1	**For the given Library database** **BOOK** (Book_ISBN [PK], Title[Not Null], Publisher_ Name, price[Check Price>0], Date_Of_Publication,Book_Copy), **BOOK_AUTHORS** (Book_ISBN [PK,FK]Author_Name [PK], Author_City) **Solve the following** a) Create view BOOK_AUTHOR_INFO consisting Book_ISBN, Title from BOOK Table and Author_Name from BOOK_AUTHORS Table in ascending order of ISBN no. b) Create an index on Book_Author on table on attribute **"Author_Name"**. c) Create table Book_Auto_Increment (BookID int Auto_increament=100, Book Name) insert five records in table. **d) Delete the book from Book table written by Author 'Korth'.** **e) Select Book Names from table Book whose copies are in between 10 to 15.**
2	**. For the given Library database** **BOOK** (Book_ISBN [PK], Title[Not Null], Publisher_ Name, price[Check Price>0], Date_Of_Publication,Book_Copy), **BOOK_AUTHORS** (Book_ISBN [PK,FK]Author_Name [PK], Author_City) **Solve the following :** **a)** Select Book_ISBN, Title, Author_Name from relations Book and Book_Authors INNER JOIN on attribute Book_ISBN. **b)** Select Book_ISBN, Title, Publisher, Author_Name from relations Book and Book_Authors LEFT OUTER JOIN on attribute Book_ISBN. **c)** Select Book_ISBN, Title, Publisher, Author_Name from relations Book and Book_Authors RIGHT OUTER JOIN on attribute Book_ISBN. **d)** Select Book_ISBN, Title from relation Book whose author is living in City ='Pune'. **e)** Select Book_ISBN, Title from relation Book, which written by more than 2 Authors.
3	**For the given Library database** **BOOK** (Book_ISBN [PK], Title[Not Null], Publisher_ Name, price[Check Price>0], Date_Of_Publication,Book_Copy), **BOOK_AUTHORS** (Book_ISBN [PK,FK]Author_Name [PK], Author_City) **Solve the following** **i)** Display name of publishers as per no of books published by them in ascending order. **ii)** Get publisher names who published at least one book written by author name like 'K%'. **iii)** Get book name and Authors names where book written by maximum authors. **iv)** Get publisher names accordingly books published alphabetically **Find the no of books published in 01 Jan 2014 to till date.**
4	**Consider insurance database with following schema :** **person(driver-id, name, address)** **car(license, model, year)**

	accident (report - no, date, location) **owns(driver-id,license)** **participated(driver-id,car,report-no,damage-amount)** **Write a query in SQL for following requirements :** i) Find the total no. of people who owned cars that were involved in accidents in 2016. ii) Retrieve the name of person whose address contains Pune. iii) Find the name of persons having more than two cars.
5	. Implement MySQL database connectivity with Java Implement Database navigation operations (add, delete, edit,) using ODBC/JDBC.
6	**For the given Employee database** EmployeeInfo(EmpID[PK],EmpFname,EmpLname,Department,Project,Address,DOB,Gender) EmployeePosition(EmpID[FK],EmpPosition,DateOfJoining,Salary) i. Write a query to fetch the EmpFname from the EmployeeInfo table in the upper case and use the ALIAS name as EmpName. ii. Write a query to fetch the number of employees working in the department 'HR'. iii. Write q query to find all the employees whose salary is between 50000 to 100000 iv. Write a query to find the names of employees that begin with 'S' v. Write a query to fetch top N records.
7 **1.**	Write a PL/SQL block of code using parameterized Cursor Merge the data available in the newly created table N_RollCall with the data available in the table O_RollCall. If the data in the first table already exist in the second table then that data should be skipped.2.
8	Write a PL/SQL block of Stored Procedure and Stored Function **proc_Grade** for following problem statement. Write a Stored Procedure namely proc_Grade for the categorization of student. If marks scored by 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.
9	Write a database trigger: Row level and Statement level triggers, Before and After delete or update of database. 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.
10	Write a PL/SQL block of code for the following requirements:- Schema:

	Borrower(Rollin, Name, DateofIssue, NameofBook, Status) **Fine**(Roll_no, Date, Amt) **a.** Accept roll_no & name of book from user. **b.** Check the number of days (from date of issue), if days are between 15 to 30 then fine amount will be Rs 5per day. **c.** If no. of days>30, per day fine will be Rs 50 per day & for days less than 30, Rs. 5 per day. **d.** After submitting the book, status will change from I to R. **e.** If condition of fine is true, then details will be stored into fine table.
11	The organization has decided to increase the salary of employees by 10% of existing salary, whose existing salary is less than Rs. 10000/- Write a PL/SQ block to update the salary as per above requirement, display an appropriate message based on the no. of rows affected by this update (using implicit cursor status variables).
12	Create The following two tables : College-info Faculty-info College-info consists of fields : college-code, college-name, address Faculty-info consists of fields : college-code, faculty-code, faculty-name, qualification, experience-in-no-of-years, address. The field college-code is foreign key. a. Design a form to accept the data from the user. b. Generate queries to do the following : c.List all those faculty members whose experience is greater than or equal to 10 years and have M. Tech degree. d. List all those faculty members, who have at least 10 years of experience but do not have M. Tech degree
13	Create the following table : Student (roll-no, name, subject-name, subject-opted) Subject(faculty-code, faculty-name, specialization) (a) Create a form to accept the data from the user with appropriate validation checks. (b) Generate queries to do the following : (i) Find the number of students who have enrolled for the subject "DBMS". (ii) Find all those faculty members who have not offered any subject.
14	Create the following table : Item (item-code, item-name, qty-in-stock, reorder-level) Supplier (supplier-code, supplier-name, address) Can-supply(supplier-code, item-code) (a) Create a form to accept the data from the user with appropriate validation checks. (b) Generate queries to do the following : (i) List all those suppliers who can supply the given item. (ii) List all those items which cannot be supplied by given company
15	Create the following tables: Student (roll-no, marks, category, district, state) Student-rank(roll-no, marks, rank) (a) Create a form to accept the data from the user with appropriate validation

	checks. (b) Generate queries to do the following : (i) List all those students who have come from Tamilnadu state and secured a rank above 100. (ii) List all those students who come from Andhra Pradesh state and belong to given category who have secured a rank above 100
16	**Create a collection named Book. (book_isbn,title,punlisher_name,author(Name, Address, Phone No[landline, mobile]), publisher_city, price,copies)** i. a. Add 5 documents in the collection with keys **b.** Give details of Books whose Publisher lives in "Pune". **c.** Delete name Book from Book whose name start with "D" **d.** Change the city of publisher "Pearson" to "Pune". **e.** Find the details of publisher named "Pearson".
17	**Create a collection named Book. (book_isbn,title,punlisher_name,author(Name, Address, Phone No[landline, mobile]), publisher_city, price,copies)** **a.** Count the number of documents in the collection. **b.** Arrange the documents in descending order of book_isbn. **c.** Select Book Names whose title is "DBMS" . **d.** Update Book Copies as "10" whose Book Publisher is "Tata MacGraw Hill". **Display name of publishers as per no of books published by them in ascending order.**

18	. Create a collection named **"ORDERS"** that contain documents of the following prototype and solve the following queries:
	{
	cust_id: "abc123",
	ord_date: new Date("Oct 04, 2012"),
	status: 'A',
	price: 50,
	items: [{ sku: "xxx", qty: 25, price: 1 },
	{ sku: "yyy", qty: 25, price: 1 }]
	}
	a. Count all records from `orders`
	b. Sum the `price` field from `orders`
	c. For each unique cust_id, sum the price field.
	d. For each unique cust_id, sum the price field, results sorted by sum.
	For each unique cust_id, ord_date grouping, sum the price field

19	Create a collection named rating that contain 5 documents of the following prototype and solve the following Queries. { movie_id: 123,

	user_id: 12, title: Toy Story(1995), status: 'A' } **a)** Creating an index on movie_id and sorts the keys in the index in ascending order. Verify the query plan **b)** Show various indexes created on movie collection. **c)** Sort movie_id in descending order. **d)** Create a descending order index on movie_id to get ratings related to "Toy Story (1995)" verify the query plan. **e)** Limit the number of items in the result of above query.

20
a)
	Design a map-reduce operations on a collection "orders" that contains documents of the following prototype. Solve the following .
	{
	cust_id: "abc123",
	ord_date: new Date("Oct 04, 2012"),
	status: 'A',
	price: 25,
	gender :'F',
	rating: 1
	}
	a) Count the number of female (F) and male (M) respondents in the orders collection
	b) Count the number of each type of rating (1, 2, 3, 4 or 5) for each orders

21	Create a collection named rating that contain 5 documents of the following prototype and solve the following Queries.
	{
	movie_id: 123,
	user_id: 12,
	title: Toy Story(1995),
	status: 'A'
	}
	a) Get ratings for the movie "ICE AGE(2005)" using the descending ordered index on movie_id and explain.
	b) Rebuild all indexes for the ratings collection.
	c) Drop index on rating collection.
	d) Create an index on movie_id and ratings fields together with movie_id (ascending order sorted) and rating (descending order sorted).
	e) Create a descending order index on movie_id to get ratings related to "Toy Story (1995)" verify the query plan.

22	Implement MongoDb database connectivity with Java Implement Database navigation operations (add, delete, edit,) using ODBC/JDBC.
23	**Create a collection named Book. Add 5 documents in the collection with keys (book_isbn,title,punlisher_name,author(Name, Address, Phone No[landline,**

	mobile]), publisher_city, price,copies) a) Select Book Names whose title is ”DBMS” . b) Update Book Copies as “10” whose Book Publisher is “Tata MacGraw Hill”. c) Display name of publishers as per no of books published by them in ascending order. d) Get publisher names who published at least one book written by author name like ‘K%’. e) **Delete the book from Book table written by Author ‘Korth’.**
24	Consider following structure for MongoDB collections and write a query for following requirements in MongoDB Teachers(Tname, dno, experience, salary, date_of joining) Students(Sname, roll_no, class) i) Write a MongoDB query to create above collections & for insertion of some sample documents. ii) Find the information about all teachers of dno = 2 and having salary greater than or equal to 10,000/- iii) Find the student information having roll_no = 2 or Sname = Anil iv) Display Total no of Students of TE Class V) update salary as 5% increment of teacher whose experience is >10 years.

25	Design a map-reduce operations on a collection “orders” that contains documents of the following prototype. Solve the following .
	{
	cust_id: "abc123",
	ord_date: new Date("Oct 04, 2012"),
	status: 'A',
	price: 25,
	gender :’F’,
	rating: 1
	}
	a) Count the number of female (F) and male (M) respondents in the orders collection
	b) Count the number of each type of rating (1, 2, 3, 4 or 5) for each orders

26	Create the following table with the fields given below : PRODUCT (P_ID, Model, Price, Name, Date_of Manufacture, Date_of Expiry) (a) Display name and date_of expiry of all the products whose price is more than 500. (b) Display name, product_ID and price of all the products whose date_of manufacture is after "01-01-2018". (c) Display name and date_of manufacture and date- of expiry of all the products whose price is between 5,000 and 10,000. (d) Display name, product_ID and model of all the products which are going to expire after two months from today.
27	Create a table named STUDENT with the following fields : 20 (FIRST NAME, MIDDLE NAME, LAST

	NAME, STUDENT_ENRLNO, DATE_OF_BIRTH, CLASS, SECTION, GENDER, YEAR_OF JOIN, ADMISSION_NO, ADDRESS1, ADDRESS2, CITY, STATE, RESPHONE, PIN_CODE) (a) Display all the list of students who are in class - 6, section - A. (b) To display all the students list whose first name starts with "A". (c) To display all the students list who are girls. (d) To display all the students whose YEAR-OF-JOIN is 2000. (e) Sort the records of students with respect to their ADMISSION_NO, in ascending order.
28	Create the following table CATALOG with the following fields : (BOOK ID, BOOK TITLE, AUTHOR, AUTHOR_ID, PUBLISHER_ID, CATEGORY_ID, YEAR, ISBN, PRICE) (a) To display all the books of the CATEGORY_ID : "COMPUTERS". (b) List all the books whose PRICE is greater than or equal to 1000/-. (c) List all the books whose PUBLISHER_ID is "Tata McGraw-Hill". (d) List all the books whose YEAR of publication is 2013. (e) List all the BOOK_TITLEs whose AUTHOR_ID is "123".
29	Create the following tables : Student(roll-no, name, date-of-birth, course-id)Course (Course-id, name, fee, duration, status) (a) Create a form to accept the data from the user with appropriate validationchecks. (b) Write PL/SQL procedure to do the following :Set the status of course to "not offered" in which the number of candidates isless than 5
30	Create the following tables : Student(roll-no, name, date-of-birth, course-id)Course (Course-id, name, fee, duration, status)(a) Create a form to accept the data from the user with appropriate validationchecks.(b) Write PL/SQL procedure to do the following :Set the status of course to "offered" in which the number of candidates is atleast 10 otherwise set it to "not offered"

31	**Structure of 'restaurants' collection:**
	`{`
	`  "address": {`
	`  "building": "1007",`
	`  "coord": [ -73.856077, 40.848447 ],`
	`  "street": "Morris Park Ave",`
	`  "zipcode": "10462"`
	`  },`
	`  "borough": "Bronx",`
	`  "cuisine": "Bakery",`
	`  "grades": [`
	`  { "date": { "$date": 1393804800000 }, "grade": "A", "score": 2 },`
	`  { "date": { "$date": 1378857600000 }, "grade": "A", "score": 6 },`
	`  { "date": { "$date": 1358985600000 }, "grade": "A", "score": 10 },`
	`  { "date": { "$date": 1322006400000 }, "grade": "A", "score": 9 },`
	`  { "date": { "$date": 1299715200000 }, "grade": "B", "score": 14 }`
	`  ],`
	`  "name": "Morris Park Bake Shop",`

	<pre> "restaurant_id": "30075445" }</pre> a.Write a MongoDB query to display the fields restaurant_id, name, borough and cuisine for all the documents in the collection restaurant. b. Write a MongoDB query to display the fields restaurant_id, name, borough and cuisine for all the documents in the collection restaurant. c. Write a MongoDB query to display the fields restaurant_id, name, borough and zip code, but exclude the field _id for all the documents in the collection restaurant.