Assessment :

Pseudo code 10 Marks

Q1 : Write a Pseudo code for verify if number entered is prime number.

Ans:

|  |
| --- |
| BEGIN  DECLARE GivenNumber, a, b=0 as number  GET number from user and store in GivenNumber  IF GivenNumber =0  Print “Number 0 is not prime”  ELSE IF GivenNumber <0  Print “Please enter a positive number”  ELSE IF GivenNumber = 1  Print “Number 1 is a Prime Number”  ELSE  For a=1, a<=GivenNumber by 1 do  IF GivenNumber mod a = 0  b=b+1  ELSE  b=b  ENDIF  END For  IF b=2  Print GivenNumber + “is a prime number”  ELSE  Print GivenNumber + “is NOT a prime number”  ENDIF  ENDIF    END |

Q2: Write a Pseudo code for transferring amount from one account to another. [Should validate whether both accounts exists]

Ans:

|  |
| --- |
| BEGIN  GET Account 1 to transfer amount from and store it in Account1  IF Account1 is valid  GET Account 2 to transfer amount to and store it in Account2  IF Account2 is valid  GET Amount to be transferred and store it in Amount  IF Amount <=0  PRINT "Are you serious?"  ELSE  GET Balance of Account1 and Account2 and store in BalAcc1 and BalAc2  IF Amount > BalAcc1  PRINT "You are not that rich"  ELSE  Transfer Amount  SET BalAcc1 = BalAcc1 - Amount  PRINT "New balance in " + Account1 + " is " + BalAcc1  Set BalAcc2 = BalAcc2 + Amount  PRINT "New balance in " + Account2 + " is " + BalAcc2  ENDIF  ENDIF  ELSE  PRINT "Enter a valid Account number in Account 2"  ENDIF  ELSE  PRINT "Enter a valid Account number in Account 1"  ENDIF  END |

RDBMS: 30 Marks

Q1. Write SQL Query to create following tables [DO NOT CREATE PRIMARY / FOREIGN KEYS ]

Customer: CustomerId, Fullname, address, city, pan number

Account: accountNo, accType, balance, customerId

Ans:

|  |
| --- |
| CREATE TABLE CUSTOMER (  CustomerId varchar2(5) unique not null,  Fullname varchar2 (20) not null,  Address varchar2(50) Default 'Please enter your address',  City varchar2(20) not null,  pan\_number varchar2(10) unique not null  );  CREATE TABLE ACCOUNT (  AccountNo number(16) unique not null,  AccType varchar2 (10) not null,  Balance number(10) not null,  CustomerId varchar2(5) not null  ); |

Q2. Write SQL to insert following records in Customer & Account tables:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| C1002 | Rajiv Bhatia | Xyz Path, Chandni chowk | Delhi | AXNSS 1234 A |
| C1003 | Alia Bhatt | Khar | Mumbai | SZAXS 5656 B |
| C1004 | Vijay Deol | Bandra | Mumbai | APOI 5675 A |
| C1005 | Ajay Deol | Bandra | Mumbai | AUIO 7676 K |

|  |  |  |  |
| --- | --- | --- | --- |
| SB122666 | Savings | 67000 | C1002 |
| CB565556 | Current | 786928.98 | C1002 |
| SB876565 | Savings | 547899.90 | C1004 |
| SB565722 | Savings | 67600 | C1003 |
| SB757676 | Savings | 66197.88 | C1003 |
| SB166778 | Current | 16000 | C1008 |

Ans:

|  |
| --- |
| INSERT INTO CUSTOMER (CustomerId,Fullname,Address,City,pan\_number)  Values ('C1002','Rajiv Bhatia','Xyz Path, Chandni chowk','Delhi','AXNSS1234A');  INSERT INTO CUSTOMER (CustomerId,Fullname,Address,City,pan\_number)  Values ('C1003','Alia Bhatt','Khar','Mumbai','SZAXS5656B');  INSERT INTO CUSTOMER (CustomerId,Fullname,Address,City,pan\_number)  Values ('C1004','Vijay Deol','Bandra','Mumbai','APOI5675A');  INSERT INTO CUSTOMER (CustomerId,Fullname,Address,City,pan\_number)  Values ('C1005','Ajay Deol','Bandra','Mumbai','AUIO7676K');  INSERT INTO ACCOUNT (AccountNo,AccType,Balance,CustomerId) Values ('SB122666','Savings','67000','C1002');  INSERT INTO ACCOUNT (AccountNo,AccType,Balance,CustomerId) Values ('CB565556','Current','786928.98','C1002');  INSERT INTO ACCOUNT (AccountNo,AccType,Balance,CustomerId) Values ('SB876565','Savings','547899.90','C1004');  INSERT INTO ACCOUNT (AccountNo,AccType,Balance,CustomerId) Values ('SB565722','Savings','67600','C1003');  INSERT INTO ACCOUNT (AccountNo,AccType,Balance,CustomerId) Values ('SB757676','Savings','66197.88','C1003');  INSERT INTO ACCOUNT (AccountNo,AccType,Balance,CustomerId) Values ('SB166778','Current','16000','C1008'); |

Q3. Write a Left Join to get all customers and accounts. Join should display all customers [Even those who DO NOT have any account].

Ans:

|  |
| --- |
| SELECT \* FROM CUSTOMER LEFT JOIN ACCOUNT USING (CUSTOMERID) ORDER BY CUSTOMERID; |

UML 10 Marks

Q1. For RDBMS Question #1, Create Class Diagram for Both tables.

Ans:

|  |
| --- |
|  |

Q2. Create an Activity Diagram to explain fund transfer.

HINT: Fund transfer is possible from Any account type to any other account type.

Must validate existence of both account

Must validate account balance before transfer

Must update balance after transaction completes.

Ans:

