

1. **a.** Consider the following database consisting of the following tables:

Employee (ssn, first name, last name, gender, designation, doj, address)

Employee-salary (ssn, basic pay, DA, TA, pay)

Department (did, dname, mgrssn)

Employee-department (ssn, deptid)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- i. For each employee, retrieve the employee first name and last name
- ii. Display the distinct department count.
- iii. Retrieve all the information about employees working in Research department including the department information.
- iv. Retrieve the doj, address of employees who work for Research department using joins.
- v. Update the basic pay of each employee with 5%

b. Write a function to find the salary FOR the employee whose department name is passed as an argument (use existing table / create table with appropriate attributes).

2. **a.** Consider the following database consisting of the following tables:

Employee (ssn, first name, last name, gender, designation, doj, address)

Employee-salary (ssn, basic pay, DA, TA, pay)

Department (did, dname, mgrssn)

Employee-department (ssn, deptid)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- i. Retrieve the employee name in the ascending order.
- ii. Find the maximum salary in each department
- iii. Retrieve all the information about employees working in Science department including the department information.
- iv. Retrieve the doj, address of employees who work for Research department using joins.
- v. Update the basic pay of each employee with 5%

b. Write a PL/SQL program to update the commission values for all employees with salary less than 2000 by adding Rs.1000 to existing employees. (Use existing table / create table with appropriate attributes).

3. **a.** Consider the following database consisting of the following tables:

Employee (ssn, first name, last name, gender, designation, doj, address)

Employee-salary (ssn, basic pay, DA, TA, pay)

Department (did, dname, mgrssn)

Employee-department (ssn, deptid)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- i. For each employee, retrieve the employee ssn, first name and dname
- ii. Retrieve the doj, address of employees who work for Research department
- iii. Display the employee ssn who earning second lowest basic pay
- iv. Display the department having employee count > 5.

- v. Update the DA and TA to 3% if the basic pay is >4000.
- b.** Write a PL/SQL program to raise the employee salary by 10%, for department id 30 and also maintain the raised details in the raise table. (use existing table / create table with appropriate attributes).
4. **a.** Consider the following database consisting of the following tables:
 - Employee (employee name, street, city)
 - Works (employee name, company name, salary, doj)
 - Company (company name, city)
 - Manager (employee name, manager name)

(Create the tables with necessary primary and foreign key)
(Enter at least five records for each relation)

 - i. Find the names, streets and cities of residence of all employees who work for First Bank Corporation and earn more than 10,000.
 - ii. Find those companies whose employees earn a higher salary on average than the average salary at First Bank Corporation.
 - iii. Find the names of all employee who earn more than every employee of 'Samll Bank Corporation' (use natural join)
 - iv. Find all employees in database who live in the city chennai and under the manager John.
 - v. Find the employee name who earning the second highest salary.
- b.** Write a PL/SQL program to displaying top 10 employee details based on their salary using cursors(use existing table / create table with appropriate attributes).
5. **a.** Consider the following database consisting of the following tables:
 - Employee (employee name, street, city)
 - Works (employee name, company name, salary, doj)
 - Company (company name, city)
 - Manager (employee name, manager name)

(Create the tables with necessary primary and foreign key)
(Enter at least five records for each relation)

 - i. Find the employees in the database who live in the same cities as the companies for which they work.
 - ii. Find all employees who earn more than the average salary of all employees of that company.
 - iii. Write the query to find the employee name who works in the Compay 'XYX' using joins.
 - iv. Find the no. of employees in each company.
 - v. Update the salary of the employee with 7%.
- b.** Write a PL/SQL procedure raise_sal which increases the salary of an employee. It accepts an employee number and salary increase amount. It uses the employee number to find the current salary from the EMPLOYEE table and update the salary. (use existing table / create table with appropriate attributes).
6. **a.** Consider the following database consisting of the following tables:
 - Department (dept id, dept name)
 - Student (rollno, name, gender, mark1, mark2, mark3, dept id)
 - Staff (staff id, name, designation, qualification, dept id)
 - Tutor (rollno, staff id)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- i. Display the student details who come under the tutor ship of the given staff name "X".
 - ii. How many students are there in CSE department?
 - iii. Display the staff details who work in CSE department.
 - iv. Display the student details who got average >85.
 - v. Display the student name who have secured second lowest total w.r.t each department.
- b.** Write a PL/SQL program to calculate the grade for each student using case statement. Assume your own grade range. (use existing table / create table with appropriate attributes).

7. **a.** Consider the following database consisting of the following tables:

Department (dept id, dept name)

Student (rollno, name, gender, mark1, mark2, mar3, total, average, dept id)

Staff (staff id, name, designation, qualification, dept id)

Tutor (rollno, staff id)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- i. Display the student details who got greater than overall average marks of their department.
- ii. Display the staff details who work in CSE department.
- iii. How many different designations and departments are there?
- iv. Display the no.of. Student under the department 'cse'.
- v. Update the qualification of staff from Assistant Professor to Professor.

b. Write a PL/SQL program to display the list of marks for a student by explicitly specifying his roll_no. (use existing table / create table with appropriate attributes).

8. **a.** Consider the following database consisting of the following tables:

Branch (bname, bcity, assets)

Account (ano, starting date, bname, balance)

Customer (cusid, name, address)

Deposit (ano, cusid, bname)

Loan (lno, banme, amt)

Borrower (cusid, lno)

Transaction (ano, amount, mode, date of trans)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- i. Display the details of the loan whose branch is at 'Salem'
- ii. Find the average account balance at each branch and display only if it is greater than 10000.
- iii. Find the largest account balance in the bank.
- iv. Find the names of all customers who have a loan with lno and amount(joins)
- v. Update the loan amount to 5% if the loan amount is greater than 50000

b. Write a PL/SQL code block that will accept an account number from the user and debit an amount of Rs. 2000 from the account, if the account has a minimum balance of 500 after the amount is debited. (use existing table / create table with appropriate attributes).

9. **a.** Consider the insurance database given below. The primary keys are made bold and the data types are specified.
 PERSON(**driver_id**:string , name:string , address:string)
 CAR(**regno**:string , model:string , year:int)
 ACCIDENT(**report_number**:int , accd_date:date , location:string)
 OWNS(**driver_id**:string , **regno**:string)
 PARTICIPATED(**driver_id**:string , **regno**:string , **report_number**:int , damage_amount:int)
(Create the tables with necessary primary and foreign key)
(Enter at least five records for each relation)
- Find the person who owns more than two cars.
 - Find the total damage amount made by the driver name "XYZ"
 - Update the damage amount to 25000 for the car with a specific Regno in the ACCIDENT table with report number 12.
 - Add a new accident to the database.
 - Find the total number of people who owned cars that were involved in accidents in the year 2012.
- b.** Consider the EMPLOYEE (ENO, SALARY, ENAME) TABLE. Write a procedure raise_sal which increases the salary of an employee. It accepts an employee number AND salary increase amount. It uses the employee number to find the current salary FROM the EMPLOYEE TABLE AND UPDATE the salary.
10. **a.** Consider the following database consisting of the following tables:
 Employee (ssn, first name, last name, gender, designation, doj, address)
 Employee-salary (ssn, basic pay, DA, TA, pay)
 Department (did, dname, mgrssn)
 Employee-department (ssn, deptid)
(Create the tables with necessary primary and foreign key)
(Enter at least five records for each relation)
- For each employee, retrieve the employee first name and last name
 - Display the distinct department count.
 - Retrieve all the information about employees working in Research department including the department information.
 - Retrieve the doj, address of employees who work for Research department using joins.
 - Update the basic pay of each employee with 5%
- b.** Write a PL/SQL program to check whether the given number is ODD or EVEN.
11. **a.** Consider the following database consisting of the following tables:
 Employee (ssn, first name, last name, gender, designation, doj, address)
 Employee-salary (ssn, basic pay, DA, TA, pay)
 Department (did, dname, mgrssn)
 Employee-department (ssn, deptid)
(Create the tables with necessary primary and foreign key)
(Enter at least five records for each relation)
- Retrieve the employee name in the ascending order.
 - Find the maximum salary in each department
 - Retrieve all the information about employees working in Science department including the department information.
 - Retrieve the doj, address of employees who work for Research department using joins.
 - Update the basic pay of each employee with 5%
- b.** Write a PL/SQL program to find the given year is leap year or not.
12. **a.** Consider the following database consisting of the following tables:

- Employee (ssn, first name, last name, gender, designation, doj, address)
 Employee-salary (ssn, basic pay, DA, TA, pay)
 Department (did, dname, mgrssn)
 Employee-department (ssn, deptid)
 (Create the tables with necessary primary and foreign key)
 (Enter at least five records for each relation)
- For each employee, retrieve the employee ssn, first name and dname
 - Retrieve the doj, address of employees who work for Research department
 - Display the employee ssn who earning second lowest basic pay
 - Display the department having employee count > 5.
 - Update the DA and TA to 3% if the basic pay is >4000.
- b.** Write a PL/SQL program to find the factorial of a given number.
13. **a.** Consider the following database consisting of the following tables:
 Employee (employee name, street, city)
 Works (employee name, company name, salary, doj)
 Company (company name, city)
 Manager (employee name, manager name)
 (Create the tables with necessary primary and foreign key)
 (Enter at least five records for each relation)
- Find the names, streets and cities of residence of all employees who work for First Bank Corporation and earn more than 10,000.
 - Find those companies whose employees earn a higher salary on average than the average salary at First Bank Corporation.
 - Find the names of all employee who earn more than every employee of 'Samll Bank Corporation' (use natural join)
 - Find all employees in database who live in the city chennai and under the manager John.
 - Find the employee name who earning the second highest salary.
- b.** Create a TRIGGER to ensure that department TABLE does NOT duplicate VALUES IN dept_no column
14. **a.** Consider the following database for a banking enterprise.
 account(account_number, branch_name, balance)
 branch (branch_name, branch_city, assets)
 customer (customer_name customer_street, customer_city)
 loan (loan_number, branch_name, amount)
 depositor((customer_name, account_number)
 borrower(customer_name, loan_number)
 (Create the tables with necessary primary and foreign key)
 (Enter at least five records for each relation)
- List all branch names and their assests
 - List all accounts of Brooklyn branch
 - Change the size of the branch_city to varchar(20).
 - For all customers who have a loan from the bank, find their names and loan numbers with the attribute loan_number replaced by loan_id.
 - Select the names of customers who have a loan at the bank, and whose names are neither Smith nor Jones
- b.** Write a pl/sql program to find the total and average of 6 subjects then display the grade
15. **a.** Consider the following database consisting of the following tables:
 Party (pid, pname, leader)
 Constituency (cid, cname)
 Contestant (ctid, ctname, ctaddr)
 Election (ctid, number of votes, pname, cname)
 (Create the tables with necessary primary and foreign key)
 (Enter at least five records for each relation)
- Display the contestant details if they secured greater than 10,000 votes.

- ii. Find the number of contestants, constituency wise.
 - iii. Display the winner details in each constituency.
 - iv. Display the difference of votes in each constituency w.r.t first and second position.
 - v. Find the contestant who got the least number of votes
- b.** Write a pl/sql code block to calculate the area of a circle. Store the radius AND the corresponding area IN an empty TABLE named area, consisting of two columns radius AND area.

16. **a.** Consider the insurance database given below. The primary keys are made bold and the data types are specified.

PERSON(driver_id:string , name:string , address:string)

CAR(regno:string , model:string , year:int)

ACCIDENT(report_number:int , accd_date:date , location:string)

OWNS(driver_id:string , regno:string)

PARTICIPATED(driver_id:string , regno:string , report_number:int , damage_amount:int)

- i. Create the above tables by properly specifying the primary keys and foreign keys.
- ii. Enter at least five tuples for each relation.
- iii. Demonstrate how you a.Update the damage amount for the car with specific regno in the accident with report number 12 to 25000.
- iv. Add a new accident to the database.
- v. Find the total number of people who owned cars that were involved in accidents in the year 2008.

b. Write a function to find the salary for the employee whose department name is passed as an argument (use existing table / create table with appropriate attributes).

17. **a.** Consider the following relations for a order processing database application in a company.

CUSTOMER(custno:int , cname:string , city:string)

ORDER(orderno:int , odate:date , custno:int , ord_amt:int)

ORDER_ITEM(orderno:int , itemno:int , quantity:int)

ITEM(itemno:int , unitprice:int)

SHIPMENT(orderno:int , warehouseno:int , ship_date:date)

WAREHOUSE(warehouseno:int , city:string)

- i. Create the above tables by properly specifying the primary keys and foreign keys.
- ii. Enter at least five tuples for each relation.
- iii. Produce a listing: custname , No_of_orders , Avg_order_amount , where the middle column is the total number of orders by the customer and the last column is the average order amount for that customer.
- iv. List the orderno for orders that were shipped from all the warehouses that the company has in a specific city.
- v. Demonstrate the deletion of an item from the ITEM table and demonstrate a method of handling the rows in the ORDER_ITEM table that contains this particular item.

b. Write a PL/SQL program to update the commission values for all employees with salary less than 2000 by adding Rs.1000 to existing employees. (Use existing table / create table with appropriate attributes).

18.**a.** Consider the following database of student enrollment in courses and books adopted for that course.

STUDENT(regno:string , name:string , major:string , bdate:date)

COURSE (courseno:int , cname:string , dept:string)

ENROLL(regno:string , courseno:int , sem:int , marks:int)

BOOK_ADOPTION(courseno:int , sem:int , book_isbn:int)

TEXT(book_isbn:int , book_title:string , publisher:string , author:string)

- i. Create the above tables by properly specifying the primary keys and foreign keys.
- ii. Enter atleast five tuples for each relation.
- iii. Demonstrate how you add a new text book to the database and make this book to be adopted by some department.
- iv. Produce a list of text books (includes courseno , book_isbn , book_title) in the alphabetical order for courses offered by the 'CS' department that use more than two books.
- v. List any department that has all its books published by a specific publisher.

b. Write a PL/SQL program to raise the employee salary by 10%, for department id 30 and also maintain the raised details in the raise table. (use existing table / create table with appropriate attributes).

19. **a.** Consider the following database consisting of the following tables:

Employee (employee name, street, city)

Works (employee name, company name, salary, doj)

Company (company name, city)

Manager (employee name, manager name)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- vi. Find the names, streets and cities of residence of all employees who work for First Bank Corporation and earn more than 10,000.
- vii. Find those companies whose employees earn a higher salary on average than the average salary at First Bank Corporation.
- viii. Find the names of all employee who earn more than every employee of 'Samll Bank Corporation' (use natural join)
- ix. Find all employees in database who live in the city chennai and under the manager John.
- x. Find the employee name who earning the second highest salary.

b. Write a PL/SQL program to displaying top 10 employee details based on their salary using cursors(use existing table / create table with appropriate attributes).

20. **a.** Consider the following database consisting of the following tables:

Employee (Name, SSN, Address, Sex, Salary, Dno)

Department (Dname, Dnumber, MGRSSN, MGRSTART Date)

Dept-Locations (Dnumber, Dlocations)

Project (Pname, Pnumber, Plocations, Dnum)

Works-On (ESSN, PNo, Hours)

Dependent (ESSN, Dependent-name, Sex, Bdate, Relationship)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

Give the queries in SQL

- i. Retrieve the names and address of employees who work for "Research"Department.
- ii. List all the project names on which employee "Smith" is working.
- iii. Retrieve all employees who either work in department 4 or make over 25000 per year or work in department 5 and make over 30,000.

- iv. Retrieve the SSN of all employees who either work in department 5 or directly supervise an employee who works in department number
- v. Retrieve names of each employee who have only daughter dependent.

b. Write a PL/SQL procedure raise_sal which increases the salary of an employee. It accepts an employee number and salary increase amount. It uses the employee number to find the current salary from the EMPLOYEE table and update the salary. (use existing table / create table with appropriate attributes).

21. **a.** Assume the three relations given below and write the queries

STUDENT whose attributes are Stud No and StudName,

ASSIGNED_TO whose attributes are Stud No and Project No

PROJECT whose attributes are Project No and Project area.

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- i. Obtain Stud No and Stud Name of Name of all those students who are working on all those students who are working on database projects.
- ii. Obtain Stud No and Stud Name of all those students who are working on both the projects having project numbers.
- iii. Obtain Stud No and Stud Name of all those students who do not work on the project number
- iv. Obtain Stud No and Stud Name of all students other than the students with Stud No 54 who work on at least one project.
- v. Alter the table by adding column student dept in student table.

b. Write a PL/SQL program to calculate the grade for each student using case statement. Assume your own grade range. (use existing table / create table with appropriate attributes).

22. **a.** Consider the following schema and write the queries for the g given below:

SAILORS(Sid, Sname, rating, age)

BOATS(bid, bname, color)

RESERVES(sid, bid, day)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- i. Find names of sailors who reserved green boat
- ii. Find the colors of boats reserved by "Ramesh"
- iii. Find names of sailors who have reserved a red or a green boat.
- iv. Find the "sids" of sailors with age over 20 who have not registered a red boat.
- v. Find the maximum number or registrations for the boat which is red boat or green boat

b. Write a PL/SQL program to display the list of marks for a student by explicitly specifying his roll_no. (use existing table / create table with appropriate attributes).

23. **a.** Consider the following schema

Emp (eid: integer, ename: string, age: integer, sal: real)

Works (eid: integer, pid: integer, no-of-hrs: integer, did: integer)

Dept (did: integer, dname:string, mgrid: integer)

Project (pid: integer, Pname: string)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

Write SQL statement to

- i. Give every employee of did = '6' and 10% raise in salary.

- ii. Add 'John' as an employee with eid = '99', age = '30', and salary = '15,000'.
- iii. Delete the 'Research' department and explain what happens when this statement is executed.
- iv. Write the query to find the employee name who works in the project 'XYX' using joins.
- v. Display all data of employee who work for projects in more than one departments

b. Write a PL/SQL code block that will accept an account number from the user and debit an amount of Rs. 2000 from the account, if the account has a minimum balance of 500 after the amount is debited. (use existing table / create table with appropriate attributes).

24. **a.** The following are maintained by a book dealer.

AUTHOR(author_id:int , name:string , city:string , country:string)

PUBLISHER(publisher_id:int , name:string , city:string , country:string)

CATALOG(book_id:int , title:string , author_id:int , publisher_id:int , category_id:int , year:int , price:int)

CATEGORY(category_id:int , description:string)

ORDER_DETAILS(order_no:int , book_id:int , quantity:int)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

Write SQL statement to

- i. Give the details of the authors who have 2 or more books in the catalog and the price of the books is greater than the average price of the books in the catalog and the year of publication is after 2000.
- ii. Find the author of the book that has maximum sales.
- iii. Demonstrate how you increase the price of books published by a specific publisher by 10%.
- iv. Find the author who published in the country "India" and publisher "Tata MachrwHill" in the category "FICTION".
- v. Find the number of quantities for the book id "5555".

b. Consider the EMPLOYEE (ENO, SALARY, ENAME) TABLE. Write a procedure raise_sal which increases the salary of an employee. It accepts an employee number AND salary increase amount. It uses the employee number to find the current salary FROM the EMPLOYEE TABLE AND UPDATE the salary.

25. **a.** Consider the following database for a banking enterprise.

account(account_number, branch_name, balance)

branch (branch_name, branch_city, assets)

customer (customer_name customer_street, customer_city)

loan (loan_number, branch_name, amount)

depositor((customer_name, account_number)

borrower(customer_name, loan_number)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- i. List all loans with amount > 1000.
- ii. List all accounts of Perryridge branch with balance < 1000.
- iii. Add a column phoneNo to customer table.
- iv. Find all customers who have an account but no loan at the bank.
- v. Find those branches where the average accounts balance is more than Rs. 1200.

- b.** Write a PL/SQL code block to display the customer details having both an account and a loan in the bank (use existing table / create table with appropriate attributes).
26. **a.** Consider the following database for a banking enterprise.
- account(account_number, branch_name, balance)
 branch(branch_name, branch_city, assets)
 customer(customer_name customer_street, customer_city)
 loan(loan_number, branch_name, amount)
 depositor((customer_name, account_number)
 borrower(customer_name, loan_number)
- (Create the tables with necessary primary and foreign key)*
(Enter at least five records for each relation)
- List all branch names and their assests
 - List all accounts of Brooklyn branch
 - Change the size of the branch_city to varchar(20).
 - For all customers who have a loan from the bank, find their names and loan numbers with the attribute loan_number replaced by loan_id.
 - Select the names of customers who have a loan at the bank, and whose names are neither Smith nor Jones
- b.** Write a PL/SQL code block that will accept an account number from the user and debit an amount of Rs. 2000 from the account, if the account has a minimum balance of 500 after the amount is debited. (use existing table / create table with appropriate attributes).
27. **a.** Consider the following database consisting of the following tables:
- Party(pid, pname, leader)
 Constituency(cid, cname)
 Contestant(ctid, ctname, ctaddr)
 Election(ctid, number of votes, pname, cname)
- (Create the tables with necessary primary and foreign key)*
(Enter at least five records for each relation)
- Display the contestant details if they secured greater than 10,000 votes.
 - Find the number of contestants, constituency wise.
 - Display the winner details in each constituency.
 - Display the difference of votes in each constituency w.r.t first and second position.
 - Find the contestant who got the least number of votes
- b.** Create a TRIGGER that raises an user defined error message and does not allow update and insertion. (use existing table / create table with appropriate attributes).
28. **a.** Consider the following database consisting of the following tables:
- Employee(ssn, first name, last name, gender, designation, doj, address)
 Employee-salary(ssn, basic pay, DA, TA, pay)
 Department(did, dname, mgrssn)
 Employee-department(ssn, deptid)
- (Create the tables with necessary primary and foreign key)*
(Enter at least five records for each relation)
- For each employee, retrieve the employee ssn, first name and dname
 - Retrieve the doj, address of employees who work for Research department
- b.** Create emp_table, and backup_tbl with the following attributes
- emp_table(empid, empname, salary)
 backup_tbl(empid, empname, salary, operation)

Write a trigger function which will be invoked before the insert, delete or update operation.

It does the following:

- i. Before delete operation, it inserts the old data into backup_tbl.
- ii. Before update operation, it inserts the old data into backup_tbl.

29. a. Consider the following database consisting of the following tables:

Department (dept id, dept name)

Student (rollno, name, gender, mark1, mark2, mark3, dept id)

Staff (staff id, name, designation, qualification, dept id)

Tutor (rollno, staff id)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- i. Display the student details who come under the tutor ship of the given staff name "X".
- ii. How many students are there in CSE department?

b. Create emp_table, and backup_tbl with the following attributes

emp_table(empid, empname, salary)

backup_tbl(empid, empname, salary, operation)

Write a trigger function which will be invoked before the insert, delete or update operation.

It does the following:

- i. Before delete operation, it inserts the old data into backup_tbl.
- ii. Before insert operation, it inserts the new data into backup_tbl.

30. a. Consider the following database consisting of the following tables:

Employee (ssn, first name, last name, gender, designation, doj, address)

Employee-salary (ssn, basic pay, DA, TA, pay)

Department (did, dname, mgrssn)

Employee-department (ssn, deptid)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- i. For each employee, retrieve the employee first name and last name
- ii. Display the distinct department count.
- iii. Retrieve all the information about employees working in Research department including the department information.
- iv. Retrieve the doj, address of employees who work for Research department using joins.
- v. Update the basic pay of each employee with 5%

b. Write a PL/SQL program to check whether the given employee number is odd or even.

31. a. Consider the following database consisting of the following tables:

Employee (ssn, first name, last name, gender, designation, doj, address)

Employee-salary (ssn, basic pay, DA, TA, pay)

Department (did, dname, mgrssn)

Employee-department (ssn, deptid)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

- i. For each employee, retrieve the employee ssn, first name and dname
- ii. Retrieve the doj, address of employees who work for Research department
- iii. Display the employee ssn who earning second lowest basic pay
- iv. Display the department having employee count > 5.
- v. Update the DA and TA to 3% if the basic pay is >4000.

b. Write a PL/SQL program to raise the employee salary by 10%, for department id 30 and also maintain the raised details in the raise table. (use existing table / create table with appropriate attributes).

32. a. Consider the following database consisting of the following tables:

Employee (ssn, first name, last name, gender, designation, doj, address)

Employee-salary (ssn, basic pay, DA, TA, pay)

Department (did, dname, mgrssn)

Employee-department (ssn, deptid)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

iv. For each employee, retrieve the employee first name and last name

v. Display the distinct department count.

vi. Retrieve all the information about employees working in Research department including the department information.

iv. Retrieve the doj, address of employees who work for Research department using joins.

v. Update the basic pay of each employee with 5%

b. Write a PL/SQL program to update the commission values for all employees with salary less than 2000 by adding Rs.1000 to existing employees. (Use existing table / create table with appropriate attributes).

33. a. Consider the following database for a banking enterprise.

account(account_number, branch_name, balance)

branch (branch_name, branch_city, assets)

customer (customer_name customer_street, customer_city)

loan (loan_number, branch_name, amount)

depositor((customer_name, account_number)

borrower(customer_name, loan_number)

(Create the tables with necessary primary and foreign key)

(Enter at least five records for each relation)

iv. List all branch names and their assets

v. List all accounts of Brooklyn branch

vi. Change the size of the branch_city to varchar(20).

iv. For all customers who have a loan from the bank, find their names and loan numbers with the attribute loan_number replaced by loan_id.

vi. Select the names of customers who have a loan at the bank, and whose names are neither Smith nor Jones

b. Write a PL/SQL program to find the given year is leap year or not.

34. a. Consider the following database of student enrollment in courses and books adopted for that course.

STUDENT(regno:string , name:string , major:string , bdate:date)

COURSE (courseno:int , cname:string , dept:string)

ENROLL(regno:string , courseno:int , sem:int , marks:int)

BOOK_ADOPTION(courseno:int , sem:int , book_isbn:int)

TEXT(book_isbn:int , book_title:string , publisher:string , author:string)

i. Create the above tables by properly specifying the primary keys and foreign keys.

ii. Enter atleast five tuples for each relation.

iii. Demonstrate how you add a new text book to the database and make this book to be adopted by some department.

- iv. Produce a list of text books (includes courseno , book_isbn , book_title) in the alphabetical order for courses offered by the 'CS' department that use more than two books.
 - v. List any department that has all its books published by a specific publisher.
- b.** Write a pl/sql code block to calculate the area of a circle. Store the radius AND the corresponding area IN an empty TABLE named area, consisting of two columns radius AND area
35. **a.** Consider the following database consisting of the following tables:
- Employee (ssn, first name, last name, gender, designation, doj, address)
 - Employee-salary (ssn, basic pay, DA, TA, pay)
 - Department (did, dname, mgrssn)
 - Employee-department (ssn, deptid)
- (Create the tables with necessary primary and foreign key)
- (Enter at least five records for each relation)
- i. For each employee, retrieve the employee ssn, first name and dname
 - ii. Retrieve the doj, address of employees who work for Research department
 - iii. Display the employee ssn who earning second lowest basic pay
 - iv. Display the department having employee count > 5.
 - v. Update the DA and TA to 3% if the basic pay is >4000.
- b.** Consider the EMPLOYEE (ENO, SALARY, ENAME) TABLE. Write a procedure raise_sal which increases the salary of an employee. It accepts an employee number AND salary increase amount. It uses the employee number to find the current salary FROM the EMPLOYEE TABLE AND UPDATE