# S.Y.B.C.A. ( Sem-3) :

# Practical work :

Students are required to solve following practical work and include them as part of their practical journal.

**Exercise-1:** Create following tables and add five records in both the tables.

**Table-1:** Employee ( Emp_no, Emp_name, age, dept_code, Mngr_no, salary)

**Table-2:** Dept (dept_no, dept_name( ‘Accounts’,’Marketing’,’Finance’), Mngr_no( reference to Employee table’s Emp_no))

- Table-1 contains records related to employees working for the organization.
- Table-2 contains records related to departments of the organization
- Every department has a manager. The manager is also an employee of the organization. Hence, the Mngr_no in Table-2 is referring to the Emp_no of the Employee table.
- Every employee works in a department and reports to a manager. Hence the Mngr_no in Table-1 is code of the manager for the specific department referring the Table-2.

## TASK-1:

1) Create a database MyDB1 and create following tables in that.
2) Create above tables by providing appropriate constraints.
3) Add records in both the tables. The first table contains minimum 10 records of employees.
4) The second table contain 3 records of departments (‘Accounts’, ’Marketing’, ’Finance’)
5) Display names of employees who are working as manager.
6) Display names of employees and their department name.
7) Display name of employees and their department name where they are working.
8) Display name of employees in ascending order based on their age.
9) Display details of employees whose salaries are within range of 18,000 to 38,000.
10) Display name, department_name and salary of managers in ascending order salary wise.
11) Name the employees who are earning more than the average salaries of employees.
12) Name the employees who are earning more than the salary of their manager.
13) Take backup of the tables using GUI and CUI modes.
14) Dump the database using GUI and CUI modes.

## Exercise-2:

**Create following table :**

**Old_Emp** ( Emp_no, Emp_name, age, dept_code, Mngr_no, salary)

**Log_Table**(log_code, emp_no, action(‘update’,’delete’))

## TASK-2:

1) Create above two tables.
2) Write a trigger that copy record from Employee table into Old_Emp table whenever any record is deleted from the Employee table.
3) Create a Log_Table that store information about the action performed on employee table. If any record is deleted or updated from Employee table, the same action is recorded in the Log_Table by storing its emp_no and action. Log_code will be an unique attribute and auto generated.
4) Export the Employee table and Log-Table to CSV file and check the CSV file using CUI mode.
5) Export the Dept table to CSV file using GUI mode.

## Exercise-3:

Create Mydatabase1 database and perform following tasks.

### Task-3:

i)Create a database called Mydatabase1.

ii)Now, establish connection with the database using SQLite code.

iii) Create following table using SQLite and then close the connection.

Student(sid INTEGER Primary key, sname text(20), city text(20),age INTEGER)

(iv) Establish connection with the database Mydatabase1.

(v) Insert ten student records using SQLite code for Student table.

(vi) Display all records of the Student table using cursor.

(vii) Extract name of student, age and city name that are from Ahmedabad.

(viii) Create user defined function to which we pass two parameters : connection and Query. The connection set the connection in the function. The query passed executes inside the function and fetch all records. The records returned by the function are received in the calling program.
(ix) Display all the records returned by the function in the main program.