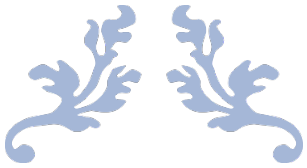

THE DATABASE

Hasaranga Jayathilake

Contents

Table of Figures

Table of Tables

Data Book: Part 1: Database Narrative – Mentoring System Database

A description of the database

Overview – Mentoring System

Mentoring is considered one of the methods that can be used to improve the knowledge level of the employees with less cost accumulation for the organization while improving the bond among employees (Clark, 2017; Orlova, 2021). Even it will facilitate employees to know more about their coworkers and their potential (Ng, 2022). Also, mentoring will help to improve productivity and accuracy (Adil et al., 2021).

When comes to the software development industry, mentoring is usually used as the mechanism for the increased knowledge level of the employees to meet the knowledge level required to complete the projects in a very agile business setting (Eriksson et al., 2022; Ng, 2022). Since this method facilitates fast-track, the knowledge transferred to the employees from their co-workers in a shorter time it may keep the organization more competitive shape to meet the demands of the customers (Ge et al., 2010; Orlova, 2021). Therefore, mentoring would help to create a business ecosystem with a sustainable competitive advantage for the organization (Eriksson et al., 2022).

Primary objectives of the database

Having a mentoring system will be able to monitor the progress of the employees as well as the effectiveness of the mentoring programs that are operating at the organization. Moreover, the system will enable to the connection of the mentoring progress with employee reward systems as it will help to attract more mentors from the organization to enhance the knowledge levels of the other employees within the organization (Eriksson et al., 2022).

- To increase the productivity and effectiveness of the mentoring process.
- To enhance the connectivity (closeness) between the mentors and mentees.

- To track the progress of the mentoring programs.
- To connect the mentee program work to the employee reward system for mentor attraction and retention.

Users of the database and their expectations (needs) Mentoring system

In this mentoring database, the primary target audience is the employees who are working in the software development industry. A-line with the primary duties and the expectations of the software development industry employees, this database is going to cater to the following user expectations as

Table 1: User Primary job roles and the user expectations from the mentoring database

User	Primary Duty as an employee in the software development industry	User Expectations from Mentoring Database
Mentees	1) Code or develop the software based on the customer requirements by utilizing new technologies available. 2) Reduce the bugs and re-works on the work that they perform 3) Reduce the software development pipeline to beat the competition 4) Increase their performance to increase their benefits and the package from the organization where they work	1) Get quick help from co-workers on the subject knowledge matter that is triggered while working on the projects 2) Quickly enhance their subject knowledge to complete the assigned tasks for them accurately. 3) Enhance the knowledge of the new technologies to be relevant to the present technology ecosystem. 4) To have time for networking with their peers and getting to know more about them by increasing interactions with them.
Mentors	1) Showcase their ability for their co-workers and the management to new	1) To understand the performance level and the progress level of the mentees and improve them,

	responsibilities to get career succession by being in the organization. 2) To get more rewards for them like pay raises during their tenure.	2) Resolve the issues faced by the mentees. 3) To interact more easily with mentees 4) To track the level of the mentoring programs and their effect on their rewards from the organization.
Management	1) Enhance the organizational performance to achieve sustainable competitive advantage over the competitors 2) Enhance customer satisfaction to retain customers for longer terms to get more projects from the customers	1) To identify the new talented employees 2) To increase the productivity of the employees 3) To find out the outstanding employees who have career success in the workplace by giving new projects 4) Understand the knowledge level and competencies of the employees

Developed by Author.

Data dictionary

The data dictionary that has been illustrated from the below Table 2, shows the all the entities that are using for creating the database for the mentoring system. Coronel & Morris, 2017, (Pg. 92), table illustration has been used to develop the titles of the below table's column titles. Importantly, for the data type of the below table used the SQL Server Data Types convention since the final database creation is going to be completed via mySQL.

Important: Since, some cases organization recruit employees, even without integrating into the any department or and transfer into the different department. Therefore, at the initial stage of the data entry they can have the employees who are not assigned for any department under the organization So that, EMPLOYEE_REGISTRY.DEPT_CODE attribute is not kept as required.

Table 2: Data Dictionary of Mentoring System

Table Name	**Attribute Name**	**Contents**	**Type**	**Format**	**Range**	**Required**	**PK or FK**	**FK Referenced Table**
EMPLOYEE_ REGISTRY	EMPLOYEE_ID	Employee Identity number	VARCHAR(5)	Xxxxx		Y	PK	

	DEPT_CODE	Department code	VARCHAR(5)	Xxxxx			FK	ORG_DEPAT
	EMP_FNAME	Employee first name	VARCHAR(20)	Xxxxxxxx		Y		
	EMP_LNAME	Employee last name	VARCHAR(20)	Xxxxxxxx		Y		
	EMP_INTIAL	Employee initial	CHAR(1)	X				
	EMP_ST_ADD	Employee street address	VARCHAR(20)	Xxxxxxxx		Y		
	EMP_APT_NUM	Employee apartment number	VARCHAR(6)	Xxxxxx				
	EMP_CITY	Employee city	VARCHAR(20)	Xxxxxxxx		Y		
	EMP_STATE	Employee State	VARCHAR(2)	Xxxxxxxx		Y		

	EMP_ZIP_CODE	Employee zip code	VARCHAR(5)	Xxxxx		Y		
	EMP_EMAIL	Employee email	VARCHAR(45)	Xxxxxxxx		Y		
	EMP_P_NUM	Employee phone number	VARCHAR(10)	Xxxxxxxx		Y		
	EMP_JOIN_DATE	Employee joined date	DATE	YYYY-MM-DD	'1000-01-01' to '9999-12-31'	Y		
	EMP_MENTOR_STATUS	Employee mentor registration status	VARCHAR(3)	Xxx		Y		
	EMP_MENTEE_STATUS	Employee mentee registration status	VARCHAR(3)	Xxx		Y		

ORG_DEPAT	DEPT_CODE	Department code	VARCHAR(5)	Xxxxx		Y	PK	
	DEPT_NAME	Department Name	VARCHAR(20)	Xxxxx xxx		Y		
	DEPT_EMAIL	Department email	VARCHAR(20)	Xxxxx xxx		Y		
	DEPT_TEL	Department phone number	VARCHAR(10)	Xxxxx xxx		Y		
	DEPT_ST_ADD	Department street address	VARCHAR(20)	Xxxxx xxx		Y		
	DEPT_BULD_NUM	Department building number	VARCHAR(6)	Xxxxx x		Y		
	DEPT_CITY	Department city	VARCHAR(20)	Xxxxx xxx		Y		
	DEPT_STATE	Department State	VARCHAR(2)	Xxxxx xxx		Y		

	DEPT_ZIP_CODE	Department zip code	VARCHAR(5)	Xxxxx		Y		
ORG_DEPAT_ROOM_ASSC	DEPT_CODE	Department code	VARCHAR(5)	Xxxxx		Y	PK&FK	ORG_DEPAT
	ROOM_CODE	Room code	VARCHAR(5)	Xxxxx		Y	PK&FK	ROOM_REGISTRY
BULD_REGISTRY	BLD_CODE	Building code	VARCHAR(5)	Xxxxx		Y	PK	
	BULD_CODINA	Building Location Name	VARCHAR(45)	Xxxxx		Y		
	NUM_FLOORS	Number of floors in a building	CHAR(2)	99	00–99	Y		

	BULD_NAME	Building Name	VARCHAR(20)	Xxxxxxxx		Y		
	BULD_OPEN_DATE	Building open date	DATE	YYYY-MM-DD	'1000-01-01' to '9999-12-31'	Y		
ROOM_REGISTRY	ROOM_CODE	Room code	VARCHAR(5)	Xxxxx		Y	PK	
	BLD_CODE	Building code	VARCHAR(5)	Xxxxx		Y	FK	BULD_REGISTRY
	ROOM_CAP	Room capacity	SMALLINT(3)	000 - 999		Y		
	ROOM_TEL_NUM	Room telephone number	VARCHAR(10)	Xxxxxxxx		Y		
	ROOM_BUL_NUM	number of bulbs at room	SMALLINT(3)	000 - 999		Y		
MENTOR	MENTOR_ID	Mentor id number	INT(5)	99999	00000 - 99999	Y	PK	

	EMPLOYEE_ID	Employee Identity number	VARCHAR(5)	Xxxxx		Y	FK	EMPLOYEE_REGISTY
	REG_DATE	Registered date	DATE	YYYY-MM-DD	'1000-01-01' to '9999-12-31'			
	MENTOR_SUB_1	Mentoring subject one	VARCHAR(10)	Xxxxxxxx				
	MENTOR_SUB_2	Mentoring subject two	VARCHAR(10)	Xxxxxxxx				
MENTEE	MENTEE_ID	Mentee id number	INT(5)	99999	00000 - 99999	Y	PK	
	EMPLOYEE_ID	Employee Identity number	VARCHAR(5)	Xxxxx		Y	FK	EMPLOYEE_REGISTY

	REQ_DATE	Requested date	DATE	YYYY-MM-DD	'1000-01-01' to '9999-12-31'			
	SUB_1REQ	First subject requested	VARCHAR(10)	Xxxxxxxx				
	SUB_2REQ	Second subject requested	VARCHAR(10)	Xxxxxxxx				
MENTOR_PROG_ASSC	PROG_ID	Program id	VARCHAR(5)	Xxxxx		Y	PK&FK	PROG_CONDUCTED
	MENTOR_ID	Mentor id number	INT(5)	99999	00000-99999	Y	PK&	MENTOR

							FK	
MENTEE_PROG_ASSC	PROG_ID	Program id	VARCHAR(5)	Xxxxx		Y	PK & FK	PROG_CONDUCTED
	MENTEE_ID	Mentee id number	INT(5)	99999	00000 - 99999	Y	PK & FK	MENTEE
PROG_CONDUCTED	PROG_ID	Program id	VARCHAR(5)	Xxxxx		Y	PK	
	PROG_NAME	Program name	VARCHAR(20)	Xxxxx xxx		Y		
	PROG_DESCP	Program description	VARCHAR(45)	Xxxxx xxx		Y		
	PROG_MODE	Program mode	VARCHAR(10)	Xxxxx xxx		Y		

	PROG_CHAPTERS	Program chapters	CHAR(2)	99	01–99	Y		
METE_FEDBCK_ASSC	MENTEE_ID	Mentee id number	INT(5)	99999	00000 - 99999	Y	PK & FK	MENTEE
	MENTEE_FDB_ID	Mentee feedback id number	CHAR(5)	99999	10000 – 99999	Y	PK & FK	MENTEE_FEEDBACK
MENTEE_FEEDBACK	MENTEE_FDB_ID	Mentee feedback id number	CHAR(5)	99999	10000 – 99999	Y	PK	
	PROG_ID	Program id	VARCHAR(5)	Xxxxx		Y	FK	PROG_CONDUCTED
	MTEE_FD_Q1	Mentee feedback question 1	VARCHAR(45)	Xxxxx		Y		

	MTEE_FD_Q2	Mentee feedback question 2	VARCHAR(45)	Xxxxx		Y		
	MTEE_FD_Q3	Mentee feedback question 3	VARCHAR(45)	Xxxxx		Y		
	MTEE_FD_Q4	Mentee feedback question 4	VARCHAR(45)	Xxxxx		Y		
MEOR_FEDBCK_ASSC	MENTOR_ID	Mentor id number	INT(5)	99999	00000 - 99999	Y	PK & FK	MENTOR
	MENTOR_FDB_ID	Mentor feedback id number	CHAR(5)	99999	10000 – 99999	Y	PK & FK	MENTOR_FEEDBACK

MENTOR_FEEDBACK	MENTOR_FDB_ID	Mentor feedback id number	CHAR(5)	99999	10000 – 99999	Y	PK	
	PROG_ID	Program id	VARCHAR(5)	Xxxxx		Y	FK	PROG_CONDUCTED
	MTOR_FD_Q1	Mentor feedback question 1	VARCHAR(45)	Xxxxx		Y		
	MTOR_FD_Q2	Mentor feedback question 2	VARCHAR(45)	Xxxxx		Y		
	MTOR_FD_Q3	Mentor feedback question 3	VARCHAR(45)	Xxxxx		Y		
	MTOR_FD_Q4	Mentor feedback question 4	VARCHAR(45)	Xxxxx		Y		

MENTOR_REWARDS	MENTOR_ID	Mentor id number	INT(5)	99999	00000 - 99999	Y	PK & FK	MENTOR
	PROG_ID	Program id	VARCHAR(5)	Xxxxx		Y	PK & FK	PROG_CONDUCTED
	REWD_VALUE	Reward value	VARCHAR(30)	Xxxxx				
	REWD_POSITION	Reward position	VARCHAR(30)	Xxxxx				
	REWD_BENIFITS	Reward benefits	VARCHAR(30)	Xxxxx				
SESSIONS_COMPLETED	SESSION_NUM	Session Number	TINYINT(1)	9	0 - 9	Y	PK	
	PROG_ID	Program id	VARCHAR(5)	Xxxxx			FK	PROG_CONDUCTED
	ROOM_CODE	Room code	VARCHAR(5)	Xxxxx			FK	ROOM_REGISRTY

	SES_DATE	Session date	DATE	YYYY-MM-DD	'1000-01-01' to '9999-12-31'	Y		
	SES_STR_TIME	Session starts time	TIME	HH-MM-SS	00:00:00 to 23:59:59	Y		
	SES_END_TIME	Session end time	TIME	HH-MM-SS	00:00:00 to 23:59:59	Y		
ATTEND_REG_ASSC	MENTEE_ID	Mentee id number	INT(5)	99999	00000-99999	Y	PK&FK	MENTEE
	ATT_MARK_ID	Attendance marking id	VARCHAR(5)	Xxxxx		Y	PK&	ATTEND_REGISTRY

							FK	
	ATT_SESS_ID	Attendance session id	VARCHAR(5)	Xxxxx		Y	PK & FK	ATTEND_REGISTRY
ATTEND_SESSION_ASSC	ATT_SESS_ID	Attendance session id	VARCHAR(5)	Xxxxx		Y	PK & FK	ATTEND_REGISTRY
	ATT_MARK_ID	Attendance marking id	VARCHAR(5)	Xxxxx		Y	PK & FK	ATTEND_REGISTRY
	SESSION_NUM	Session Number	TINYINT(1)	9	0 - 9	Y	PK & FK	SESSIONS_COMPLETED

ATTEND_REGISTRY	ATT_SESS_ID	Attendance session id	VARCHAR(5)	Xxxxx		Y	PK	
	ATT_MARK_ID	Attendance marking id	VARCHAR(5)	Xxxxx		Y	PK	
	METR_ARRI_TIME	Mentor arrival time	TIME	HH-MM-SS	00:00:00 to 23:59:59			
	ATTE_NUM	Number of Attendance	TINYINT(2)	0 - 99				
	AB_NUM	Number of absences	TINYINT(2)	0 - 99				

CONV_MENTEE_ASSC	CON_MENTEE_ID	Conversation mentee id number	CHAR(5)	99999	10000 – 99999	Y	PK & FK	CONVERSATION_REGISTY
	CON_MENTOR_ID	Conversation mentor id number	CHAR(5)	99999	10000 – 99999	Y	PK & FK	CONVERSATION_REGISTY
	MENTEE_ID	Mentee id number	INT(5)	99999	00000 - 99999	Y	PK & FK	MENTEE
CONV_MENTOR_ASSC	CON_MENTOR_ID	Conversation mentor id number	CHAR(5)	99999	10000 – 99999	Y	PK & FK	CONVERSATION_REGISTY

	CON_MENTEE_ID	Conversation mentee id number	CHAR(5)	99999	10000 – 99999	Y	PK & FK	CONVERSATION_REGISTY
	MENTOR_ID	Mentor id number	INT(5)	99999	00000 - 99999	Y	PK & FK	MENTOR
CONVERSATION_REGISTY	CON_MENTEE_ID	Conversation mentee id number	CHAR(5)	99999	10000 – 99999	Y	PK	
	CON_MENTOR_ID	Conversation mentor id number	CHAR(5)	99999	10000 – 99999	Y	PK	
	PROG_ID	Program id	VARCHAR(5)	Xxxxx		Y	FK	PROG_CONDUCTED

	NUM_CHATS	Number of chats	INT(6)	999999	000000 - 999999			
	NUM_ACT_DAYS	Number of active days	SMALLINT(3)	999	0 - 999			

Developed by Author.

Business Rules

To develop the database system for the mentoring system, need to understand the main business practices that are associated with that area of the business process. And based on those identified business practices following business rules were developed to create the database that will cover all the activities that are related to the mentoring process in the software development organization.

1. Between EMPLOYEE and ORG_DEPAT
    a. An Employee is only registered under one organizational department.
2. Between ORG_DEPAT and ROOM_REGISTRY
    a. Each room must belong only to a department
    b. One Department can occupy multiple rooms
    c. Every organizational department can have one or more rooms inside a building or buildings.
    d. Need to create associative table (ORG_DEPAT_ROOM_ASSC) to connect tables of ORG_DEPAT and ROOM_REGISTRY
        i. This "d" business rule under the 2 point is not a business rule, it is a technical point to illustrate the requirements of connecting table to make the final relationship between ORG_DEPAT and ROOM_REGISTRY entities.
3. Between BULD_REGISTRY and ROOM_REGISRTY
    a. A building consisted with many rooms which can be belonged to different departments or one department.
4. Between EMPLOYEE and MENTOR
    a. A mentor is an employee.
    b. An employee can be a mentor, but it is not compulsory.
5. Between EMPLOYEE and MENTEE
    a. A mentee is an employee.
    b. An employee can be a mentee, but it is not compulsory.
6. Between MENTOR and PROG_CONDUCTED
    a. Mentees can join several mentoring programs at the same time.

b. One mentoring program can have more mentees
c. Need to create associative table (MENTOR_PROG_ASSC) to connect tables of MENTOR and PROG_CONDUCTED
    i. *This "c" business rule under the 6 point is not a business rule, it is a technical point to illustrate the requirements of connecting table to make the final relationship between MENTOR and PROG_CONDUCTED entities.*

7. Between MENTEE and PROG_CONDUCTED
    a. Mentors can do several mentoring programs at the organization
    b. Many mentors can jointly conduct one mentoring program.
    c. Need to create associative table (MENTEE_PROG_ASSC) to connect tables of MENTEE and PROG_CONDUCTED
        i. *This "c" business rule under the 7 point is not a business rule, it is a technical point to illustrate the requirements of connecting table to make the final relationship between MENTEE and PROG_CONDUCTED entities.*
8. Between MENTEE and MENTEE_FEEDBACK
    a. A mentee can give mentee feedback on the programs in which the mentee is enrolled.
    b. Mentee feedback is received from the mentees who have been enrolled on the program.
9. Between MENTOR and MENTOR_FEEDBACK
    a. A mentor can give mentor feedback on the programs that the mentor is delivering
    b. Mentor feedback is received from the mentors who have been delivering the program.
10. Between MENTOR_FEEDBACK and PROG_CONDUCTED
    a. Mentor feedbacks are only available for the programs that are conducted
11. Between MENTEE_FEEDBACK and PROG_CONDUCTED.
    a. Mentee feedbacks are only available for the programs that have been delivered.

12. Between MENTOR_REWARDS and PROG_CONDUCTED
    a. Mentor rewards may be distributed on the program that concluded.
    b. Receiving a mentor reward is not mandatory for any program that is conducted.
13. Between MENTOR_REWARDS and MENTOR
    a. Mentor may be received the mentor rewards for the completed session/s
    b. Mentor is not granted to receive the mentor rewards
14. Between PROG_CONDUCTED and SESSIONS_COMPLETED
    a. One mentoring program can have one or more sessions to cover the objective of the program.
15. Between SESSIONS_COMPLETED and ROOM_REGISTRY
    a. Only one room allocation is given for each session to be conducted.
16. Between SESSIONS_COMPLETED and ATTEND_ REGISTRY
    a. Session number need to have for mark the attendance
17. Between ATTEND_ REGISTRY and MENTEE
    a. Attendance of the mentees who registered for the program on the specific session.
18. Between CONVERSATION_REGISTY and PROG_CONDUCTED
    a. Each mentoring program has a conversation chat room to facilitate mentees to get more help from the mentors
19. Between CONVERSATION_REGISTY and MENTEE
    a. Mentees who register for each specific mentoring session will be invited to specific conversation chat rooms to participate
20. Between CONVERSATION_REGISTY and MENTOR
    a. Mentors who conduct each mentoring session will be invited to specific conversation chat rooms to participate

*Assumption 1 - To make sure the quality of the mentoring session mentoring sessions is only available for the 25 employees at one given time.

*Assumption 2 – Since only 25 employees can be registered into one program at a time, one room in a building can be used as the facilitation promise to conduct the sessions.

Constraint 1 – Mentee can get help for 2 subjects only at given time since time constraints with job

Constraint 2 – Mentor can only teach 2 subjects at a one given time since time constraints with job

Entity relationship model (ERM) components

The below table demonstrates the entity-relationship model components that are going to be used to develop the entity relationships among different entities of the mentoring system to be developed. The below relationship was developed based on the business rules that were discussed under the previous subheading of “business rules”. To develop the entity-relationship model components table used Coronel’s table (Coronel & Morris, 2017, Pg. 146).

Table 3: Components of ERM of Mentoring System

Entity	**Relationship**	**Connectivity**	**Entity**
ORG_DEPAT	Belongs	1:M	EMPLOYEE_ REGISTRY
ORG_DEPAT	Has	1:M	ORG_DEPAT_ROOM_ASSC
ROOM_REGISTRY	Has	1:M	ORG_DEPAT_ROOM_ASSC
BULD_REGISRTY	Has	1:M	ROOM_REGISTRY
EMPLOYEE_ REGISTRY	Is a	0:M	MENTEE
EMPLOYEE_ REGISTRY	Is a	0:M	MENTOR
MENTEE	Enroll	1:M	MENTEE_PROG_ASSC
PROG_CONDUCTED	Include	1:M	MENTEE_PROG_ASSC
MENTOR	Register	1:M	MENTOR_PROG_ASSC
PROG_CONDUCTED	Include	1:M	MENTOR_PROG_ASSC
MENTEE	Give	1:M	METE_FEDBCK_ASSC
MENTEEE_FEEDBACK	Receive	1:M	METE_FEDBCK_ASSC
PROG_CONDUCTED	Fill-up	1:M	MENTEEE_FEEDBACK
MENTOR	Give	1:M	MEOR_FEDBCK_ASSC
MENTOR_FEEDBACK	Receive	1:M	MEOR_FEDBCK_ASSC

PROG_CONDUCTED	Fill-up	1:M	MENTOR_FEEDBACK
MENTOR_REWARDS	Given	0:1	PROG_CONDUCTED
MENTOR_REWARDS	Transferred	0:1	MENTOR
SESSION_COMPLETED	Conducted	M:1	PROG_CONDUCTED
ROOM_REGISTRY	Used	1:1	SESSION_COMPLETED
MENTEE	Marked	1:M	ATTEND_REG_ASSC
ATTEND_REGISTRY	Labeled	1:M	ATTEND_REG_ASSC
ATTEND_REGISTRY	Sorted	1:M	ATTEND_SESSION_ASSC
SESSION_COMPLETED	Found	1:M	ATTEND_SESSION_ASSC
CONVERSATION_REGISTY	Happened	M:1	PROG_CONDUCTED
CONVERSATION_REGISTY	Attended	1:M	CONV_MENTEE_ASSC
MENTEE	Chat	1:M	CONV_MENTEE_ASSC
CONVERSATION_REGISTY	Answered	1:M	CONV_MENTOR_ASSC
MENTOR	Participated	1:M	CONV_MENTOR_ASSC

Developed by Author.

Data Book: Part 2: The Entity Relationship Diagram - Mentoring System Database

1. Relational schemas

Write shorthand relational schemas following the required format for each table.

Entity	Relational Schema
EMPLOYEE_ REGISTRY	EMPLOYEE_ REGISTRY (**EMPLOYEE_ID**, DEPT_CODE, EMP_FNAME, EMP_LNAME, EMP_INTIAL, EMP_ST_ADD, EMP_APT_NUM, EMP_CITY, EMP_STATE, EMP_ZIP_CODE, EMP_EMAIL, EMP_P_NUM, EMP_JOIN_DATE, EMP_MENTOR_STATUS, EMP_MENTEE_STATUS)
ORG_DEPAT	ORG_DEPAT (**DEPT_CODE**, DEPT_NAME, DEPT_EMAIL, DEPT_TEL, DEPT_ST_ADD, DEPT_BULD_NUM, DEPT_CITY, DEPT_STATE, DEPT_ZIP_CODE)
ORG_DEPAT_ROOM_ASSC	ORG_DEPAT_ROOM_ASSC (**DEPT_CODE**, **ROOM_CODE**)
ROOM_REGISTRY	ROOM_REGISTRY (**ROOM_CODE**, BLD_CODE, ROOM_CAP, ROOM_TEL_NUM, ROOM_BUL_NUM)
BULD_REGISTRY	BULD_REGISTRY (**BLD_CODE**, BULD_CODINA, NUM_FLOORS, BULD_NAME, BULD_OPEN_DATE)
MENTOR	MENTOR (**EMPLOYEE_ID**, MENTOR_ID, REG_DATE, MENTOR_SUB_1, MENTOR_SUB_2)
MENTOR_PROG_ASSC	MENTOR_PROG_ASSC (**MENTOR_ID**, **PROG_ID**)

MENTEE	MENTEE (**EMPLOYEE_ID**, MENTEE_ID, REQ_DATE, 1_SUB_REQ, 2_SUB_REQ)
MENTEE_PROG_ASSC	MENTEE_PROG_ASSC (**MENTEE_ID**, **PROG_ID**)
PROG_CONDUCTED	PROG_CONDUCTED (**PROG_ID**, PROG_NAME, PROG_DESCP, PROG_MODE, PROG_CHAPTERS)
MEOR_FEDBCK_ASSC	MEOR_FEDBCK_ASSC (**MENTOR_ID**, **MENTOR_FDB_ID**)
METE_FEDBCK_ASSC	METE_FEDBCK_ASSC (**MENTEE_ID**, **MENTEE_FDB_ID**)
MENTEE_FEEDBACK	MENTEE_FEEDBACK (**MENTEE_ID**, PROG_ID, MTEE_FD_Q1, MTEE_FD_Q2, MTEE_FD_Q3, MTEE_FD_Q4)
MENTOR_FEEDBACK	MENTOR_FEEDBACK (**MENTOR_ID**, PROG_ID, MTOR_FD_Q1, MTOR_FD_Q2, MTOR_FD_Q3, MTOR_FD_Q4)
MENTOR_REWARDS	MENTOR_REWARDS (**MENTOR_ID**, **PROG_ID**, REWD_VALUE, REWD_POSITION, REWD_BENIFITS)
SESSIONS_COMPLETED	SESSIONS_COMPLETED (**SESSION_NUM**, PROG_ID, ROOM_CODE, SES_DATE, SES_STR_TIME, SES_END_TIME)
ATTEND_REG_ASSC	ATTEND_REG_ASSC (**MENTEE_ID**, **ATT_MARK_ID**, **ATT_SESS_ID**)

ATTEND_SESSION_ASSC	ATTEND_SESSION_ASSC (**SESSION_NUM**, **ATT_SESS_ID, ATT_MARK_ID**)
ATTEND_REGISTRY	ATTEND_REGISTRY (**ATT_MARK_ID** , **ATT_SESS_ID**, METR_ARRI_TIME, ATTE_NUM, AB_NUM)
CONVERSATION_REGISTY	CONVERSATION_REGISTY (**CON_MENTOR_ID**, **CON_MENTEE_ID**, PROG_ID, NUM_CHATS, NUM_ACT_DAYS)
CONV_MENTOR_ASSC	CONV_MENTOR_ASSC (**MENTOR_ID**, **CON_MENTOR_ID**, **CON_MENTEE_ID**)
CONV_MENTEE_ASSC	CONV_MENTEE_ASSC (**MENTEE_ID**, **CON_MENTEE_ID**, **CON_MENTOR_ID**)

2. Using Crow's Foot notation, build a diagram of all tables in Microsoft Visio or draw.io.

To access the original file draw.io file [please click here]

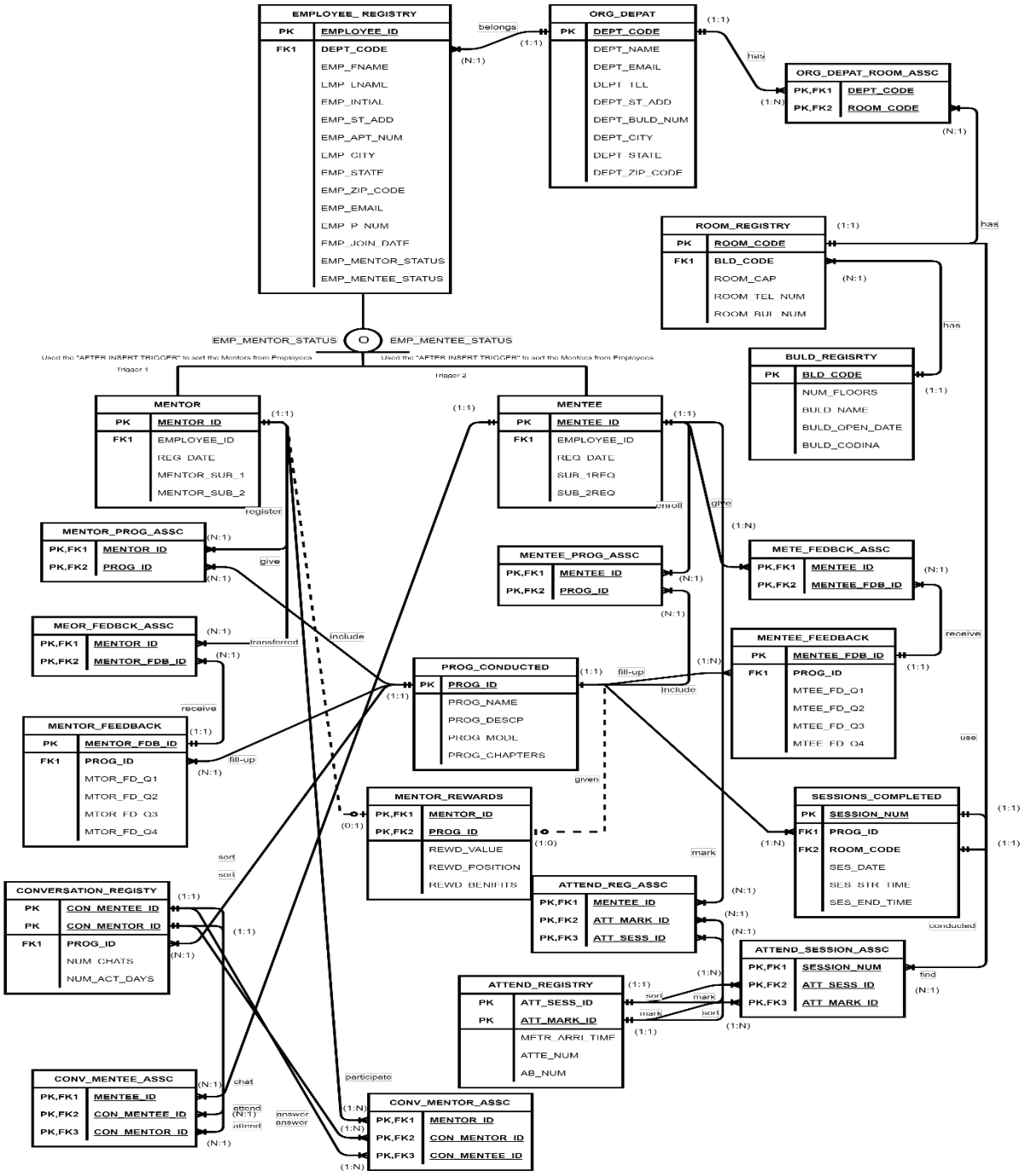

Figure 1: ER diagram on Mentoring System Database: Developed by Author via draw.io.

3. Using Crow's Foot notation, build a subtype/supertype diagram among one set of related tables in Microsoft Visio or draw.io.

Used the trigger type of **"After insert trigger"** for creating the relationship between subtype and supertype tables under this assignment via MYSQL (Oracle, 2022). Descriptive explanation has been given under the heading of Creation of the Subtype/Supertype relationship which is in page number 39 of this assignment.

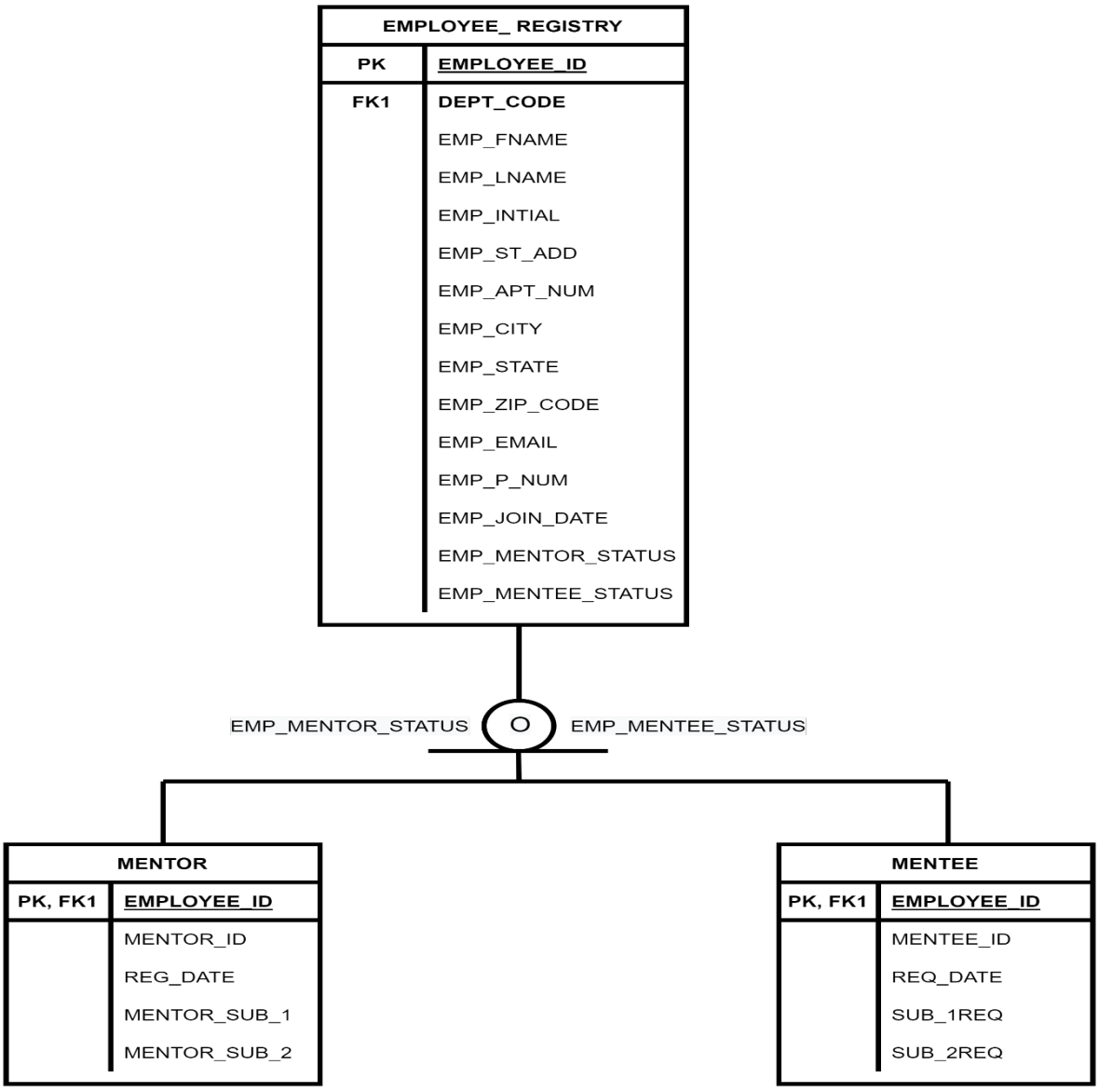

Figure 2: Extracted view on Super/subtype ER diagram : Developed by Author via draw.io. (conceptual only)

4. A documented walk-through of one normalized table.

Attributes that selected to make the normalization forms as below table of EMPLOYEE_ REGISTRY entity.

Table 4: Newly created EMPLOYEE_REGISRTY entity for answer on normalization using the existing entities of the mentoring system database

Entity Name	**Attribute Name**	**Attribute Designations**
EMPLOYEE_ REGISTRY	EMPLOYEE_ID	Primary Key
	DEPT_CODE	Primary Key
	DEPT_NAME	-
	DEPT_EMAIL	-
	DEPT_TEL	-
	ROOM_CODE	-
	ROOM_CAP	-
	ROOM_TEL_NUM	-
	ROOM_BUL_NUM	-
	EMP_FNAME	-
	EMP_LNAME	-
	EMP_INTIAL	-
	EMP_EMAIL	-
	EMP_P_NUM	-

	EMP_JOIN_DATE	-

Developed by Author

First Normal Form (1NF)

Relational Schema (1NF)

EMPLOYEE_REGISTRY (**EMPLOYEE_ID**, **DEPT_CODE**, DEPT_NAME, DEPT_EMAIL, DEPT_TEL, ROOM_CODE, ROOM_CAP, ROOM_TEL_NUM, ROOM_BUL_NUM, EMP_FNAME, EMP_LNAME, EMP_INTIAL, EMP_EMAIL, EMP_P_NUM, EMP_JOIN_DATE)

Dependencies

Partial Dependencies:

(EMPLOYEE_ID → EMP_FNAME, EMP_LNAME, EMP_INTIAL, EMP_EMAIL, EMP_P_NUM, EMP_JOIN_DATE)

(DEPT_CODE → DEPT_NAME, DEPT_EMAIL, DEPT_TEL, ROOM_CODE, ROOM_CAP, ROOM_TEL_NUM, ROOM_BUL_NUM)

Transitive dependencies:

(ROOM_CODE → ROOM_CAP, ROOM_TEL_NUM, ROOM_BUL_NUM)

Dependency Diagram (1NF)

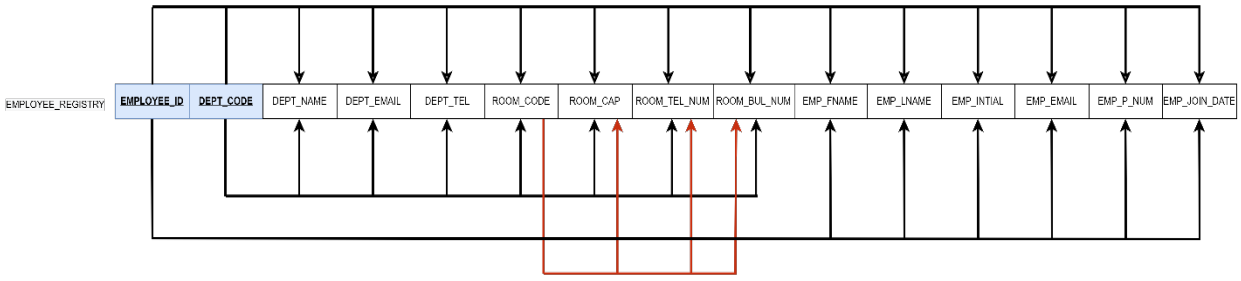

Figure 3: 1NF Dependency Diagram based on relational schemas on 1NF: Developed by Author

To view the original work, [please click here]

Second Normal Form (2NF)

Identified Partial Dependencies and Other Dependencies with New Tables (2NF)

Partial Dependencies:

1. (EMPLOYEE_ID → EMP_FNAME, EMP_LNAME, EMP_INTIAL, EMP_EMAIL, EMP_P_NUM, EMP_JOIN_DATE)
    a. New Table Name: **EMPLOYEE**
2. (DEPT_CODE → DEPT_NAME, DEPT_EMAIL, DEPT_TEL, ROOM_CODE, ROOM_CAP, ROOM_TEL_NUM, ROOM_BUL_NUM)
    a. New Table Name: **ORG_DEPAT**

Other Dependencies (Transitive Dependencies):

1. (ROOM_CODE → ROOM_CAP, ROOM_TEL_NUM, ROOM_BUL_NUM)

Relational Schema (2NF)

EMPLOYEE (**EMPLOYEE_ID**, EMP_FNAME, EMP_LNAME, EMP_INTIAL, EMP_EMAIL, EMP_P_NUM, EMP_JOIN_DATE

ORG_DEPAT (**DEPT_CODE**, DEPT_NAME, DEPT_EMAIL, DEPT_TEL, ROOM_CODE, ROOM_CAP, ROOM_TEL_NUM, ROOM_BUL_NUM)

EMPLOYEE_REGISTRY (**EMPLOYEE_ID**, **DEPT_CODE**)

Dependency Diagram (2NF)

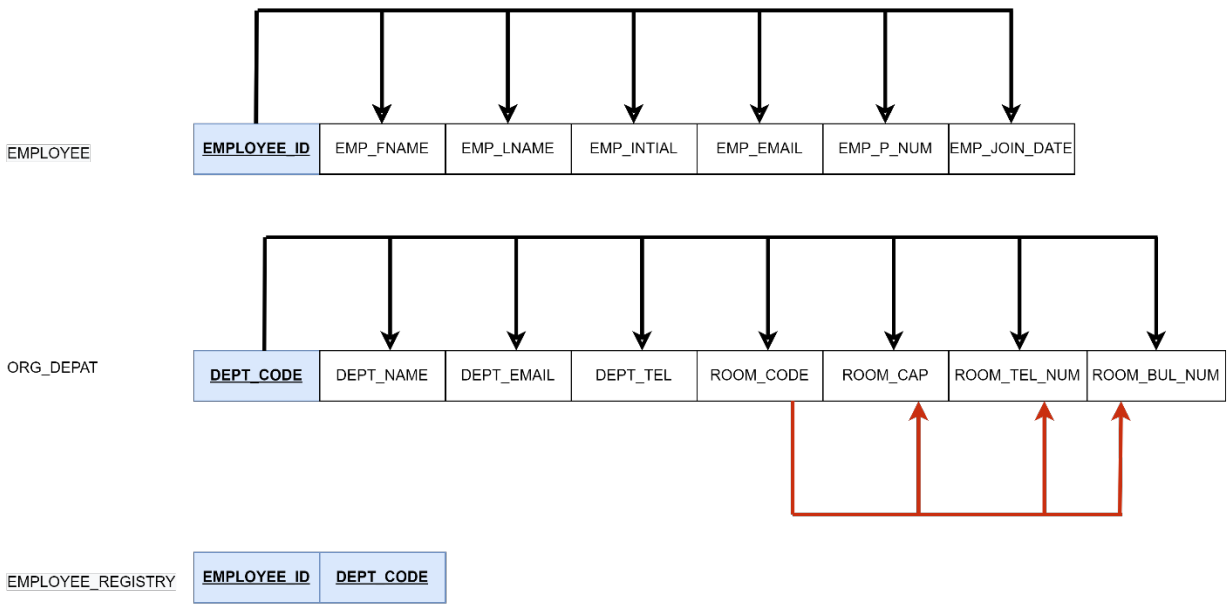

Figure 4: 2NF Dependency Diagram based on relational schemas on 2NF : Developed by Author

To view the original work, [please click here]

Third Normal Form (3NF)

Identified Transitive Dependencies with Related New Tables (3NF)

1. (ROOM_CODE → ROOM_CAP, ROOM_TEL_NUM, ROOM_BUL_NUM)
    a. New Table Name: ROOM_REGISTRY

Relational Schema (3NF)

EMPLOYEE (**EMPLOYEE_ID**, EMP_FNAME, EMP_LNAME, EMP_INTIAL, EMP_EMAIL, EMP_P_NUM, EMP_JOIN_DATE

ORG_DEPAT (**DEPT_CODE**, DEPT_NAME, DEPT_EMAIL, DEPT_TEL, ROOM_CODE)

ROOM_REGISTRY (**ROOM_CODE**, ROOM_CAP, ROOM_TEL_NUM, ROOM_BUL_NUM)

EMPLOYEE_REGISTRY (**EMPLOYEE_ID**, **DEPT_CODE**)

Dependency Diagram (3NF)

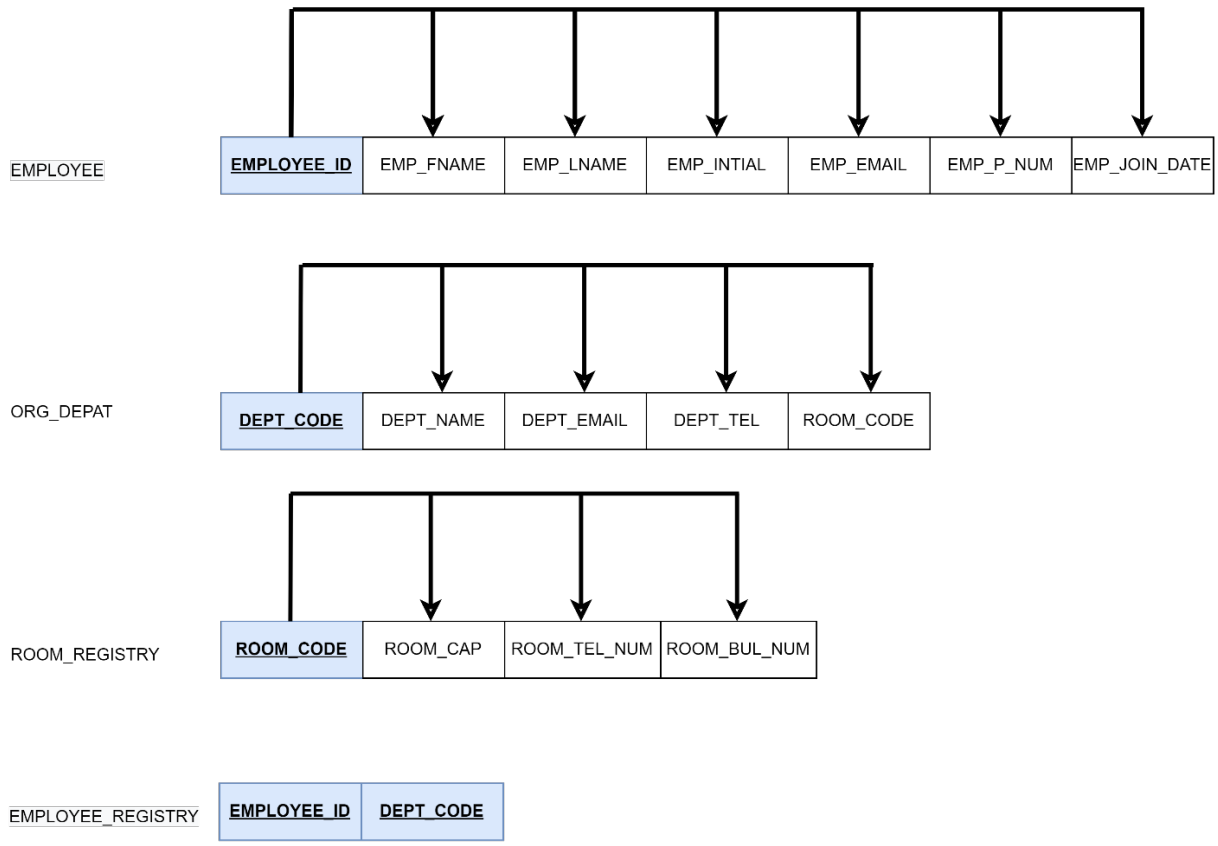

Figure 5: 3NF Dependency Diagram based on relational schemas on 3NF : Developed by Author

To view the original work, [please click here]

Data Book: Part 3: The Database

Build the relationships

Creation of the Subtype/Supertype relationship

Used 2 "**AFTER INSERT" TRIGGER** type under the SQL to build the subtype/supertype relationships among the tables of EMPLOYEE_REGISTRY, MENTOR and MENTEE (Oracle, 2022).

To select the employee data into the MENTOR table from EMPLOYEE_REGISRTY table

To selection of the data to create the AFTER INSERT TRIGGER used the reference data that has been inputted into the EMPLOYEE_REGISTRY. EMP_MENTOR_STATUS column. Below code has been created to sort the data as per the subtype/supertype relationship between the EMPLYEE_REGISTY and MENTOR tables.

```
CREATE DEFINER = CURRENT_USER TRIGGER
`Project`.`EMPLOYEE_REGISTRY_AFTER_INSERT_MENTOR` AFTER INSERT ON
`EMPLOYEE_REGISTRY` FOR EACH ROW

BEGIN

    IF NEW.EMP_MENTOR_STATUS IN ('Y', 'y', 'YES', 'yes') THEN

        INSERT INTO MENTOR(EMPLOYEE_ID)

        VALUES(NEW.EMPLOYEE_ID);

    END IF;

END
```

```
CREATE DEFINER = CURRENT_USER TRIGGER `Project`.`EMPLOYEE_REGISTRY_AFTER_INSERT_MENTOR` AFTER INSERT ON `EMPLOYEE_REGISTRY` FOR EACH ROW
BEGIN
    IF NEW.EMP_MENTOR_STATUS IN ('Y', 'y', 'YES', 'yes') THEN
        INSERT INTO MENTOR(EMPLOYEE_ID)
        VALUES(NEW.EMPLOYEE_ID);
    END IF;
END
```

Figure 6: AFTER INSERT Trigger - Mentor Selection - Developed by Author via MYSQL Workbench

To select the employee data into the MENTEE table from EMPLOYEE_REGISRTY table

To selection of the data to create the AFTER INSERT TRIGGER used the reference data that has been inputted into the EMPLOYEE_REGISTRY. EMP_MENTEE_STATUS column. Below code has been created to sort the data as per the subtype/supertype relationship between the EMPLYEE_REGISTY and MENTEE tables.

```
CREATE DEFINER = CURRENT_USER TRIGGER `Project`.`EMPLOYEE_REGISTRY_AFTER_INSERT_MENTEE` AFTER INSERT ON `EMPLOYEE_REGISTRY` FOR EACH ROW

BEGIN

    IF NEW.EMP_MENTEE_STATUS IN ('Y', 'y', 'YES', 'yes') THEN

        INSERT INTO MENTEE(EMPLOYEE_ID)

        VALUES(NEW.EMPLOYEE_ID);

    END IF;

END
```

```
CREATE DEFINER = CURRENT_USER TRIGGER `Project`.`EMPLOYEE_REGISTRY_AFTER_INSERT_MENTEE` AFTER INSERT ON `EMPLOYEE_REGISTRY` FOR EACH ROW
BEGIN
    IF NEW.EMP_MENTEE_STATUS IN ('Y', 'y', 'YES', 'yes') THEN
        INSERT INTO MENTEE(EMPLOYEE_ID)
        VALUES(NEW.EMPLOYEE_ID);
    END IF;
END
```

Figure 7: AFTER INSERT Trigger - Mentee Selection - Developed by Author via MYSQL Workbench

Working SQL Query #1: Data from One Table

Question:

How many people are coming to work at company from zip code “46142”?

Answer:

USE Project; * Optional

SELECT COUNT(EMPLOYEE_REGISTRY.EMPLOYEE_ID)

FROM EMPLOYEE_REGISTRY

WHERE EMP_ZIP_CODE = ‘46142’

```
66    USE Project;
67    SELECT COUNT(EMPLOYEE_REGISTRY.EMPLOYEE_ID)
68    FROM EMPLOYEE_REGISTRY
69    WHERE EMP_ZIP_CODE = '46142';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

COUNT(EMPLOYEE_REGISTRY.EMPLOYEE_ID)
2

Figure 8: Output of Working SQL on Data from One Table - Developed by Autor via MYSQL Workbench

Working SQL Query #2: Data from Two Tables

Question:

How many sessions were completed for ‘JAVA’ computer programing?

Answer:

```
SELECT PROG.PROG_NAME, COUNT(SESSIONS_COMPLETED.SESSION_NUM)
FROM PROG_CONDUCTED AS PROG RIGHT JOIN SESSIONS_COMPLETED
        ON PROG.PROG_ID = SESSIONS_COMPLETED.PROG_ID
WHERE PROG.PROG_NAME = 'JAVA'
GROUP BY PROG.PROG_ID;
```

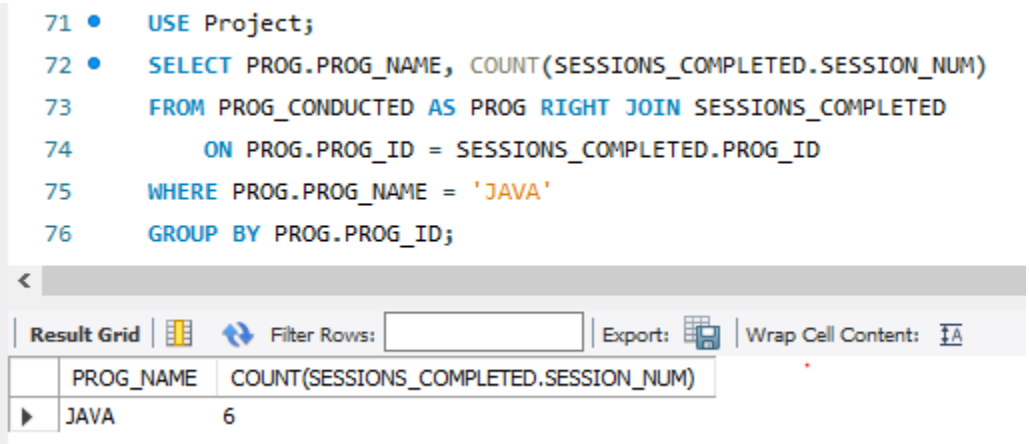

Figure 9: Output of Working SQL on Data from Two Tables - Developed by Autor via MYSQL Workbench

Working SQL Query #3: Subquery

Question:

Who are the employees (employees who join before 12-11-2021) being registered as mentor and mentee both at same time with their selected subject name?

Answer:

```
SELECT EMPLOYEE.EMPLOYEE_ID,  EMPLOYEE.EMP_FNAME,
EMPLOYEE.EMP_LNAME,

        MENTEE.SUB_1REQ, MENTEE.SUB_2REQ, MENTOR.MENTOR_SUB_1,
MENTOR.MENTOR_SUB_2

FROM EMPLOYEE_REGISTRY AS EMPLOYEE INNER JOIN (

        SELECT EMPLOYEE_ID, MENTOR_SUB_1, MENTOR_SUB_2

        FROM MENTOR

) AS MENTOR

        ON EMPLOYEE.EMPLOYEE_ID = MENTOR.EMPLOYEE_ID

INNER JOIN (

        SELECT EMPLOYEE_ID, SUB_1REQ, SUB_2REQ

        FROM MENTEE

) AS MENTEE

        ON EMPLOYEE.EMPLOYEE_ID = MENTEE.EMPLOYEE_ID

WHERE EMPLOYEE.EMP_JOIN_DATE < DATE('2021-12-11');
```

```sql
USE Project;
SELECT EMPLOYEE.EMPLOYEE_ID,  EMPLOYEE.EMP_FNAME, EMPLOYEE.EMP_LNAME,
    MENTEE.SUB_1REQ, MENTEE.SUB_2REQ, MENTOR.MENTOR_SUB_1, MENTOR.MENTOR_SUB_2
FROM EMPLOYEE_REGISTRY AS EMPLOYEE INNER JOIN (
    SELECT EMPLOYEE_ID, MENTOR_SUB_1, MENTOR_SUB_2
    FROM MENTOR
) AS MENTOR
    ON EMPLOYEE.EMPLOYEE_ID = MENTOR.EMPLOYEE_ID
INNER JOIN (
    SELECT EMPLOYEE_ID, SUB_1REQ, SUB_2REQ
    FROM MENTEE
) AS MENTEE
    ON EMPLOYEE.EMPLOYEE_ID = MENTEE.EMPLOYEE_ID
WHERE EMPLOYEE.EMP_JOIN_DATE < DATE('2021-12-11');
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

EMPLOYEE_ID	EMP_FNAME	EMP_LNAME	SUB_1REQ	SUB_2REQ	MENTOR_SUB_1	MENTOR_SUB_2
EP001	Hasaranga	Jayathilake	R	Python	SQL	BI
EP004	Jayanga	Anushan	JAVA	R	Python	PHP
EP006	Damayathi	Shamali	Python	SQL	JAVA	PHP

Figure 10: Output of Working SQL Subquery - Developed by Autor via MYSQL Workbench

Reference

Adil, A., Kausar, S., Ameer, S., Ghayas, S., & Shujja, S. (2021). Impact of organizational socialization on organizational citizenship behavior: mediating role of knowledge sharing and role clarity. *Current Psychology 2021*, 1–9. https://doi.org/10.1007/S12144-021-01899-X

Clark, K. R. (2017). Managing Multiple Generations in the Workplace. *Radiologic Technology*, *88*(4), 379–396. http://www.radiologictechnology.org/content/88/4/379.full

Coronel, C., & Morris, S. (2017). *DATABASE SYSTEMS*. www.cengage.com/highered

Eriksson, T., Näppä, A., & Robertson, J. (2022). Crafting a paying-it-forward mindset in business: Five principles for a competitive employer branding advantage. *Business Horizons*. https://doi.org/10.1016/J.BUSHOR.2022.01.001

Ge, J., Su, X., & Zhou, Y. (2010). Organizational socialization, organizational identification and organizational citizenship behavior: An empirical research of Chinese high-tech manufacturing enterprises. *Nankai Business Review, International*, *1*(2), 166–179. https://doi.org/10.1108/20408741011052573

Oracle. (2022). *MySQL :: MySQL 8.0 Reference Manual :: 25.3.1 Trigger Syntax and Examples*. https://dev.mysql.com/doc/refman/8.0/en/trigger-syntax.html

Ng, K. Y. N. (2022). Effects of organizational culture, affective commitment and trust on knowledge-sharing tendency. *Journal of Knowledge Management*, *ahead-of-print*(ahead-of-print). https://doi.org/10.1108/JKM-03-2022-0191/FULL/PDF

Orlova, E. v. (2021). Design of Personal Trajectories for Employees' Professional Development in the Knowledge Society under Industry 5.0. *Social Sciences 2021, Vol. 10, Page 427*, *10*(11), 427. https://doi.org/10.3390/SOCSCI10110427

Appendix

Links below for the Crow's foot notation entity relationship diagram and 1NF, 2NF, and 3NF

Crow's foot notation entity relationship diagram

https://drive.google.com/file/d/1i6LvQOztmVZ0gG4vVjxFtLpliw7VqST9/view?usp=sharing

1NF, 2NF, and 3NF Dependency Diagrams

https://drive.google.com/file/d/1mkHSOEtgp9Yza2toUdPQNLUnU5AR0QKa/view?usp=sharing