INTERNATIONAL INSTITUTE OF PROFESSIONAL STUDIES, DAVV, INDORE
M. Tech. (IT) 5 Years IV SEMESTER
IT-407B DBMS Lab Assignment

Section -A

1. Study and implementation of following DDL commands:
   a. CREATE TABLE
   b. ALTER TABLE
   c. DROP TABLE
   d. RENAME
   e. TRUNCATE TABLE
2. Study and implementation of following DML commands:
   a. INSERT INTO
   b. SELECT
   c. UPDATE
   d. DELETE
3. Study and implementation of following DCL commands:
   a. GRANT
   b. REVOKE
4. Study and implementation of following TCL commands:
   a. COMMIT
   b. ROLLBACK
   c. SAVEPOINT

Section -B

1. Define the schema for the following databases with specific data type and constraints, the table name and its fields name are to be taken from database description which are given below :

   A database is being constructed for storing sales information system.
   A product can be described with a unique product number, product name, selling price, manufacturer name. The product can sale to a particular client and each client have it own unique client number, client name, client addresses, city, pin code, state and total balance to be required to paid. Each client orders to buy product from the salesman. In the order, it has unique sales order number, sales order date, client number, salesman number (unique), billed whole payment by the party or not and its delivery date.
   The salesman have the name, addresses, city, pin code, state, salary of the sales man, delivery date, total quantity ordered, product rate.

   a) Draw an Entity Relationship diagram for above scenario. Make and state your assumptions where required.
   b) Convert above ER model into relational model in step by step manner.
   c) Write the SQL queries for the following –
      1. Create above tables with all constraints required and insert data into above tables.
      2. Retrieve the list of names and the cities of all the clients.
      3. List the various products available.
      4. Find the names of all clients having ‘a’ as the second letter in their names.
      5. List all the clients who are located in TEZPUR.
      6. Find products whose selling price is greater than 2000& less than or equal to 5000
      7. Add a new column NEW_PRICE into the product_master table.
      8. Rename the column product_rate of Sales_Order_Details to new_product_rate.
      9. List the products in sorted order of their description.
      10. Display the order number and date on which the clients placed their order.
      11. Delete all the records having delivery date before 25th August, 2008.
      12. Change the delivery date of order number ON01008 to 16-08-08
      13. Change the bal_due of client_no CN01003 to 1200
      14. Find the product with description as ‘HDD1034’ and ‘DVDRW’
      15. List the names, city and state of the clients not in the state of ‘ASSAM’
      16. List of all orders that were canceled in the of March .

2. **Consider the schema for the following databases with specific data type and constraints, the table structure is given below:**

   1.Employee F_Name varchar2(15) NOT NULL
   LName varchar2(15) NOT NULL,

Emp_id varchar2(5) Primary Key,
city varchar(10),
Gender char(1) ,
Emp_hire_date date
Job_code varchar(5)
Supervisor_id varchar(5)
Dept_no number(4)

Constraint- Emp_id pK
Emp CHECK (Sex IN ('M', 'm', 'F', 'f')),
Supervisor_id Foreign key references emp_id of employee
Dept_no foreign key references Dep_no of Department

2.Department DName varchar(15) ,
DepNo unmber(4)
Mgr_id char(9) NOT NULL

Constraints- unique(DName),
Primary Key (DepNo),
Foreign Key (Mgr_id) REFERENCES employee (emp_id)

3.Project PName varchar(15) not ,
PNumber number(5) not null,
DepNo number(4),

Constraints - Primary Key (PNumber),
Foreign Key (DepNo) REFERENCES department (DepNo)

4.Works_on emp_id varchar(5) ,
PNo number(5)

Constraints - Primary Key (ESSN, PNo),
Foreign Key (emp_id) REFERENCES employee (emp_id)
Foreign Key (PNo) REFERENCES project (PNumber)

5.Dependent Emp_id varchar(5) ,
Dependent_Name varchar(15) not null,
gender char(1)

Constraints - Primary Key (emp_id, Dependent_Name),
Check (Gender IN ('M', 'm', 'F', 'f')),
Foreign Key (emp_id) REFERENCES employee (emp_id)

Write SQL queries for following:

1. Create above tables with all constraints mentioned.
2. Insert data into above tables.
3. Write the SQL code to change the job code to 501 for the person whose emp_id is'888665555'. After you have completed the task, examine the results, and then reset the job code to its original value.
4. Write the SQL code that lists all details of employees with a job code of 502.
5. Write the SQL code to delete the row for the person named William Smithfield, who was hired on June 22, 2004, and whose job code classification is 500. (*Hint*: Use logical operators to include all the information given in this problem.)
6. List the names of all employees who work in department 508.
7. Add a new column named salary in employee table.
8. List names and salaries of all employee ordered by salary.
9. List the name of employees whose salary is between 30000 and 50000.
10. List the name of employees who lives in Houston.
11. List department number and number of employees in each department, ordered by number of employees in each department
12. List department number and number of employees in departments that have more than 2 employees, ordered by department number.

13. List the emp_id of employees who works on project 3388 or project 1945.
14. list department with their manager name(join)
15. List the name of all female employees.
16. List the first name of all employee whose last name begins with letter ‘sm’
17. Find the total no of departments.
18. Find the name of senior most employee (max(hire date)
19. Display from the Employees table the first name (fname), last name (lname), employeeID(emp_id) and job level (job_lvl) columns for those employees with a job level greater than 200; and rename the column headings to: “First Name,” “Last Name,” “IDENTIFICATION#” and “Job Level.”
20. Show all the different projects for which employee work. Display only projects in which more than four employees are employed.
21. find emp_id of all employees working in the project in department named research
22. list employees who joined on the date on which ‘john’ joined.
23. Find the emp_id who works on project named ‘projectF’
24. list the name of female dependents of employee named ‘maria’
25. Execute query 23 using join.
26. List employee details along with their dependent’s details(use join)
27. List employee details along with their dependent’s details and also include employees those do not have dependents
28. List employees with their supervisor name.
29. Change the name of table employee to employee_details
30. List the name of employees who doesn't has supervisor
31. increase salary of employee with emp_id 5 by 10%

32 delete all the tables.

Section -C

1. Study and implementation of basic controls and their properties of Visual Basic 6.0 with help of designing simple forms.
2. Design a form for entering, storing and displaying employee details in employee table mentioned in question no. 2.

SQL Quick Reference

	Syntax
AND / OR	SELECT column_name(s) FROM table_name WHERE condition AND|OR condition
ALTER TABLE	ALTER TABLE table_name ADD column_name datatype or ALTER TABLE table_name DROP COLUMN column_name
AS (alias)	SELECT column_name AS column_alias FROM table_name or SELECT column_name FROM table_name AS table_alias
BETWEEN	SELECT column_name(s) FROM table_name WHERE column_name BETWEEN value1 AND value2
CREATE DATABASE	CREATE DATABASE database_name
CREATE TABLE	CREATE TABLE table_name (column_name1 data_type, column_name2 data_type, column_name2 data_type, ...

	)
DELETE	DELETE FROM table_name WHERE some_column=some_value or DELETE FROM table_name (Note: Deletes the entire table!!) DELETE * FROM table_name (Note: Deletes the entire table!!)
DROP DATABASE	DROP DATABASE database_name
DROP TABLE	DROP TABLE table_name
GROUP BY	SELECT column_name, aggregate_function(column_name) FROM table_name WHERE column_name operator value GROUP BY column_name
HAVING	SELECT column_name, aggregate_function(column_name) FROM table_name WHERE column_name operator value GROUP BY column_name HAVING aggregate_function(column_name) operator value
IN	SELECT column_name(s) FROM table_name WHERE column_name IN (value1,value2,..)
INSERT INTO	INSERT INTO table_name VALUES (value1, value2, value3,....) *or* INSERT INTO table_name (column1, column2, column3,...) VALUES (value1, value2, value3,....)
INNER JOIN	SELECT column_name(s) FROM table_name1 INNER JOIN table_name2 ON table_name1.column_name=table_name2.column_name
LEFT JOIN	SELECT column_name(s) FROM table_name1 LEFT JOIN table_name2 ON table_name1.column_name=table_name2.column_name
RIGHT JOIN	SELECT column_name(s) FROM table_name1 RIGHT JOIN table_name2 ON table_name1.column_name=table_name2.column_name
FULL JOIN	SELECT column_name(s) FROM table_name1 FULL JOIN table_name2 ON table_name1.column_name=table_name2.column_name
LIKE	SELECT column_name(s) FROM table_name WHERE column_name LIKE pattern
ORDER BY	SELECT column_name(s) FROM table_name ORDER BY column_name [ASC|DESC]
SELECT	SELECT column_name(s) FROM table_name
SELECT *	SELECT * FROM table_name
SELECT DISTINCT	SELECT DISTINCT column_name(s) FROM table_name
SELECT INTO	SELECT * INTO new_table_name [IN externaldatabase]

	FROM old_table_name *or* SELECT column_name(s) INTO new_table_name [IN externaldatabase] FROM old_table_name
SELECT TOP	SELECT TOP number|percent column_name(s) FROM table_name
TRUNCATE TABLE	TRUNCATE TABLE table_name
UNION	SELECT column_name(s) FROM table_name1 UNION SELECT column_name(s) FROM table_name2
UNION ALL	SELECT column_name(s) FROM table_name1 UNION ALL SELECT column_name(s) FROM table_name2
UPDATE	UPDATE table_name SET column1=value, column2=value,... WHERE some_column=some_value
WHERE	SELECT column_name(s) FROM table_name WHERE column_name operator value