVIEW AND ANALYSIS (5)	5
ANALYSIS (5)	5
RECORD (5)	5
TOTAL (20)	20
WITH DATE	

18.7.25

Normalized Schema

- ① Employee (Employee_id, Name, Department_id, Job_title, Hire_date, Salary)
- ② Department (Department_id, Manager_id)
- ③ Manager (Manager_id, Name)

VSL TECH	
1 NO	✓
PERFORMANCE (5)	✓
REVIEW AND ANALYSIS (5)	✓
VIVA VOCE (5)	✓
RECORD (5)	✓
TOTAL (25)	✓
2024 WITH DATE	✓

Result: Thus the Normalization database using functional dependencies upto BCNF has been executed successfully.