//Creation of table branch

SQL> create table branch(branch\_name varchar(15), branch\_city varchar(15),assets numeric(10), primary key(branch\_name));

Table created.

SQL> insert into branch values('Brighton' , 'Brooklyn' , 7100000);

1 row created.

SQL> insert into branch values('Downtown' , 'Brooklyn' , 9000000);

1 row created.

SQL> insert into branch values('Mianus' , 'Horseneck' , 400000);

1 row created.

SQL> insert into branch values('North Town' , 'Rye' , 3700000);

1 row created.

SQL> insert into branch values('Perryridge' , 'Horseneck' , 1700000);

1 row created.

SQL> insert into branch values('Pownal' , 'Bennington' , 300000);

1 row created.

SQL> insert into branch values('Redwood' , 'Palo Alto' , 3000000);

1 row created.

SQL> insert into branch values('Round Hill' , 'Horseneck' , 800000);

1 row created.

SQL> select \* from branch;

BRANCH\_NAME BRANCH\_CITY ASSETS

--------------- --------------- ----------

Brighton Brooklyn 7100000

Downtown Brooklyn 9000000

Mianus Horseneck 400000

North Town Rye 3700000

Perryridge Horseneck 1700000

Pownal Bennington 300000

Redwood Palo Alto 3000000

Round Hill Horseneck 800000

8 rows selected.

//Creation table customer

SQL> create table customer(customer\_name varchar(15), customer\_street varchar(20), customer\_city varchar(15));

Table created.

SQL> insert into customer values('Adams', 'Spring', 'Pittsfield');

1 row created.

SQL> insert into customer values('Brooks', 'Senator', 'Brooklyn');

1 row created.

SQL> insert into customer values('Curry', 'North', 'Rye');

1 row created.

SQL> insert into customer values('Glenn', 'Sand Hill', 'Woodside');

1 row created.

SQL> insert into customer values('Green', 'Walnut', 'Stamford');

1 row created.

SQL> insert into customer values('Hayes', 'Main', 'Harrison');

1 row created.

SQL> insert into customer values('Johnson', 'Alma', 'Palo Alto');

1 row created.

SQL> insert into customer values('Jones', 'Main', 'Harrison');

1 row created.

SQL> insert into customer values('Lindsay', 'Park', 'Pittsfield');

1 row created.

SQL> insert into customer values('Smith', 'North', 'Rye');

1 row created.

SQL> insert into customer values('Turner', 'Putnam', 'Stamford');

1 row created.

SQL> insert into customer values('Williams', 'Nassau', 'Princeton');

1 row created.

SQL> select \* from customer;

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY

--------------- -------------------- ---------------

Adams Spring Pittsield

Brooks Senator Brooklyn

Curry North Rye

Glenn Sand Hill Woodside

Green Walnut Stamford

Hayes Main Harrison

Johnson Alma Palo Alto

Jones Main Harrison

Lindsay Park Pittsfield

Smith North Rye

Turner Putnam Stamford

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY

--------------- -------------------- ---------------

Williams Nassau Princeton

12 rows selected.

SQL> update customer set customer\_street='Park' where customer\_name='Lindsay';

1 row updated.

SQL> select \* from customer;

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY

--------------- -------------------- ---------------

Adams Spring Pittsield

Brooks Senator Brooklyn

Curry North Rye

Glenn Sand Hill Woodside

Green Walnut Stamford

Hayes Main Harrison

Johnson Alma Palo Alto

Jones Main Harrison

Lindsay Park Pittsfield

Smith North Rye

Turner Putnam Stamford

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY

--------------- -------------------- ---------------

Williams Nassau Princeton

12 rows selected.

//Creation of Table loan

SQL> create table loan(loan\_number varchar(15), branch\_name varchar(15), amount integer, foreign key(branch\_name) references branch);

Table created.

SQL> insert into loan values('L-11', 'Round Hill', 900);

1 row created.

SQL> insert into loan values('L-14', 'Downtown', 1500);

1 row created.

SQL> insert into loan values('L-15', 'Perryridge', 1500);

1 row created.

SQL> insert into loan values('L-16', 'Perryridge', 1300);

1 row created.

SQL> insert into loan values('L-17', 'Downtown', 1000);

1 row created.

SQL> insert into loan values('L-23', 'Redwood', 1200);

1 row created.

SQL> insert into loan values('L-93', 'Mianus', 500);

1 row created.

///Creation of table Borrower

SQL> create table borrower(customer\_name varchar(15), loan\_number varchar(15), foreign key(customer\_name) references customer, foreign key(loan\_number) references loan);

create table borrower(customer\_name varchar(15), loan\_number varchar(5), foreign key(customer\_name) references customer, foreign key(loan\_number) references loan)

\*

ERROR at line 1:

ORA-02268: referenced table does not have a primary key

SQL> alter table customer add primary key(customer\_name);

Table altered.

SQL> create table borrower(customer\_name varchar(15), loan\_number varchar(15), foreign key(customer\_name) references customer, foreign key(loan\_number) references loan);

Table created.

SQL> insert into borrower values('Adams', 'L-16');

1 row created.

SQL> insert into borrower values('Curry', 'L-93');

1 row created.

SQL> insert into borrower values('Hayes', 'L-15');

1 row created.

SQL> insert into borrower values('Jackson', 'L-14');

insert into borrower values('Jackson', 'L-14')

\*

ERROR at line 1:

ORA-02291: integrity constraint (SYS.SYS\_C0011564) violated - parent key not

found

SQL> insert into borrower values('Johnson', 'L-14');

1 row created.

SQL> insert into borrower values('Smith', 'L-11');

1 row created.

SQL> insert into borrower values('Smith', 'L-23');

1 row created.

SQL> insert into borrower values('Williams', 'L-17');

1 row created.

SQL> insert into borrower values('Jones', 'L-17');

1 row created.

SQL> create table account(account\_number varchar(20), branch\_name varchar(15) , balance integer, primary key(branch\_name), foreign key(branch\_name) references branch);

Table created.

SQL> alter table account drop primary key(branch\_name);

alter table account drop primary key(branch\_name)

\*

ERROR at line 1:

ORA-01735: invalid ALTER TABLE option

SQL> drop table account;

Table dropped.

SQL> create table account(account\_number varchar(20), branch\_name varchar(15) , balance integer, foreign key(branch\_name) references branch);

Table created.

SQL> insert into account values('A-101', 'Downtown', 500);

1 row created.

SQL> insert into account values('A-102', 'Perryridge', 400);

1 row created.

SQL> insert into account values('A-201', 'Brighton', 900);

1 row created.

SQL> insert into account values('A-215', 'Mianus', 700);

1 row created.

SQL> insert into account values('A-222', 'Redwood', 700);

1 row created.

SQL> insert into account values('A-305', 'Round Hill', 350);

1 row created.

SQL> insert into account values('A-217', 'Brighton', 750);

1 row created.

SQL> create table depositor(customer\_name varchar(15), account\_number varchar(6), foreign key(customer\_name) references customer);

Table created.

SQL> insert into depositor values('Hayes', 'A-102');

1 row created.

SQL> insert into depositor values('Johnson', 'A-101');

1 row created.

SQL> insert into depositor values('Johnson', 'A-201');

1 row created.

SQL> insert into depositor values('Jones', 'A-217');

1 row created.

SQL> insert into depositor values('Lindsay', 'A-222');

1 row created.

SQL> insert into depositor values('Smith', 'A-215');

1 row created.

SQL> insert into depositor values('Turner', 'A-305');

1 row created.

SQL> select \* from branch;

BRANCH\_NAME BRANCH\_CITY ASSETS

--------------- --------------- ----------

Brighton Brooklyn 7100000

Downtown Brooklyn 9000000

Mianus Horseneck 400000

North Town Rye 3700000

Perryridge Horseneck 1700000

Pownal Bennington 300000

Redwood Palo Alto 3000000

Round Hill Horseneck 800000

8 rows selected.

SQL> select \* from customer;

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY

--------------- -------------------- ---------------

Adams Spring Pittsield

Brooks Senator Brooklyn

Curry North Rye

Glenn Sand Hill Woodside

Green Walnut Stamford

Hayes Main Harrison

Johnson Alma Palo Alto

Jones Main Harrison

Lindsay Park Pittsfield

Smith North Rye

Turner Putnam Stamford

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY

--------------- -------------------- ---------------

Williams Nassau Princeton

12 rows selected.

SQL> select \* from loan;

LOAN\_NUMBER BRANCH\_NAME AMOUNT

--------------- --------------- ----------

L-11 Round Hill 900

L-14 Downtown 1500

L-15 Perryridge 1500

L-16 Perryridge 1300

L-17 Downtown 1000

L-23 Redwood 1200

L-93 Mianus 500

7 rows selected.

SQL> select \* from borrower;

CUSTOMER\_NAME LOAN\_NUMBER

--------------- ---------------

Adams L-16

Curry L-93

Hayes L-15

Johnson L-14

Smith L-11

Smith L-23

Williams L-17

Jones L-17

8 rows selected.

SQL> select \* from account;

ACCOUNT\_NUMBER BRANCH\_NAME BALANCE

-------------------- --------------- ----------

A-101 Downtown 500

A-102 Perryridge 400

A-201 Brighton 900

A-215 Mianus 700

A-222 Redwood 700

A-305 Round Hill 350

A-217 Brighton 750

7 rows selected.

SQL> select \* from depositor;

CUSTOMER\_NAME ACCOUNT\_NUMBER

--------------- ---------------

Hayes A-102

Johnson A-101

Johnson A-201

Jones A-217

Lindsay A-222

Smith A-215

Turner A-305

7 rows selected.

**QUERIES**

**Q1. Find the number of ‘loan\_numbers’ taken and ‘account\_numbers’ made in every branch.**

SQL> select count(distinct account\_number), count(distinct loan\_number), branch\_name

from account natural full outer join loan

group by branch\_name;

COUNT(DISTINCTACCOUNT\_NUMBER) COUNT(DISTINCTLOAN\_NUMBER) BRANCH\_NAME

----------------------------- -------------------------- ---------------

2 0 Brighton

1 2 Downtown

1 1 Mianus

1 2 Perryridge

1 1 Redwood

1 1 Round Hill

6 rows selected.

**Q2 If the rate of SI is 15% per annum, what is the amount that each borrower has to repay after one year.**

SQL> select customer\_name, loan\_number, amount + amount\*.15 as amt

from borrower natural join loan;

CUSTOMER\_NAME LOAN\_NUMBER AMT

--------------- --------------- ----------

Smith L-11 1035

Johnson L-14 1725

Hayes L-15 1725

Adams L-16 1495

Jones L-17 1150

Williams L-17 1150

Smith L-23 1380

Curry L-93 575

8 rows selected.

**Q35 Increase all accounts with balances over $600 by 6%, all other accounts receive 5%.**

SQL> update account set balance = balance ∗ 1.06

2 where balance > 600;

4 rows updated.

SQL> update account set balance = balance ∗ 1.05

2 where balance <= 600;

**ACCOUNT\_NUMBER BALANCE**

**-------------------- ----------**

**A-101 525**

**A-102 420**

**A-201 954**

**A-215 742**

**A-222 742**

**A-305 368**

**A-217 795**

**7 rows selected.**

**//Q3. Give customer details of all depositors**

SQL> select customer.\*

from customer, depositor

where customer.customer\_name=depositor.customer\_name;

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY

--------------- -------------------- ---------------

Hayes Main Harrison

Johnson Alma Palo Alto

Johnson Alma Palo Alto

Jones Main Harrison

Lindsay Park Pittsfield

Smith North Rye

Turner Putnam Stamford

7 rows selected.

**Q4. Select all depositors living North Street**

SQL> select \*

from customer natural join depositor

where customer\_street='North';

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY ACCOUNT\_NUMBER

--------------- -------------------- --------------- ---------------

Smith North Rye A-215

**Q5 What is the maximum account balance?**

SQL> select max(balance)

from account

where balance > some (select balance from account);

MAX(BALANCE)

------------

954

**Q6. What is the maximum amount of loan given?**

SQL> select max(amount)

2 from loan

3 where amount > some(select amount from loan);

MAX(AMOUNT)

-----------

1500

**Q.7 Give the customer details of all borrowers**

SQL> select distinct customer.\*

2 from customer, borrower

3 where customer.customer\_name=borrower.customer\_name;

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY

--------------- -------------------- ---------------

Adams Spring Pittsield

Hayes Main Harrison

Williams Nassau Princeton

Curry North Rye

Jones Main Harrison

Smith North Rye

Johnson Alma Palo Alto

7 rows selected.

**Q.8 Accounts (account-numbers) made in Horseneck or Brooklyn**

SQL> select account\_number, branch\_city

2 from account natural join branch

3 where branch\_city ='Brooklyn' or branch\_city='Horseneck' order by branch\_city;

ACCOUNT\_NUMBER BRANCH\_CITY

-------------------- ---------------

A-101 Brooklyn

A-201 Brooklyn

A-217 Brooklyn

A-215 Horseneck

A-305 Horseneck

A-102 Horseneck

6 rows selected.

**Q9. Find all details of customers who have a loan and an account**

SQL> select \*

2 from (customer natural join borrower) join depositor using (customer\_name);

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY LOAN\_NUMBER

--------------- -------------------- --------------- ---------------

ACCOUNT\_NUMBER

---------------

Hayes Main Harrison L-15

A-102

Johnson Alma Palo Alto L-14

A-201

Johnson Alma Palo Alto L-14

A-101

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY LOAN\_NUMBER

--------------- -------------------- --------------- ---------------

ACCOUNT\_NUMBER

---------------

Jones Main Harrison L-17

A-217

Smith North Rye L-11

A-215

Smith North Rye L-23

A-215

6 rows selected.

**Q10 Find all details of Smith**

SQL> select \*

2 from (customer natural join borrower) join depositor using (customer\_name)

3 where customer\_name='Smith';

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY LOAN\_NUMBER

--------------- -------------------- --------------- ---------------

ACCOUNT\_NUMBER

---------------

Smith North Rye L-11

A-215

Smith North Rye L-23

A-215

**Q.11 Find the loan numbers and the account numbers of all the customers who live in Harrison.**

SQL> select loan\_number,w

CUSTOMER\_NAME CUSTOMER\_STREET CUSTOMER\_CITY LOAN\_NUMBER

--------------- -------------------- --------------- ---------------

ACCOUNT\_NUMBER

---------------

Smith North Rye L-11

A-215

Smith North Rye L-23

A-215

CUSTOMER\_N LOAN\_ ACCOUN CUSTOMER\_S CUSTOMER\_CITY

---------- ----- ------ ---------- ---------------

Hayes L-15 A-102 Main Harrison

Jones L-17 A-217 Main Harrison

**Q12 Find the names of all the customers who have taken loan from Perryrigde, Mianus or Redwood branch**

SQL> select b.customer\_name

from borrower b, loan l

where b.loan\_number=l.loan\_number and branch\_name in('Perryrigde', 'Mianus' , 'Redwood');

CUSTOMER\_NAME

---------------

Adams

Curry

Hayes

Smith

**Q13 List the names of all the customers in alphabetical order who have a balance of more than Rs. 500 in their accounts**

SQL> select customer\_name

from depositor, account

where depositor.account\_number = account.account\_number and balance > 500 order by customer\_name;

CUSTOMER\_NAME

---------------

Johnson

Jones

Lindsay

Smith

**Q14 Calculate the number of customers city wise**

SQL> select customer\_city, count(customer\_name)

from customer

group by customer\_city;

CUSTOMER\_CITY COUNT(CUSTOMER\_NAME)

--------------- --------------------

Woodside 1

Pittsfield 1

Princeton 1

Pittsield 1

Brooklyn 1

Harrison 2

Stamford 2

Palo Alto 1

Rye 2

9 rows selected.

**Q15 Double the amount of all the customers of Downtown or Mianus branch**

SQL> select branch\_name, loan\_number, amount\*2

from loan

where branch\_name='Downtown' or branch\_name='Mianus';

BRANCH\_NAME LOAN\_NUMBER AMOUNT\*2

--------------- --------------- ----------

Downtown L-14 3000

Downtown L-17 2000

Mianus L-93 1000

**Q16 List the names of all branches ending with n**

SQL> select branch\_name

from branch

where branch\_name like '%n';

BRANCH\_NAME

---------------

Brighton

Downtown

North Town

**Q17 Find the sum of all the assets branch wise**

SQL> select branch\_city, sum(assets)

from branch

group by branch\_city;

BRANCH\_CITY SUM(ASSETS)

--------------- -----------

Horseneck 2900000

Brooklyn 16100000

Palo Alto 3000000

Bennington 300000

Rye 3700000

**Q18 Find the average balance of each customer who lives in Perryridge and has at least 2 accounts.**

SELECT depositor.customer\_name,avg(balance)

FROM depositor,account,customer

WHERE depositor.account\_number=account.account\_number AND depositor.customer\_name=customer.customer\_name AND customer\_city='Perryridge'

GROUP BY Depositor.customer\_name

HAVING COUNT ( DISTINCT Depositor.account\_number) >= 2;

no rows selected

**Q20 Find the branch that has the higher average balance**

SQL>SELECT branch\_name

FROM account

GROUP BY branch\_name

HAVING avg(balance) >= ALL (SELECT avg(balance)

FROM account

GROUP BY branch\_name);

BRANCH\_NAME

---------------

Brighton

**Q28 Find the average loan amount for each branch.**

SQL> SELECT branch\_name, AVG(amount) AS avg\_amt

FROM loan

GROUP BY branch\_name;

BRANCH\_NAME AVG\_AMT

--------------- ----------

Round Hill 900

Mianus 500

Perryridge 1400

Redwood 1200

Downtown 1250

**Q33 Delete the record of all accounts with balances below the average at the bank.**

delete from account

where balance < (select avg (balance)

from account)

**Q19 Find the names of all branches that have assets greater than that of all branch in Brooklyn .**

SELECT branch\_name

FROM branch

WHERE assets > ALL (SELECT assets

FROM branch

WHERE branch\_name='Brookylyn');

BRANCH\_NAME

---------------

Pownal

Mianus

Round Hill

Perryridge

Redwood

North Town

Brighton

Downtown

8 rows selected.

**Q21** **Find all loan number for loans made at the Perryridge branch with loan amounts greater than $1200.**

SQL> SELECT loan\_number

FROM Loan

WHERE branch\_name='Perryridge' AND amount > 1200;

LOAN\_NUMBER

---------------

L-15

L-16

**Q22 Find the number of branches that currently have loans.**

SQL> SELECT COUNT (DISTINCT branch\_name)

FROM loan;

COUNT(DISTINCTBRANCH\_NAME)

--------------------------

5

**Q23 Find the ID of the largest loan at each branch, including the branch name and the amount of the loan.**

SQL> SELECT \*

FROM loan

WHERE (branch\_name, amount)

IN (SELECT branch\_name,MAX(amount )

FROM loan

GROUP BY branch\_name);

LOAN\_NUMBER BRANCH\_NAME AMOUNT

--------------- --------------- ----------

L-11 Round Hill 900

L-93 Mianus 500

L-15 Perryridge 1500

L-23 Redwood 1200

L-14 Downtown 1500

**Q24 Find customers with an account but not a loan.**

SQL> SELECT DISTINCT customer\_name

FROM depositor d

WHERE NOT EXISTS (SELECT \*

FROM borrower b

WHERE b.customer\_name = d.customer\_name);

CUSTOMER\_NAME

---------------

Turner

Lindsay

**Q25 Find all branches with assets greater than at least one branch in Harrison**

SQL> SELECT branch\_name

FROM branch

WHERE assets > SOME ( SELECT assets

FROM branch

WHERE branch\_name='Harrison');

no rows selected

**Q26 Find all cities with more than two customers living in the city.**

SQL> SELECT customer\_city, COUNT(\*) AS num\_customers

FROM customer

GROUP BY customer\_city

HAVING COUNT(\*) > 2;

no rows selected

**Q.29 Find the number of tuples in the customer relation.**

SQL> SELECT COUNT(\*)

FROM Customer;

COUNT(\*)

----------

12

**Q30 Find the names of customers who do have a loan at the bank, and whose names are neither Smith nor Jones.**

SQL> SELECT DISTINCT customer\_name

FROM Borrower

WHERE customer\_name NOT IN ('Smith', 'Jones');

CUSTOMER\_NAME

---------------

Adams

Johnson

Curry

Hayes

Williams

**Q32 Find the names of all customers whose street includes the substring “Main**”.

SQL> select customer\_name

from customer

where customer\_street like '%Main%';

CUSTOMER\_NAME

---------------

Hayes

Jones

**Q34 Find all customers who have either an account or a loan (but not both) at the bank.**

SQL> select customer\_name

from depositor natural full outer join borrower

where account\_number is null or loan\_number is null;

CUSTOMER\_NAME

---------------

Adams

Curry

Williams

Lindsay

Turner

**Q36 A view consisting of branches and their customers**

create view all-customer as (select branch\_name, customer\_name from depositor, account where depositor.account\_number = account.account\_number) union (select branch\_name, customer\_name from borrower, loan where borrower.loan\_number = loan.loan\_number)

**Q37 Find all customers who have both an account and a loan at the Perryridge branch**

SQL> select distinct customer\_name from borrower, loan where borrower.loan\_number = loan.loan\_number and branch\_name ='Perryridge' and (branch\_name, customer\_name) in (select branch\_name, customer\_name from depositor, account where depositor.account\_number = account.account\_number);

CUSTOMER\_N

----------

Hayes

**Q38 Find all branches that have greater assets than some branch located in Brooklyn.**

select distinct T.branch-name from branch as T, branch as S

where T.assets > S.assets and S.branch-city = ‘Brooklyn’

BRANCH\_NAME

---------------

Downtown

**Q39 Find the loan info and account info of all customers whose nme include ‘a’**

select \* from depositor natural full outer join borrower where customer\_name like '%a%';

CUSTOMER\_NAME ACCOUNT\_NUMBER LOAN\_NUMBER

--------------- --------------- ---------------

Adams L-16

Hayes A-102 L-15

Williams L-17

Lindsay A-222

**Q40 Find the streets where all account holder live**

CUSTOMER\_STREET

--------------------

Alma

North

Main

Park

Putnam