

Theory Part:

Task 1:

a) Describe the functional dependencies in the given tables.

Answer: Functional Dependency is a constraint that determines the relation of one attribute to another attribute in a Database Management System (DBMS). Functional Dependency helps to maintain the quality of data in the database. It plays a vital role to find the difference between good and bad database design.

Employee no	Employee Name	Salary	Department
1	Amit	50000	OBIEE
2	Divya	40000	COGNOS
3	Rama	57000	Administrator

b) For each of the given tables of this database determine they are in which normal form.

Answer: Database is in 2nd Normal Form (2NF). Because The first condition in the 2nd NF is that the table has to be in 1st NF. The table also should not contain partial dependency. Here partial dependency means the proper subset of candidate key determines a non-prime attribute. To understand in a better way lets look at the below example.

c) Convert all the tables to 3rd normal forms.

Employee No	Employee Name	Department no	Department	Salary Slip No	Salary
1	Amit	OBIEE,ETL	101	0001	50000
2	Divya	COGNOS	102	0002	40000
2	Rama	Administrator	103	0003	57000

d) Design an enhanced entity relationship model based on normalized tables.

Answer: Normalization is the process of organizing data in a database. This includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency. Normalized table has been created Screenshot has given below.

Lab Part:

Task 2:































a) Develop a database on normalized tables and show the results through screenshots.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 employee_no	int(200)			No	None			Change Drop More
<input type="checkbox"/>	2 employee_name	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/>	3 department_no	int(200)			No	None			Change Drop More
<input type="checkbox"/>	4 department_name	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/>	5 salary_slip_no	int(255)			No	None			Change Drop More
<input type="checkbox"/>	6 salary	int(255)			No	None			Change Drop More

employee_no	employee_name	department_no	department_name	salary_slip_no	salary
1	Amit	101	OBIEE	1	50000
1	Divya	102	COGNOS	2	40000
1	Rama	103	Administrator	3	57000

b) Insert at least 10 entity in each tables.

Employee Table:

←T→				employee_no	employee_name	department_name
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	KAYLING	Operations
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	BLAZE	Accounting
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	CLARE	Research
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	JONAS	Accounting
<input type="checkbox"/>	 Edit	 Copy	 Delete	5	SCARLET	Research
<input type="checkbox"/>	 Edit	 Copy	 Delete	6	SCOTT	Sales
<input type="checkbox"/>	 Edit	 Copy	 Delete	7	FORD	Sales
<input type="checkbox"/>	 Edit	 Copy	 Delete	8	ADELYN	Operations
<input type="checkbox"/>	 Edit	 Copy	 Delete	9	MADDEN	Sales
<input type="checkbox"/>	 Edit	 Copy	 Delete	10	TUCKER	Operations

Department Table:

department_no	department_name
1001	Accounting
1002	Research
1003	Sales
1004	Operations

Salary Table:

employee_no	salary_clip_no	employee_name	salary
1	1	KAYLING	35000
2	2	BLAZE	15000
3	3	CLARE	55000
4	4	JONAS	45000
5	5	SCARLET	30000
6	6	SCOTT	65000
7	7	FORD	38000
8	8	ADELYN	25000
9	9	MADDEN	28000
10	10	TUCKER	48000

c) Write a query to show average salary of employee each department.

Answer:

```
SELECT AVG(salary.salary), employee.department_name FROM salary INNER JOIN employee ON salary.employee_name = employee.employee_name GROUP BY employee.department_name;
```

AVG(salary.salary)	department_name
30000.0000	Accounting
36000.0000	Operations
42500.0000	Research
43666.6667	Sales

d) Write a query to show second maximum salary of employee in each department.

Answer:

```
SELECT MAX(salary.salary), employee.department_name FROM salary INNER JOIN employee ON salary.employee_name = employee.employee_name GROUP BY employee.department_name;
```

MAX(salary.salary)	department_name
45000	Accounting
48000	Operations
55000	Research
65000	Sales

e) Write a query to show total number of employee in each department.

Answer:

```
SELECT COUNT(employee.department_name) FROM salary INNER JOIN employee ON salary.employee_name = employee.employee_name GROUP BY employee.department_name;
```

COUNT(employee.department_name)
2
3
2
3