branch (branch_name, branch_city, assets)
customer (customer_name, customer_street, customer_city)
loan (loan_number, branch_name, amount)
borrower (customer_name, loan_number)
account (account_number, branch_name, balance)
depositor (customer_name, account_number)

1. Find the names of all branches in the loan relations, and remove duplicates
   select distinct branch_name from loan

2. Find the average account balance at the Perryridge branch.
   select avg (balance) from account where branch_name = 'Perryridge'

3. Find the number of tuples in the customer relation.

   select count (*) from customer
4. Find the number of depositors in the bank.

   select count (distinct customer_name) from depositor
5. Find all the branch names where customer “Ramesh” is having account.

Select customer_name from account, depositor where customer_name='Ramesh' and depositor.account_number=account.accout_number

6. List in alphabetic order the names of all customers having a loan in Perryridge branch
   select distinct customer_name from borrower, loan where borrower loan_number = loan.loan_number and branch_name = 'Perryridge' order by customer_name

7. Find the names of all customers whose street includes the substring “Main”.

   select customer_name from customer where customer_street like '% Main%'

8. Find the names of all branches that have greater assets than some branch located in Brooklyn. (Hint: Single table 2 instances)

   select distinct T.branch_name from branch as T, branch as S where T.assets > S.assets and S.branch_city = 'Brooklyn'

9. Find the name, loan number and loan amount of all customers having a loan at the Perryridge branch.

   select customer_name, borrower.loan_number, amount from borrower, loan where borrower.loan_number = loan.loan_number and branch_name = 'Perryridge'

10. Find the loan number of those loans with loan amounts between 90,000 and 100,000.

    select loan_number from loan where amount between 90000 and 100000

11. Find all customers who have an account at all branches located in Brooklyn using exists clause.

select distinct S.customer_name from depositor as S where not exists (
(select branch_name from branch where branch_city = 'Brooklyn')

except
(select R.branch_name from depositor as T, account as R where T.account_number = R.account_number and S.customer_name = T.customer_name))

12. Find all customers who have at most one account at the Perryridge branch using unique clause.

select T.customer_name from depositor as T where unique (select R.customer_name from account, depositor as R where T.customer_name = R.customer_name and R.account_number = account.account_number and account.branch_name = 'Perryridge')

13. Find the names of all branches that have greater assets than all branches located in Brooklyn. (atleast 2 ways)

select branch_name from branch where assets > all (select assets from branch where branch_city = 'Brooklyn')

14. Find all branches that have greater assets than some branch located in Brooklyn.(atleast 2 ways)

select branch_name from branch where assets > some (select assets from branch where branch_city = 'Brooklyn')

15. Find all customers who have both an account and a loan at the Perryridge branch (atleast 3 ways)

(select customer_name from borrower, loan where borrower.loan_number = loan.loan_number and branch_name = 'Perryridge') intersect(select customer_name from depositor, account where depositor.account_number = account.account_number)

select customer_name from borrower, loan where borrower.loan_number = loan.loan_number and branch_name = 'Perryridge and customer_name in (select customer_name from depositor, account where depositor.account_number = account.account_number)

16. Find all customers who have loan and but not an account at the Perryridge branch (atleast 3 ways)

(select customer_name from borrower, loan where borrower.loan_number = loan.loan_number and branch_name = 'Perryridge') exceptt(select customer_name from depositor, account where depositor.account_number = account.account_number)
s

select customer_name from borrower, loan where borrower.loan_number = loan.loan_number and branch_name = 'Perryridge and customer_name not in (select customer_name from depositor, account where depositor.account_number = account.account_number)

17. Find all customers who have account and but not loan at the Perryridge branch (atleast 3 ways)

same as above just exchange inner and outer queries

18. Find all loan number which appear in the loan relation with null values for amount.

select loan_number from loan where amount is null

19. Find the names of all branches where the average account balance is more than 1,200.

select branch_name, avg (balance) from account group by branch_name having avg (balance) > 1200

20. Find the number of depositors for each branch.

select branch_name, count (distinct customer_name) from depositor, account where depositor.account_number = account.account_number group by branch_name

21. Find the branch with highest assets.

select branch_name from branch where assets >= all (select assets from branch)

22. Find the branch with lowest assets.

select branch_name from branch where assets <= all (select assets from branch)

23. Find the branch with fifth highest assets. (Using Corelated Nested Query)
    select b1.branch_name, b1.assets from branch as b1 where 4 = (select count(distinct assets) from branch as b2 where b2.assets>b1.assets)

24. Find the branch with fifth lowest assets. (Using Corelated Nested Query)
    select b1.branch_name, b1.assets from branch as b1 where 4 = (select count(distinct assets) from branch as b2 where b2.assets<b1.assets)

25. Find customer names who have atleast 3 accounts in branches of city “Heritage”.

Select customer_name from account, depositor, branch where branch_city='Heritage' and depositor.account_number=account.accout_number and branch.branch_name=account.branch_name group by customer_name having count(account_number)>2