Course code: 15BCS-1DB21T

Course title: Database Systems

**Practical 1**: **SQL: Getting familiar with SQL, parts of SQL, data types used in SQL**

**Practical 2**: **Exercise on basic DDL commands (Create, Alter, Drop, and Rename) and DML commands (insert, update, delete)**

* Write down general syntax of every command mentioned above.
* Perform the following tasks:

1. Create following tables:

|  |  |
| --- | --- |
| **EMPLOYEE** | |
| E\_id | Varchar(5) |
| ename | Char(7) |
| age | Number(2,0) |
| dob | date |
| salary | Number(6,0) |
| city | Char(7) |
| dno | Varchar(5) |

|  |  |
| --- | --- |
| **Department** | |
| d\_id | Varchar(5) |
| dname | Char(7) |
| location | Char(7) |

|  |  |
| --- | --- |
| **student** | |
| s\_id | Varchar(5) |
| sname | Char(7) |

|  |  |
| --- | --- |
| **Works** | |
| E\_id | Varchar(5) |
| d\_id | Varchar(5) |
| hours | Int |

1. Insert five rows each in table.
2. Increase the size of column city to 10 in the table Employee.
3. Rename the table department to office.
4. Drop the table student.
5. Delete the data of employee working in department number 1.
6. Increase the salary by 10% of those employees having salary less than 10000.
7. Add a column State with char (10) in the table Employee.
8. Now insert values for the newly added column state in the employee table.
9. Change the name of city in employee to address
10. Add a column manager\_id with varchar (5) in the table Department.
11. Now insert values for the newly added column manager\_id in the Departmenttable.
12. Drop column age from Employee.

**Practical 3**: **Exercise on retrieval of data (using logical operators, relational operators, between operator, in/not in predicate, pattern matching)**

* Write down the general syntax of every operator/predicate mentioned above.
* Perform the following queries:

1. List down the employees who are having salary greater than equal to 20000 and less then equal to 50000.
2. List down the departments located in Noida.
3. List the eid, name of those employees whose name starts with letter a and ends with letter a.
4. List those employees who are having age greater than 30 yrs and are residing in Delhi.
5. List the eid ,age,name of employees living in either Noida, Delhi or Gurgaon using in predicate.
6. List the details of the departments whose location name has second character as a and second last character as a.

**Practical 4**: **Assignment 1**

**Q1. Create the following tables:**

**client\_master**

client\_no-------varchar

name--------------varchar

city-----------------varchar

state---------------varchar

bal\_due----------number

**Product\_master**

Product\_no-------varchar

description --------varchar

Qty\_on\_hand------number

Sell\_price-----------number**2- Insert the following data into these respective tables:**

**Data for Client Master:**

Clientno Name city state bal.due

0001 Ivan Bombay Maharashtra 15000

0002 Vandana Madras Tamilnadu 20000

0003 Pramada Bombay Maharashtra 5000

0004 Basu Bombay Maharashtra 30000

0005 Ravi Delhi null 2000 0

0006 Rukmini Bombay Maharashtra 10000

**Data for Product Master:**

Product No. Desciption Qty Sell price

P00001 1.44floppies 100 500

P03453 Monitors 10 11200

P06734 Mouse 20 1050

P07865 1.22 floppies 100 525

P07868 Keyboards 10 3050

P07885 CD Drive 10 5100

P07965 540 HDD 10 8000

P07975 1.44 Drive 10 1050

P08865 1.22 Drive 5 1050

**Q3:- On the basis of above two tables answer the following Questions:**

1. Find out the names of all the clients.

2. Retrieve the list of names and cities of all the clients.

3. List the various products available from the product\_master table.

4. List all the clients who are located in Bombay.

5. Display the information for client no 0001 and 0002.

6. Find the products with description as '1.44 drive' and '1.22 Drive'.

7) Find all the products whose sell price is greater then 5000.

8) Find the product whose selling price is greater than 2000 and less than or equal to 5000.

9) List the name, city and state of clients not in the state of 'Maharashtra'

10) Change the city of client no c00005 to Bombay

11) Delete all the details of the product where quantity is equal to 100.

12) Add a column telephone to the table client with domain number (10, 0) and add the values for it.

13) Change the size of sell price to 10, 2

14) Find the names of the clients having a as the second letter in their names.

**Practical 5**:  **Functions (scalar/group) & Assignment 2**

* Write down the general syntax of every function mentioned above.

**Assignment 2**

* Consider the previous table and answer the following queries:

1. Count the total number of orders placed
2. Calculate the average price of all the products
3. Determine the maximum and minimum product prices
4. Count the number of products having price greater then equal to 1500
5. Find all the product whose quantity is less than 50
6. Retrieve the city of all clients in lower case letters
7. Update the table client by changing the first letter of client in caps
8. Display the state of client no C0002 in upper case
9. Retrieve the city of client by extracting substring from the city column with m=1,n=4
10. Calculate the length of names of clients having client\_no=C0004
11. Update the table product by adding Rs 1000 to the sell price of all products
12. Display the order no and day on which the clients placed their order
13. Display the month in alphabet and date when the order must be delivered
14. Display the order date in the format “dd-month-yy”
15. Find the date 15 days after today’s date
16. Find the number of days elapsed between today’s date and the delay date of orders placed by the clients.

**Practical 6**: **Integrity Constraint (Domain Integrity, Entity Integrity, Referential Integrity, Key Constraint), Check Constraint and Default Value concept.**

* Write down the general concept of every constraint mentioned above.
* Execute the following queries:

1. Create a table Emp with attributes ename, ecity, salary, enumber with following constraints: enumber is the primary key and should start with ‘E’, ecity should have default values as Delhi.
2. Create another Table Department with attributes did, budget and manager\_id, dname with following constraints: budget should not be less than 10000.
3. Create another Table Works with attributes did and eid& combination of both as primary key
4. Alter table Department and make did as the primary key.
5. Alter table Department and make manager\_id as the foreign key.
6. Alter table Works and make did and eid as foreign key.
7. Create table book with attributes ISBN, title, price, pid with following constraints: ISBN is the primary key, price should be >=20/<=200, title should not have null values.
8. Create table publisher with attributes pid, pname, state with following constraints: pname should not have null values; state should be either Haryana, Tamil nadu, Maharashtra.
9. Alter the table publisher by making pid as primary key.
10. Alter the table book by making pid as foreign key.
11. Remove the foreign key constraint on book.

**Practical 7**: **Set Operations (Union, Intersect, Minus)**

* Create table sailors with attributes-s\_id, s\_name, s\_age, s\_rating with s\_id as primary key.
* Create table reserves with attributes-s\_id, b\_id, day (date) with (s\_id+b\_id) as primary key and s\_id and b\_id as foreign key.
* Create table boats with attributes-b\_id, bname, bcolour with b\_id as primary key.
* Execute the following queries:

1. Find the sid of the sailors who have reserved red boat but not green boat.
2. Find the sid of the sailors who have reserved red boat or green boat.
3. Find the sid of the sailors who have reserved both red boat and green boat.

**Practical 8**: **Joins (Equi join, Self join, Natural join, Outer join) and Nested queries**

* On the above mentioned tables execute the following queries:
* **Using natural join** :

1. Find names of sailors who have reserved boat number 103
2. Find names of sailors who have reserved at least one boat
3. Find names of sailors who have reserved a red boat
4. Find the colors of boat reserved by lubber

* **Using Equi join** :

1. Find names of sailors who have reserved boat number 103
2. Find names of sailors who have reserved at least one boat
3. Find names of sailors who have reserved a red boat
4. Find the colors of boat reserved by lubber

* **Nested queries :**

1. Find the sailors whose rating is better than some sailor called Horatio
2. Find the sailors whose rating is better than every sailor called Horatio
3. Find the sailor with highest rating

* **Self join :**

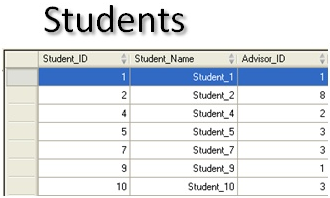
1. Create a table employee (emp\_no, name, manager\_no) where emp\_no is the primary key and manager\_no is the foreign key.
2. Insert following rows:

|  |  |  |
| --- | --- | --- |
| emp\_no | name | manager\_no |
| e01 | basu | e02 |
| e02 | rukmani | e05 |
| e03 | carol | e04 |
| e04 | bayross | null |
| e05 | ivan | null |

1. Execute a query to retrieve names of the employees and the name of their respective managers from the employee table.

* **Outer join :**

1. Create tables and insert these rows:

****

1. Query for retrieving the student details along with their advisory detail
2. Query for retrieving the details of all the student along with details of advisory (using outer join)
3. Query for retrieving the details of student along with details of all the advisory (using outer join)

**Practical 9**: Group by, having clause, construction of English like statements with data from table columns:

* On the tables mentioned in question 7 above execute the following queries:

1. Find the age of youngest sailors for each rating level.
2. Find the age of youngest sailor who is eligible to vote for each rating level with at least two such sailor
3. For each red boat, find the number of reservations for this boat.
4. Find average age of sailor for each rating level that has at least two such sailors.
5. **Write a query to print the following sentence** :

Sailor named ……….. with sailor id……….has reserved boat number………..on date…………….

**Practical 10**: **Views**

* Basic definition and advantages of views.
* Create a table staff with attributes staff\_no, firstname, lastname, position, DOB, salary, branch\_no with staff\_no as primary key.
* Insert six rows in it.
* **Execute the following queries:**

1. Create a view so that manager at branch 200 can see only the details for staff who work in his/her branch office.
2. Create a view staffdetails200 that excludes salary info so that only managers can access the salary details for the staff who work at their branch.

