Problem statement 1(Triggers)

Write a database trigger on Library table. The System should keep track of the records that are being updated or deleted. The old value of updated or deleted records should be added in Library_Audit table.

Problem statement 2

Employee(emp_id, emp_name,salary,designation)

Salary_Backup(rmp_id, old_salary, new_salary, salary_difference)

Create a Trigger to record salary change of the employee. Whenever salary is updated insert the details in Salary_Backup table.

Problem statement 3 (Aggregation & Indexing)

Create the Collection Movies_Data(Movie_ID, Movie_Name, Director, Genre, BoxOfficeCollection) and solve the following:

1. Display a list stating how many Movies are directed by each “Director”.

2. Display list of Movies with the higest BixOfficeCollection in each Genre.

3. Display list of Movies with the higest BoxOfficeCollection in each Genre in ascending order of BoxOfficeCollection.

4. Create an index on field Movie_ID.

5. Create an indwx on fields “Movie_Name” and “Director”.

6. Drop an index on field Movie_ID.

7. Drop an index on fields “Movie_Name” and “Director”.

Problem Statement 4 (Procedures / Functions)

Consider following schema for Bank database.

Account(Account_No, Cust_Name, Balance, NoOfYears)

Earned_Interest(Account_No, Interst_Amt)

1. Write a PL/SQL procedure for following requirement. Take as input Account_No and Interest Rate from User.Calculate the Interest_Amt as simple interest for the given Amount_No and store it in Earned_interest table.Display all the details of Earned_interest Table.
2. Write a PLSQL function to display all records from Account table having Balance greater than 50,000.

Problem Statement 5

Implement MangoDB database connectivity with Java/Python.

Problem Statement 6

Implement MYSQL database connectivity with PHP/Python.

Problem Statement 7

Consider Following Schema

Employee(Employee_id, First_name, Last_name, Hire_date, Salary, Job_title, Manager_id, department_id)

Department(Department_id, Department_name, Manager_id, Location_id)

Locations(Location_id, Street_address, Postal_code, city, state, Country_id)

Manager(Manager_id, Manager_name)

Create the tables with referential integrity. Solve following queries using joins and subqueries.

1. Write a query to find the names(first_name, last_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.
2. Write a query to find the names(first_name, last_name), the salary of the employees who earn the same salary as minimum salary for all departments.
3. Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments.
4. Write a query to display the department name, manager name, and city.
5. Write a query to display the name(first_name, last_name), hire date, salary of all managers whose experience is more than 15 years.

Problem Statement 8(Cursors)

Consider the following schema for Products table.
Products(Product_id, Product_Name, Product_Type, Price)
1.Write a parameterized cursor to display all products in the given price range of price and type ‘Apparel’.
Hint: Take the user input for minimum and maximum price for price range.
2.Write an explicit cursor to display information of all products with Price grater than 5000.
3.Write an implicit cursor to display to display the number of records affected by the update operation incrementing Price of all products by 1000.

Problem Statement 9(DML Using MYSQL)

Create following tables using a given schema and insert appropriate data into the same:

Customer (CustID, Name, Cust_Address, Phone_no, Email_ID, Age)

Account(Account_no,Branch_Name, Address)

Account(Account_no, Branch ID, CustID, date_open, Account_type, Balance)

1.Modify the size of column “Email_Address” to 20 in Customer table

2.Change the column”Email_Address” to Not Null in Customer table.

3.Display the total customers with the balance>50,000Rs.

4.Display average balance for account type=”Saving Account”.

5.Display the customer details that lives in Pune or name starts with’A’.

6.Create a table Saving_Account with(Account_no, Branch ID, CustID, date_open, Balance) using Account Table.

7.Display the customer details Age wise with balance>=20,000Rs.

Problem Statement 10(DDL Using MYSQL)

Create following tables using a given schema and insert appropriate data into the same:

Customer(CustID, Name, Cust_Address, Phone_no, Email_ID, Age)

Branch(Branch ID, Branch_Name, Address)

Account(Account_no, Branch ID, Cust ID, date_open, Account_type, Balance)

1.Create the tables with referential integrity.

2.Draw the ER diagram for the same.

3.Create an Index on primary key column of table Account.

4.Create the view as Customer_info displaying the customer details for age less than 45.

5.Update the view with open date as 16/4/2017

6.Create a sequence on Branch table.

7.Create synonym ‘Branch_info’ for branch table.

Problem Statement 11(CRUD using MangoDB)

Create a collection Social_media having fields as User_Id, User_Name, No_of_Posts, No_of_Friends,Friends_List, Interests.(Hint:Friends_list and Interests can be of array type)

Interest 20 documents in the collection Social_Media.Write queries for following.

1.List all users from collection Social_Media in formatted manner.

2.Find all users having number of posts greater than 100.

3.List the usersnhaving number of posts greater than 100.

4.Display the user ids and Friends list of users who have more than 5 friends.

5.Display all usersnwith no of posts in descending order.

Problem Statement 12 (Map Reduce using MongoDB)

Create Book Collection with(Title, Author_name, Borrowed_status) as fields. Write Map Reduce Functions for following requirements.

1.Display Author wise list of books.

2.Display Author wise list of books having Borrowed status as “True”

3.Display Author wise list of books having price greater than 300.

Problem Statement 13 PL/SQL Stored Procedure and Stored Function.

Write a Stored Procedure namely proc_Grade for the categorization of student. If marks scoredby

students in examination is <=1500 and marks>=990 then student will be placed in distinction

category if marks scored are between 989 and 900 category is first class, if marks 899 and 825

category is Higher Second Class.

Write a PL/SQL block to use procedure created with above requirement.

Stud_Marks(name, total_marks) Result(Roll,Name, Class)

Problem Statement 14 Unnamed PL/SQL code block: Use of Control structure and Exception handling is mandatory.

Consider Tables:

1. Borrower(Roll_no, Name, DateofIssue, NameofBook, Status)

2. Fine(Roll_no,Date,Amt)

Accept Roll_no and Name of Book from user.

Check the number of days (from date of issue).

If days are between 15 to 30 then fine amount will be Rs 5per day.

If no. of days>30, per day fine will be Rs 50 per day and for days less than 30, Rs. 5 per day.

After submitting the book, status will change from I to R.

If condition of fine is true, then details will be stored into fine table.Also handles the exception by named exception handler or user define exception handler.

Problem Statement 15 (Joins & Subqueries using MySQL)

Consider Following Schema

Employee(Employee_id, First_name, Last_name, hire_date, salary, Job_title, manager_id, department_id)

Department(Department_id, Department_name, Manager_id, Location_id)

Locations(location_id, street_address, postal_code, city, state, country_id)

Manager(Manager_id,Manager_name)

Create the tables with referential integrity. Solve following queries using joins and subqueries.

1.Write a query to find the names(first_name, last_name), the salary of employees who earn more than the average salary and who works in any of the IT departments.

2.Write a query to find the names(first_name, last_name) the salary of employees who earn the same salary as the minimum salary for all departments.

3.Write a query to display the employee ID, first name, last name , salary of all employees whose salary is above average for their departments.

4.Write a query to display the department name, manager name and city.

5.Write a query to display thename(first_name, last_name) hire date, salary of all managers whose experience is more than 15 years.
