1. Write a stored function in PL/SQL for given requirement and use the same in PL/SQL block. Account no. and branch name will be accepted from user. The same will be searched in table acct\_details. If status of account is active then display appropriate message and also store the account details in active\_acc\_details table, otherwise display message on screen “account is inactive”.
2. Write a PL/SQL stored Procedure for following requirements and call the procedure in appropriate PL/SQL block. 1. Borrower(Rollin, Name, DateofIssue, NameofBook, Status) 2. Fine(Roll\_no,Date,Amt) Accept roll\_no• & 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 & 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.

Implicit Cursor

1. The bank manager has decided to activate all those accounts which were previously marked as inactive for performing no transaction in last 365 days. Write a PL/SQ block (using implicit cursor) to update the status of account, display an approximate message based on the no. of rows affected by the update. (Use of %FOUND, %NOTFOUND, %ROWCOUNT)

EXPLICIT CURSOR:

1. Organization has decided to increase the salary of employees by 10% of existing salary, who are having salary less than average salary of organization, Whenever such salary updates takes place, a record for the same is maintained in the increment\_salary table. EMP (E\_no , Salary) increment\_salary(E\_no , Salary)
2. Write PL/SQL block using explicit cursor for following requirements: College has decided to mark all those students detained (D) who are having attendance less than 75%. Whenever such update takes place, a record for the same is maintained in the D\_Stud table. create table stud21(roll number(4), att number(4), status varchar(1)); create table d\_stud(roll number(4), att number(4))

Parameterized Cursor

1. Write a PL/SQL block of code using parameterized Cursor, that will 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.

Parameterized Cursor

1. Write the PL/SQL block for following requirements using parameterized Cursor: Consider table EMP(e\_no, d\_no, Salary), department wise average salary should be inserted into new table dept\_salary(d\_no, Avg\_salary)
2. Write PL/SQL block using explicit cursor: Cursor FOR Loop for following requirements: College has decided to mark all those students detained (D) who are having attendance less than 75%. Whenever such update takes place, a record for the same is maintained in the D\_Stud table. create table stud21(roll number(4), att number(4), status varchar(1)); create table d\_stud(roll number(4), att number(4));
3. Consider table Stud(Roll, Att,Status) Write a PL/SQL block for following requirement and handle the exceptions. Roll no. of student will be entered by user. Attendance of roll no. entered by user will be checked in Stud table. If attendance is less than 75% then display the message “Term not granted” and set the status in stud table as “D”. Otherwise display message “Term granted” and set the status in stud table as “ND”
4. Account(Acc\_no, branch\_name,balance)

branch(branch\_name,branch\_city,assets)

customer(cust\_name,cust\_street,cust\_city)

Depositor(cust\_name,acc\_no)

Loan(loan\_no,branch\_name,amount)

Borrower(cust\_name,loan\_no)

**Solve following query:**

Create above tables with appropriate constraints like primary key, foreign key, check constrains, not null etc.

Q1. Find the names of all branches in loan relation.

Q2. Find all loan numbers for loans made at Akurdi Branch with loan amount > 12000.

Q3. Find all customers who have a loan from bank. Find their names,loan\_no and loan amount.

Q4. List all customers in alphabetical order who have loan from Akurdi branch.

Q5. Find all customers who have an account or loan or both at bank.

1. Account(Acc\_no, branch\_name,balance)

branch(branch\_name,branch\_city,assets)

customer(cust\_name,cust\_street,cust\_city)

Depositor(cust\_name,acc\_no)

Loan(loan\_no,branch\_name,amount)

Borrower(cust\_name,loan\_no)

Solve following query

Q1. Find all customers who have both account and loan at bank.

Q2. Find all customer who have account but no loan at the bank.

Q3. Find average account balance at Akurdi branch.

Q4. Find the average account balance at each branch

Q5. Find no. of depositors at each branch.

1. Create following Tables

cust\_mstr(cust\_no,fname,lname)

add\_dets(code\_no,add1,add2,state,city,pincode)

**Retrieve the address of customer Fname as 'xyz' and Lname as 'pqr'**

Create following Tables

cust\_mstr(custno,fname,lname)

acc\_fd\_cust\_dets(codeno,acc\_fd\_no)

fd\_dets(fd\_sr\_no,amt)

**List the customer holding fixed deposit of amount more than 5000**

1. Create following Tables

emp\_mstr(e\_mpno,f\_name,l\_name,m\_name,dept,desg,branch\_no)

branch\_mstr(name,b\_no)

**List the employee details along with branch names to which they belong**

Create following Tables

emp\_mstr(emp\_no,f\_name,l\_name,m\_name,dept)

cntc\_dets(code\_no,cntc\_type,cntc\_data)

**List the employee details along with contact details using left outer join & right join**

1. CREATE TABLE countries (

  code CHAR(2) NOT NULL,

  year INT NOT NULL,

  gdp\_per\_capita DECIMAL(10, 2) NOT NULL,

  govt\_debt DECIMAL(10, 2) NOT NULL

)

Solve following query

1. What are the top 3 average government debts in percent of the GDP for those countries whose GDP per capita was over 40’000 dollars in every year in the last four years
2. Are there any countries at all with a GDP per capita of more than 50’000 dollars?

|  |
| --- |
| **CUSTOMER** |
| Id |
| FirstName |
| LastName |
| City |
| Country |
| Phone |

1. Consider the above table and solve the query

* List the number of customers in each country. Only include countries with more than 10 customers.
* List the number of customers in each country, except the USA, sorted high to low. Only include countries with 9 or more customers

1. Account(Acc\_no, branch\_name,balance)

branch(branch\_name,branch\_city,assets)

customer(cust\_name,cust\_street,cust\_city)

Depositor(cust\_name,acc\_no)

Loan(loan\_no,branch\_name,amount)

Borrower(cust\_name,loan\_no)

* Solve following query

Q1. Find the branches where average account balance > 12000.

Q2. Find number of tuples in customer relation.

Q3. Calculate total loan amount given by bank.

Q4. Delete all loans with loan amount between 1300 and 1500.

Q5. Delete all tuples at every branch located in Nigdi.

1. Database Trigger (All Types: Row level and Statement level triggers, Before and After 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.
2. Unnamed PL/SQL code block: Use of Control structure and Exception handling is mandatory. Write a PL/SQL block of code for the following requirements:- Schema:

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

2. Fine(Roll\_no,Date,Amt)

1. Accept roll\_no & name of book from user.
2. Check the number of days (from date of issue), if days are between 15 to 30 then fine amount will be Rs 5per day.
3. If no. of days>30, per day fine will be Rs 50 per day & for days less than 30, Rs. 5 per day.
4. After submitting the book, status will change from I to R.
5. If condition of fine is true, then details will be stored into fine table.
6. Cursors: (All types: Implicit, Explicit, Cursor FOR Loop, Parameterized Cursor)

Write a PL/SQL block of code using parameterized Cursor, that will 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.

1. PL/SQL Stored Procedure

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 and900 category is first class, if marks 899 and 825 category is Higher Second Class Write a PL/SQL block for using procedure created with above requirement. Stud\_Marks(name, total\_marks) Result(Roll,Name, Class)

1. Write a PL/SQL block for following requirement using user defined exception handling. The account\_master table records the current balance for an account, which is updated whenever, any deposits or withdrawals takes place. If the withdrawal attempted is more than the current balance held in the account. The user defined exception is raised, displaying an appropriate message. Write a PL/SQL block for above requirement using user defined exception handling.
2. Design and Develop MongoDB Queries using CRUD operations. (Use CRUD operations, SAVE method, logical operators)
3. Implement aggregation and indexing with suitable example using MongoDB
4. Implement Map reduces operation with suitable example using MongoDB.
5. Design and Implement any 5 query using MongoDB

|  |  |
| --- | --- |
|  |  |

1. Design and Develop MongoDB Queries using CRUD operations. (Use CRUD operations, SAVE method, logical operators)