CPS510 Fall2021 Section 04

Group-11

Assignment 8

Normalization / BCNF

Conversion of 3NF to BCNF Normalize form by using Algorithm

Application Name: Online Job Bank System.

Group Member names:

1. Tusaif Azmat
   Student#: 500660278
2. Ankit Dheedsa
   Student#: 500975118
3. Mahdi Alam
   Student#: 500969935

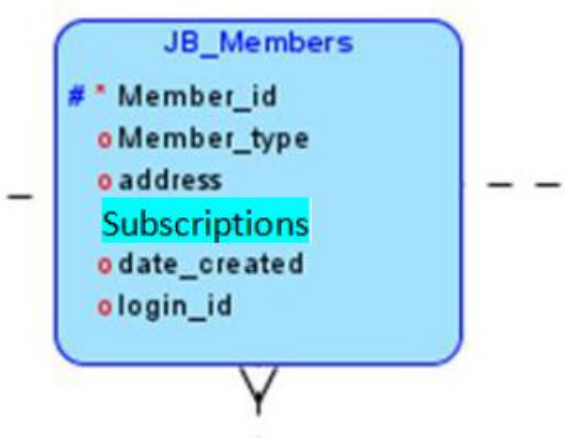
-- Normalization/ Conversion of 3NF to BCNF with the help of Algorithm.
--- We used Bernstein's Algorithm to achieve this task:

Bernstein's Algorithm - Broken down into 4 steps:

1) **Determine all the functional dependencies**
2) **a) Find and remove redundancies**
   b) Find and remove partial dependencies
3) **Find keys**
4) **Create tables**

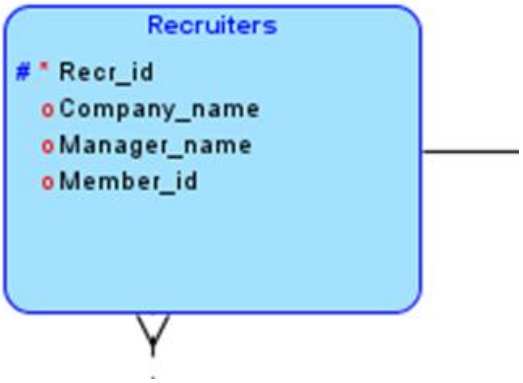
Step 1 (finding all functional dependencies)

- **Member_id** → {Member_type, address, date_created, Subscriptions}

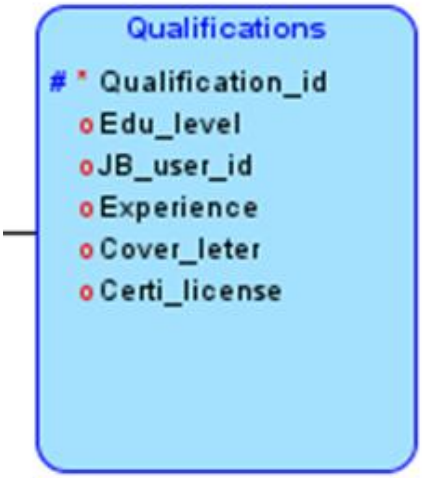


- **Login_id** → { password}

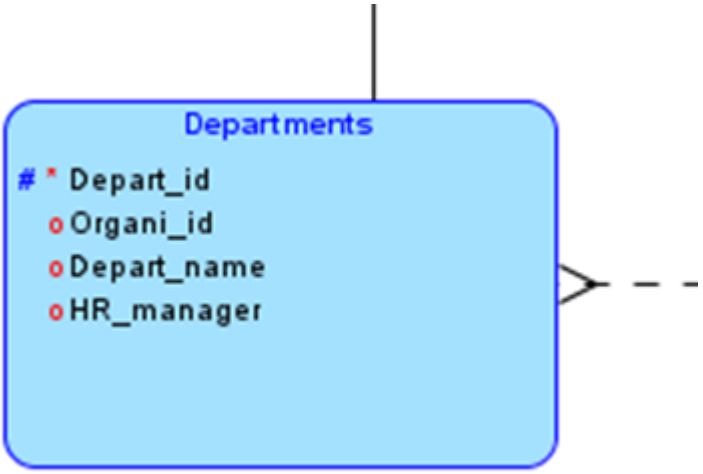
- **Recr_id** → {company_name, Manager_name}



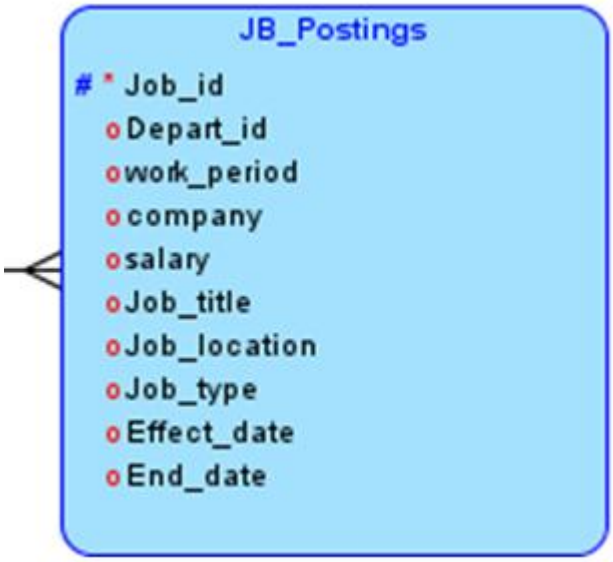
- **Qualification_id**→ {Edu_level, Experience, cover_letter, Certi_license}



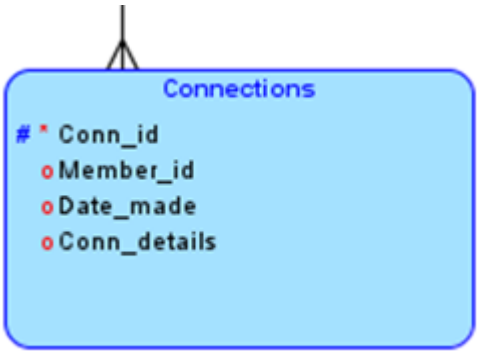
- **depart_id** → {depart_name, HR_manager}



- **job_id** → {work_period, company, salary, job_title, job_location, job_type,Effect_date, End_date}



- **Conn_id** → {Date_made, Conn_details}



Step 2a (Break RHS and find redundancies)

Get rid of Redundancies

- **Member_id** → {Member_type, address, date_created, Subscriptions}
  - Reduced list of FD's:
  - **Member_id**→ {Member_type}
  - **Member_id** →{address}
  - **Member_id** →{date_created}
  - **Member_id** →{Subscripts}
  - **No redundancies**

- **Login_id** → { password}

    - Reduced list of FD's:
    - **Login_id** -> {password}
    - **No redundancies**

- **Recr_id** → {company_name, Manager_name}
    - Reduced list of FD's:
    - **Recr_id** → {company_name}
    - **Recr_id** → {Manger_name}
    - **No redundancies**

- **Qualification_id →** {Edu_level, experience, cover_letter, Certi_license}
    - Reduced list of FD's:
    - **Qualification_id** → {user_degree}
    - **Qualification_id** →{ experience}
    - **Qualification_id** →{cover_letter}
    - **Qualification_id** →{Certi_license}
    - **No redundancies**

- **depart_id →** {depart_name, HR_manager}
    - Reduced list of FD's:
    - **depart_id** → {depart_name}
    - **depart_id** → {HR_manager}
    - **No redundancies**

- **job_id** → {work_period, company, salary, job_title, job_location, job_type,Effect_date, End_date}
    - Reduced list of FD's:
    - **job_id** →{work_period}
    - **job_id** →{company}
    - **job_id** → {salary}
    - **job_id** → {job_title}
    - **job_id** → {job_location}
    - **job_id** →{job_type}
    - **job_id** →{Effect_date}
    - **job_id** →{End_date}
    - **No redundancies**

- **Conn_id** → {money}
    - Reduced list of FD's:
    - **conn_id →** {Date_made}
    - **conn_id →** {conn_details}
    - **No redundancies**

Step 2b (Minimize LHS, find and remove partial dependencies)

- LHS is already minimized, therefore there are no partial dependencies

Step 3(Find keys) (relational schema)

- **Member_id→** {Member_type, address, date_created, subscriptions}
    - Attributes on RHS but not on LHS (cannot be keys)
        - Member_type
        - Address
        - Date_created
        - Subscriptions
    - Possible Keys
        - Member_id

- **Login_id** → { password}
    - Attributes on RHS but not on LHS (cannot be keys)
        - password
    - Possible Keys
        - Login_id

- **Recr_id** → {company_name, manager_name}
    - Attributes on RHS but not on LHS (cannot be keys)
        - Company_name
        - Manger_name
    - Possible Keys
        - Recr_id

- **depart_id** → {depart_name, HR_manager}
    - Attributes on RHS but not on LHS (cannot be keys)
        - depart_name
        - HR_Manger
    - Possible Keys
        - depart_id

- **qualification_id** → {Edu_level, experience, cover_letter, Certi_license}
    - Attributes on RHS but not on LHS (cannot be keys)
        - Edu_level
        - Experience
        - Cover_letter
        - Certi_license
    - Possible Keys
        - Qualification_id

- **job_id** → {work_period, company, salary, job_title, job_location, job_type,Effect_date, End_date}
    - Attributes on RHS but not on LHS (cannot be keys)
        - work_period
        - company
        - salary
        - job_title
        - job_location
        - job_type
        - Effect_date
        - End_date
    - Possible Keys
        - job_id

- **Conn_id** → {Date_made, Conn_details}
    - Attributes on RHS but not on LHS (cannot be keys)
        - Date_made
        - Conn_details
    - Possible Keys
        - Conn_id

Step 4(Make tables)

R1 (**member_id**, member_type, address, date_created, subscriptions, login_id)
With
FD: **member_id** → { member_type, address, date_created , subscriptions }

R2 (**Login_id**, password)
With
FD: **user_id** →{ user_password}

R3 (**recr_id**, company_name, manager_name, member_id)
With
FD: **recr_id** →{company_name, manager_name }

R4 (**depart_id**, depart_name, hr_manager,Org_id)
With
FD: **depart_id** →{ depart_name, hr_manager }

R5 (**qualification_id**, edu_level, experience, cover_letter, Certi_license, user_id)
With
FD: **qualification_id** → { edu_level, experience, cover_letter, Certi_license }

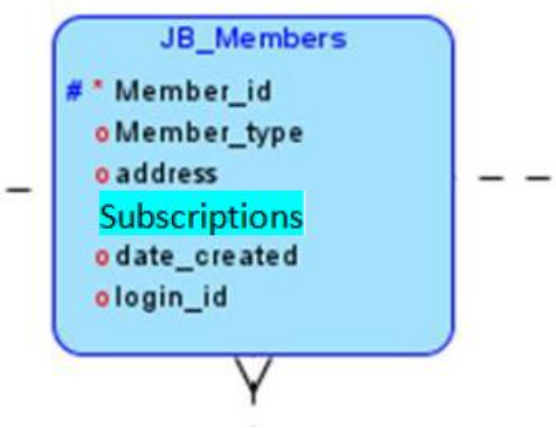
R6 (**job_id**, work_period, company, salary, job_title, job_location, job_type, effect_date,end_date, depart_id)
With
FD: **job_id** →{, work_period, company, salary, job_title, job_location, job_type, effect_date,end_date }

R7 (**conn_id**, date_made, conn_details, member_id)
With
FD: **conn_id** → { date_made, conn_details }

BCNF (Boyce/Codd Normal Form)

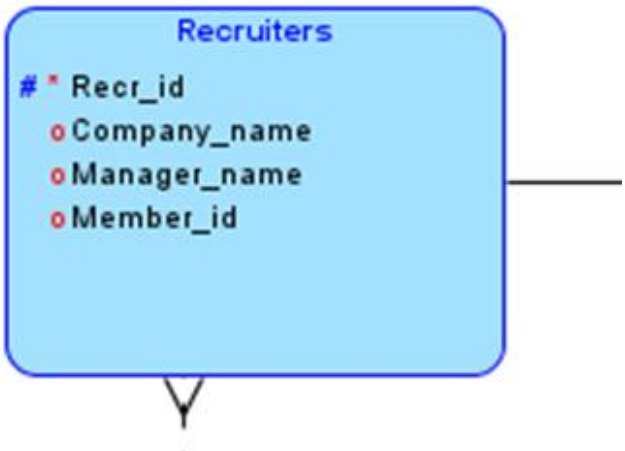
Step 1 (finding all functional dependencies (List of all attributes and FDs))

- **member_id** → { member_type, address, date_created, subscriptions, login_id }

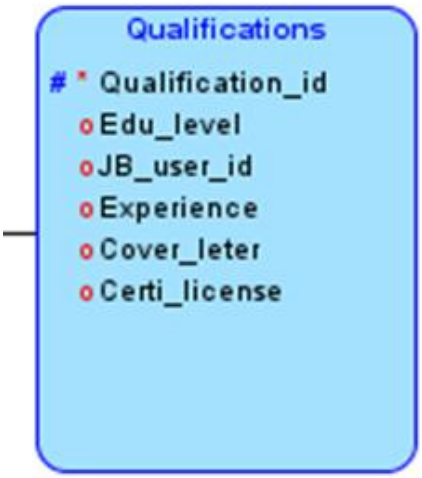


- **Login_id** → { password}

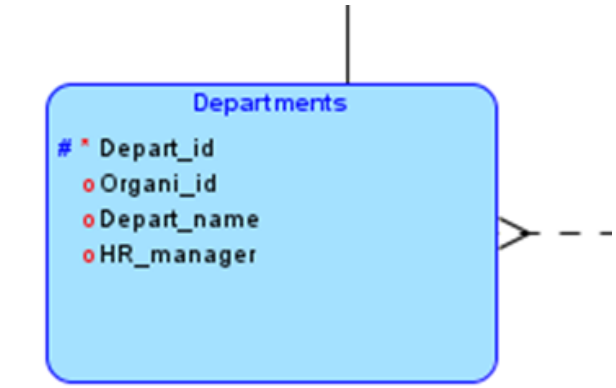
- **Recr_id** → {company_name, Manager_name, member_id}



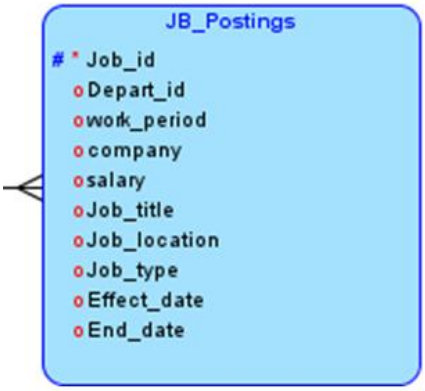
- **Qualification_id→** {Edu_level, Experience, cover_letter, Certi_license, jb_user_id}



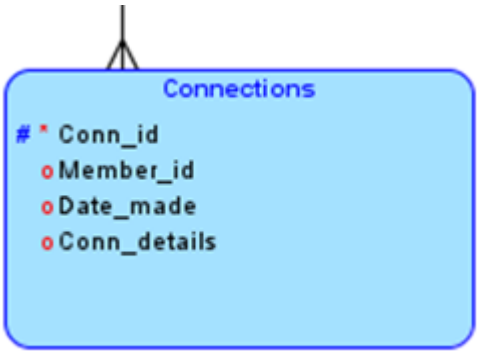
- **depart_id** → {depart name, HR_manager, Organi_id}



- **job_id** → {work_period, company, salary, job_title, job_location, job_type,Effect_date, End_date, depart_id, Depart_id}



- **Conn_id** → {Date_made, Conn_details, member_id}



Step 2 (make sure that the left hand side are keys, if not decompose)

Consider the relation schema
R1 (**member_id**, member_type, address, date_created, subscriptions, login_id)
With FD
member_id → { member_type, address, date_created , subscriptions }
This schema has one candidate key
member_id
Therefore this schema is in BCNF

Consider the relation schema
R2 (**Login_id**, password)
With FD
Login_id → {password}
This schema has one candidate key
login_id
Therefore this schema is in BCNF

Consider the relation schema
R3 (**recr_id**, company_name, manager_name, member_id)
With FD
recr_id → {company_name, manager_name }
This schema has one candidate key
recr_id
Therefore this schema is in BCNF

Consider the relation schema
R4 (**depart_id**, depart_name, hr_manager,Org_id)
With FD
depart_id → { depart_name, hr_manager }
This schema has one candidate key
depart_id

Therefore this schema is in BCNF

Consider the relation schema
R5 (**qualification_id**, edu_level, experience, cover_letter, Certi_license, user_id)
With FD
qualification_id → { edu_level, experience, cover_letter, Certi_license }
This schema has one candidate key
qualification_id
Therefore this schema is in BCNF

Consider the relation schema
R6 (**job_id**, work_period, company, salary, job_title, job_location, job_type, effect_date,end_date, depart_id)
With FD
job_id → { work_period, company, salary, job_title, job_location, job_type,effect_date,end_date}
This schema has one candidate key
job_id
Therefore this schema is in BCNF

Consider the relation schema
R7 (**conn_id**, date_made, conn_details, member_id)
With FD
conn_id → { date_made, conn_details }
This schema has one candidate key
conn_id
Therefore this schema is in BCNF

Step 3 (final BCNF schema for R)

R1 (**member_id**, member_type, address, date_created, subscriptions, login_id)
R2 (**Login_id**, password)
R3 (**recr_id**, company_name, manager_name, member_id)
R4 (**depart_id**, depart_name, hr_manager,Org_id)
R5 (**qualification_id**, edu_level, experience, cover_letter, Certi_license, user_id)
R6 (**job_id**, work_period, company, salary, job_title, job_location, job_type, effect_date,end_date, depart_id)
R7 (**conn_id**, date_made, conn_details, member_id)

Note: All the tables are of the form BCNF and we didn't need to combine any tables for this stage.