

## Laboratory 3

Title of the Laboratory Exercise: data model to relational model

### 1. Introduction and Purpose of Experiment

The ER schema is to be converted into a relational schema as data cannot be stored in an ER schema. A relation schema gives the basic information describing a table or relation. It is the logical definition of a table. This includes a set of column names, and the data types associated with each column. By doing this lab, students will be able to map ER schema to relational schema.

### 2. Aim and Objectives

Aim

- To map data model to relational model

Objectives

At the end of this lab, the student will be able to

- Map ER schema to relational schema
- Insert tuples using SQL commands for the developed database schema

### 3. Experimental Procedure

- i. Map all the components in the ER diagram to corresponding relation entities and instances
- ii. Insert tuples using SQL commands
- iii. Design SQL commands using aggregate functions in SQL
- iv. Execute SQL commands
- v. Test the executed commands
- vi. Document the Results
- vii. Analyse and discuss the outcomes of your experiment

### 4. Questions

- a. Consider the ER diagram you have drawn in Laboratory 2. Convert the ER diagram to corresponding relational database schema.
- b. Insert the tuples (minimum five) for the developed database schema using SQL commands. Perform aggregate functions in SQL based on the developed database schema.

## 5. Calculations/Computations/Algorithms

## Step 1: Mapping of Regular Entity Types

## MANAGER

<u>mgr_id</u>	mgr_username	mgr_password
---------------	--------------	--------------

## EMPLOYEE

<u>emp_id</u>	emp_name	emp_type	emp_contact_no	emp_address	last_paid_on
---------------	----------	----------	----------------	-------------	--------------

## PROJECT

project_id	project_name	project_type	project_status
------------	--------------	--------------	----------------

## DEPARTMENT

<u>dept_id</u>	dept_name	dept_budget
----------------	-----------	-------------

## Step 2: Mapping of Weak Entity Types

## SALARY

<u>emp_id</u>	<u>salary_type</u>	base_salary	gross_salary	allowance
---------------	--------------------	-------------	--------------	-----------

## Step 3: Mapping of Binary 1:1 Relation Types

## SALARY

<u>sal_emp_id</u>	<u>salary_type</u>	base_salary	gross_salary	allowance
-------------------	--------------------	-------------	--------------	-----------

## Step 4: Mapping of Binary 1:N Relation Types

## EMPLOYEE

<u>emp_id</u>	<u>mgr_id</u>	dept_id	emp_name	emp_type	last_paid_on
---------------	---------------	---------	----------	----------	--------------

## Step 5: Mapping of Binary M:N Relationship Types

## UNDERTAKES

emp_id	project_id
--------	------------

## Step 6: Mapping of Multivalued attributes

## EMP\_CONTACT\_NOS

emp_id	emp_contact_no
--------	----------------

## EMP\_ADDRESSES

emp_id	emp_address
--------	-------------

## Step 7: Mapping of N-ary Relationship Types

There are no N-ary relationships in the ER Diagram

## 6. Presentation of Results

## EMPLOYEE

<u>emp_id</u>	<u>mgr_id</u>	dept_id	emp_name	emp_type	last_paid_on
---------------	---------------	---------	----------	----------	--------------

## EMP\_CONTACT\_NOS

emp_id	emp_contact_no
--------	----------------

## EMP\_ADDRESSES

emp_id	emp_address
--------	-------------

## MANAGER

<u>mgr_id</u>	mgr_username	mgr_password
---------------	--------------	--------------

## PROJECT

project_id	project_name	project_type	project_status
------------	--------------	--------------	----------------

## UNDERTAKES

emp_id	project_id
--------	------------

## SALARY

<u>sal_emp_id</u>	<u>salary_type</u>	base_salary	gross_salary	allowance
-------------------	--------------------	-------------	--------------	-----------

## DEPARTMENT

<u>dept_id</u>	dept_name	dept_budget
----------------	-----------	-------------

## 7. Conclusions

The ER Model is intended as a description of real-world entities. Although it is constructed in such a way as to allow easy translation to the relational schema model, this is not an entirely trivial process. The ER diagram represents the conceptual level of database design meanwhile the relational schema is the logical level for the database design.

We can generate relational database schema using the ER diagram. Following are some key points to keep in mind while doing so:

1. Entity gets converted into Table, with all the attributes becoming fields(columns) in the table.
2. Relationship between entities is also converted into table with primary keys of the related entities also stored in it as foreign keys.
3. Primary Keys should be properly set.
4. For any relationship of Weak Entity, if primary key of any other entity is included in a table, foreign key constraint must be defined.

## 8. Comments

### 1. Limitations of Experiments

Not for unstructured data: Unless the data is cleanly delineated into different fields, rows or columns, ER diagrams are probably of limited use. The same is true of semi-structured data, because only some of the data will be useful.

Difficulty integrating with an existing database: Using ER Models to integrate with an existing database can be a challenge because of the different architectures.

### 2. Limitations of Results

In the results, we haven't yet defined the primary keys and foreign keys, even though entity integrity and referential integrity is preserved, its not clear from the schema made from the ER diagram.

### 3. Learning happened

We learnt how to convert an ER diagram to a Relational Schema.

### 4. Recommendations

Beyond concerns over meeting the constraint requirements for primary keys, we must also assure adherence to the referential integrity constraints. We identify the referential integrity constraints by locating the corresponding attribute in each relation that is linked via a relationship. We then

determine which of the relations contain the tuple that if the reference attribute were deleted or changed would jeopardize the integrity of the database.