

DBMS(KCS 551) Lab Manual

Course Outcome

At the end of course , the student will be able to:

CO Number	Statement	Bloom's Knowledge Level
CO1	Understand and apply oracle 11 g products for creating tables, views, indexes, sequences and other database objects.	K2, K4
CO2	Design and implement a database schema for company data base, banking data base, library information system, payroll processing system, student information system.	K3, K5, K6
CO3	Write and execute simple and complex queries using DDL, DML, DCL and TCL	K4, K5
CO4	Write and execute PL/SQL blocks, procedure functions, packages and triggers, cursors.	K4, K5
CO5	Enforce entity integrity, referential integrity, key constraints, and domain constraints on database.	K3, K4

PART-A: SQL Programming • Design, develop, and implement the specified queries for the following problems using Oracle, MySQL, MS SQL Server, or any other DBMS under LINUX/Windows environment. • Create Schema and insert at least 5 records for each table. Add appropriate database constraints.

PART-B: Mini Project • Use Java, PHP, Python, or any other similar front-end tool. • All applications must be demonstrated on desktop/laptop as a stand-alone or web based application (Mobile apps on Android/IOS are not permitted.)

PART A:

SQL PROGRAMMING

➤ Installation of MySQL

➤ Important commands to work with MySQL

1. **show databases;**

It displays all the databases present in mysql.

2. **create database** database name;

It creates a database with the mentioned name.

3. **use** database name;

Database will be changed to the mentioned database.

4. **show tables;**

It displays all the tables in the current database.

5. **Table Creation**

```
mysql> CREATE TABLE employee (  
    id int NOT NULL AUTO_INCREMENT,  
    name varchar(45) NOT NULL,  
    occupation varchar(35) NOT NULL,  
    age int NOT NULL,  
    PRIMARY KEY (id)  
);
```

6. **describe table** table name;

It displays all the fields with their type of the mentioned table.

7. **Row Insertion into Table**

(a) If we want to store single records for all fields, use the syntax as follows:

```
INSERT INTO employee (id, name, occupation, age)  
VALUES (101, 'Peter', 'Engineer', 32);
```

(b) If we want to store multiple records, use the following statements where we can either specify all field names or don't specify any field.

```
INSERT INTO employee VALUES
```

```
(102, 'Joseph', 'Developer', 30),  
(103, 'Mike', 'Leader', 28),  
(104, 'Stephen', 'Scientist', 45);
```

- (c) If we want to store records without giving all fields, we use the following **partial field** statements. In such case, it is mandatory to specify field names.

```
INSERT INTO employee (name, occupation)  
VALUES ('Stephen', 'Scientist'), ('Bob', 'Actor');
```

8. Display the table

```
mysql> SELECT * FROM EMPLOYEE;
```

9. MySQL Date Data Types

MySQL comes with the following data types for storing a date or a date/time value in the database:

- **DATE** - format YYYY-MM-DD
- **DATETIME** - format: YYYY-MM-DD HH:MI:SS
- **TIMESTAMP** - format: YYYY-MM-DD HH:MI:SS
- **YEAR** - format YYYY or YY

The **default format** of the date in MySQL is **YYYY-MM-DD**.

10. To display only 5 records of the table

```
mysql> select * from employee order by id limit 5;
```

Assignment-1

Create the following tables:-

Programmer (Name, DOB, DOJ, Sex, Prof1, Prof2, Salary)

Studies (Name, SPLACE, Course, CCost)

Software (Name, Title, Dev_in, SCost, DCost, Sold)

Insert the following values into tables:-

Programmer Table

PNAME	DOB	DOJ	SEX	PROF1	PROF2	SALARY
ANAND	12-APR-66	21-APR-92	M	PASCAL	BASIC	3200
ALTAF	02-JUL-64	13-NOV-90	M	CLIPPER	COBOL	2800
JULIANA	31-JAN-60	21-APR-90	F	COBOL	DBASE	3000
KAMLA	30-OCT-68	02-JAN-92	F	C	DBASE	2900
MARY	24-JUN-70	01-FEB-91	F	CPP	ORACLE	4500
NELSON	11-SEP-85	11-MAR-89	M	COBOL	DBASE	2500
PATTRICK	10-NOV-65	21-APR-90	M	PASCAL	COBOL	2800
QADIR	31-AUG-65	21-APR-90	M	ASSEMBLY	C	3000
RAMESH	03-MAY-67	26-FEB-91	M	PASCAL	DBASE	3200
REBECCA	01-JAN-67	01-DEC-90	F	BASIC	COBOL	2500
REMITHA	19-APR-70	20-APR-93	F	C	ASSEMBLY	3000
REVATHI	02-DEC-69	02-JAN-92	F	PASCAL	BASIC	3200
VIJAYA	14-DEC-65	02-MAY-92	F	FOXPRO	C	4500

Software Table

PNAME	TITLE	DEV_IN	SCOST	DCOST	SOLD
MARY	README	CPP	100.00	1200	84
ANAND	PARACHUTES	BASIC	400	6000	43
ANAND	VIDEOTITLING	PASCAL	7500	16000	9
JULIANA	INVENTORY	COBOL	3000	3500	0
KAMLA	PAYROLLPRG	DBASE	9000	20000	7
MARY	FINANCIALACC	ORACLE	18000	85000	4
MARY	CODEGENRRATOP	C	4500	20000	23
PATTRICK	README	CPP	300	1200	84
QADIR	BOMBSAWAY	ASSEMBLY	750	5000	11
QADIR	VACCINES	C	1900	3400	21
RAMESH	HOTLIMGMT	DBASE	12000	35000	4
RAMESH	DEADLEE	PASCAL	600	4500	73
REMITHA	PCUTILITIES	C	725	5000	51
REMITHA	TSRHELPPKG	ASSEMBLY	2500	6000	7

REVATHI	HOSPITALMGMT	PASCAL	1100.00	75000	2
VIJAYA	TSRECTOR	C	900.00	700	6

Studies Table

PNAME	SPLACE	COURSE	CCOST
ANAND	SABHARI	PGDCA	4500
ALTAF	COIT	DCA	7200
JULIANA	BITS	MCA	22000
KAMLA	PRAGATHI	DCP	5000
MARY	SABHARI	PGDCA	4500
NELSON	PRAGATHI	DAP	6200
PATRICK	PRAGATHI	DCAP	5200
QADIR	APPLE	HDCA	14000
RAMESH	SABHARI	PGDCA	4500
RESECCA	BRILLIANT	DCAP	11000
REMITHA	BDPS	DCS	6000
VIJAYA	BDPS	DCA	48000

QUERIES

1) Find out the SELLING COST AVERAGE for the packages developed in PASCAL?

Select avg(scost) from software where dev_in="Pascal";

2) Display the names and ages of all programmers.

Select pname, date_format(from_days(datediff(curdate(),dob)),'%y') as Age from programmer;

Or

Select pname, timestampdiff(year,dob,curdate()) as Age from programmer;

3) Display the course cost of PGDCA course.

Select distinct(ccost) from studies where course="PGDCA";

4) What is the highest numbers of copies sold by a package?

Select max(sold) from software;

5) Display the names and date of birth of all the programmer born in JANUARY.

Select pname,dob from programmer where month(dob)=01;

6) Display lowest course fee.

Select min(ccost) from studies;

7) How many programmer has done PGDCA course.

Select count(pname) from studies where course="PGDCA";

8) How much revenue has been earned through sales of packages in C.

Select sum(scost*sold-dcost) from software where dev_in="C";

9) Display the details of software developed by Ramesh?

Select * from software where pname="Ramesh";

10) How many programmers studied at SABHARI?

Select count(pname) from studies where splace="SABHARI";

11) Display the details of PACKAGES whose sales crossed the 20000 mark.

Select title from software where scost*sold>20000;

12) Find out the number of copies which should be sold in order to recover the development cost of each package.

Select title, ceiling(dcost/scost) as num_of_copies from software;

13) What is the price of the costliest software developed in BASIC?

Select max(scost) from software where dev_in="BASIC";

14) Display the details of packages for which development cost has been recovered.

Select * from software where scost>dcost;

15) How many packages were developed in dbase?

Select count(title) from software where dev_in="dbase";

16) How many programmers studies at Pragathi?

Select count(pname) from studies where splace="Pragathi";

17) How many programmers paid 5000 to 10000 for their course?

Select count(pname) from studies where ccost>=5000 and ccost<=10000;

18) What is the average course fee?

Select avg(ccost) from studies;

19) Display the details of programmers knowing C?

Select * from programmer where prof1="C" or prof2="C";

20) How many programmers know either COBOL or Pascal?

Select count(pname) from programmer where (prof1="COBOL" or prof1="Pascal") or (prof2="COBOL" or prof2="Pascal");

21) How many programmers don't know Pascal & C?

Select count(pname) from programmer where prof1 not in ("Pascal","C") and prof2 not in ("Pascal","C");

22) How old is the oldest male programmers?

```
Select max(timestampdiff(year,dob,curdate())) from programmer where sex="m";
```

23) What is the average age of female programmers?

```
Select avg(timestampdiff(year,dob,curdate())) from programmer where sex="f";
```

24) Calculate the experience in years for each programmers and display along with the names in descending order?

```
Select pname, timestampdiff(year,doj,curdate()) from programmer order by pname desc;
```

25) Who are the programmers who celebrate their birthday during the current month?

```
Select pname from programmer where month(dob)=month(curdate());
```

26) How many female programmers are there?

```
Select count(pname) from programmer where sex="f";
```

27) What are the languages known by the male programmers?

```
Select prof1,prof2 from programmer where sex="m";
```

28) What is the Average salary?

```
Select avg(salary) from programmer;
```

29) How many people draw 2000 to 4000?

```
Select count(pname) from programmer where salary>=2000 and salary<=4000;
```

30) Display the details of those who don't know Clipper, COBOL or Pascal?

Select * from programmer where prof1 not in ("Clipper","cobol","Pascal") and prof2 not in ("Clipper","cobol","Pascal");

31) How many Female programmers knowing C are above 24 years of age?

Select count(pname) from programmer where (prof1="C" or prof2="C") and sex="f" and timestampdiff(year,dob,now())>24;

32) Who are the programmers who will be celebrating their Birthday within a week?

Select pname from programmer where week(dob) = week(now());

33 Display the details of those with less than a year's experience?

Select * from programmer where timestampdiff(day,doj,now())<365;

34 Display the details of those who will be completing 2 years of service this year?

Select * from programmer where timestampdiff(year,doj,"2022-12-31")=2;

35 Calculate the amount to be recovered for those packages whose development cost has not been recovered?

Select title, scost*sold from software where scost*sold<dcost;

36) List the packages which have not been sold so far?

Select title from software where sold=0;

37) Find out the cost of the software developed by Mary?

Select dcost from software where pname="Mary";

38) Display the institute's names from the studies table without duplicates?

Select distinct(splace) from studies;

39) How many different courses are mentioned in the studies table?

Select course(distinct(course)) from studies;

40) Display the names of the programmers whose names contain 2 occurrences of the letter A?

Select pname from programmer where pname like "%A%A%";

41) Display the names of programmers whose names contain upto 5 characters.

Select pname from programmer where length(pname)<=5;

42) How many female programmers knowing COBOL have more than 2 years experience?

Select count(pname) from programmer where (prof1="cobol" or prof2="cobol") and sex="f" and timestampdiff(year,doj,now())>2;

43) What is the length of the shortest name in the programmer table?

Select min(length(pname)) from programmer;

44) What is the average development cost of a package developed in COBOL?

Select avg(dcost) from software where dev_in="cobol";

45) Display the name, sex, dob(DD/MM/YY format), doj(DD/MM/YYYY format), for all the programmers .

Select pname, sex,date_format(dob,'%d/%m/%y') as dob,date_format(doj,'%d/%m/%Y') as doj from programmer;

46) Who are the programmers who were born on the last day of the month?

Select pname from programmer where last_day(dob)=dob;

47) What is the amount paid in salaries of the male programmers who do not know Cobol?

Select sum(salary) from programmer where sex="m" and prof1<>"cobol" and prof2<>"cobol";

48) Display the title, scost, dcost and difference between scost and dcost in descending order of difference?

Select title,scost,dcost, dcost-scost as diff from software order by diff desc;

49) Display the name, dob, doj of those month of birth and month of joining are same?

Select pname,dob,doj from programmer where month(dob)=month(doj);

50) Display the names of the packages whose names contain more than 1 word?

Select title from software where title like "% %";

Solution of QUERIES:

- (1) Select avg(scost) from software where dev_in="PASCAL";
- (2) Select pname, date_format(from_days(datediff(curdate(),dob)),'%y') as Age from programmer;
or
Select pname, timestampdiff(year,dob,curdate()) as Age from programmer;
- (3) Select distinct(ccost) from studies where course="PGDCA";
- (4) Select max(sold) from software;
- (5) Select pname,dob from programmer where month(dob)=01;
- (6) Select min(ccost) from studies;
- (7) Select count(pname) from studies where course="PGDCA";
- (8) Select sum(scost*sold-dcost) from software where dev_in="C";
- (9) Select * from software where pname="Ramesh";
- (10) Select count(pname) from studies where splace="SABHARI";
- (11) Select title from software where scost*sold>20000;
- (12) Select title, ceiling(dcost/scost) as num_of_copies from software;

- (13) Select max(scost) from software where dev_in="BASIC";
- (14) Select * from software where scost*sold>dcost;
- (15) Select count(title) from software where dev_in="dbase";
- (16) Select count(pname) from studies where splace="Pragathi";
- (17) Select count(pname) from studies where ccost>=5000 and ccost<=10000;
- (18) Select avg(ccost) from studies;
- (19) Select * from programmer where prof1="C" or prof2="C";
- (20) Select count(pname) from programmer where (prof1="COBOL" or prof1="Pascal")
or (prof2="COBOL" or prof2="Pascal");
- (21) Select count(pname) from programmer where prof1 not in ("Pascal","C") and
prof2 not in ("Pascal","C");
- (22) Select max(timestampdiff(year,dob,curdate())) from programmer where sex="m";
- (23) Select avg(timestampdiff(year,dob,curdate())) from programmer where sex="f";
- (24) Select pname, timestampdiff(year,doj,curdate()) from programmer order by
pname desc;
- (25) Select pname from programmer where month(dob)=month(curdate());
- (26) Select count(pname) from programmer where sex="f";
- (27) Select prof1,prof2 from programmer where sex="m";
- (28) Select avg(salary) from programmer;
- (29) Select count(pname) from programmer where salary>=2000 and salary<=4000;
- (30) Select * from programmer where prof1 not in ("Clipper","cobol","Pascal") and
prof2 not in ("Clipper","cobol","Pascal");
- (31) Select count(pname) from programmer where (prof1="C" or prof2="C") and
sex="f" and timestampdiff(year,dob,now())>24;
- (32) Select pname from programmer where week(dob) = week(now());
- (33) Select * from programmer where timestampdiff(day,doj,now())<365;
- (34) Select * from programmer where timestampdiff(year,doj,"2022-12-31")=2;
- (35) Select title, scost*sold from software where scost*sold<dcost;
- (36) Select title from software where sold=0;
- (37) Select dcost from software where pname="Mary";
- (38) Select distinct(splace) from studies;
- (39) Select course(distinct(course)) from studies;
- (40) Select pname from programmer where pname like "%A%A%";
- (41) Select pname from programmer where length(pname)<=5;
- (42) Select count(pname) from programmer where (prof1="cobol" or prof2="cobol")
and sex="f" and timestampdiff(year,doj,now())>2;

- (43) Select min(length(pname)) from programmer;
- (44) Select avg(dcost) from software where dev_in="cobol";
- (45) Select pname, sex,date_format(dob,'%d/%m/%y') as dob,
 date_format(doj,'%d/%m/%Y') as doj from programmer;
- (46) Select pname from programmer where last_day(dob)=dob;
- (47) Select sum(salary) from programmer where sex="m" and prof1<>"cobol" and
 prof2<>"cobol";
- (48) Select title,scost,dcost, dcost-scost as diff from software order by diff desc;
- (49) Select pname,dob,doj from programmer where month(dob)=month(doj);
- (50) Select title from software where title like "% %";

Experiment-2

Consider the following relation schemas of a database:-

Sailors(sid: integer, sname: string, rating: integer, age: real)

Boats(bid: integer, bname: string, color: string)

Reserves(sid: integer, bid: integer, day: date)

Consider the following table as instance:-

Sailors

<i>sid</i>	<i>sname</i>	<i>rating</i>	<i>age</i>
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

Reserves

<i>sid</i>	<i>bid</i>	<i>day</i>
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

<i>bid</i>	<i>bname</i>	<i>color</i>
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Write the following queries in SQL.

1. Find the names of sailors who have reserved boat number 103.
2. Find all sailors with a rating above 7.
3. Find the sids of sailors who have reserved a red boat.
4. Find the names of sailors who have reserved a red boat.
5. Find the colors of boats reserved by Lubber.
6. Find the names of sailors who have reserved at least one boat.
7. Compute increments for the ratings of persons who have sailed two different boats on the same day.
8. Find the ages of sailors whose name begins and ends with B and has at least three characters.
9. Find the names of sailors who have reserved a red or a green boat.
10. Find the names of sailors who have reserved both a red and a green boat.
11. Find the sids of all sailors who have reserved red boats but not green boats.
12. Find the names of sailors who have not reserved a red boat.
13. Find sailors whose rating is better than some sailor called Horatio.
14. Find the sailors with the highest rating.
15. Find the names of sailors who have reserved all boats.
16. Find the average age of all sailors.
17. Find the average age of sailors with a rating of 10.
18. Find the name and age of the oldest sailor.
19. Count the number of different sailor names.
20. Find the names of sailors who are older than the oldest sailor with a rating of 10.
21. Find the age of the youngest sailor for each rating level.
22. Find the age of the youngest sailor who is eligible to vote (i.e., is at least 18 years old) for each rating level with at least two such sailors.
23. For each red boat, find the number of reservations for this boat.
24. Find the average age of sailors for each rating level that has at least two sailors.
25. Find the average age of sailors who are of voting age (i.e., at least 18 years old) for each rating level that has at least two sailors.
26. Find the average age of sailors who are of voting age (i.e., at least 18 years old) for each rating level that has at least two such sailors.

27. Find those ratings for which the average age of sailors is the minimum overall ratings.

Experiment-3

Consider the following employee database:-

Employee (employee-name, street, city)

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

Company (company-name, city)

Manages (employee-name, manager-name)

Where the primary keys are underlined. Give an expression in SQL for each of the following queries.

The tables for the above schemas are the following:-

Employee Table

Employee-name	Street	City
Neha	Naini	Prayagraj
Rakesh	Indira Nagar	Lucknow
Ritika	Kakadev	Kanpur
Ashok	Chandani Chowk	Delhi
Manish	Civil Line	Prayagraj
Shyam	Gomti Nagar	Lucknow
Abhijeet	Harsh Nagar	Kanpur

Work Table

Employee-name	Company-name	Salary
Neha	First Bank Corporation	40000
Rakesh	Small Bank Corporation	30000
Ritika	HDFC Bank	35000
Ashok	Small Bank Corporation	25000
Manish	First Bank Corporation	15000
Shyam	First Bank Corporation	10000
Abhijeet	HDFC Bank	20000

Company Table

Company-name	City
First Bank Corporation	Prayagraj
Small Bank Corporation	Delhi
HDFC Bank	Kanpur
Yes Bank Corporation	Prayagraj
Reliance	Lucknow

Manages Table

Employee-name	Manager-name
Neha	Manish
Rakesh	Ashok
Ritika	Null
Ashok	Null
Manish	Null
Shyam	Manish

Queries:

(1) Find the names of all employees who work for First Bank Corporation.

- (2) Find the names and cities of residence of all employees who work for First Bank Corporation.
- (3) Find the names, street addresses, and cities of residence of all employees who work for First Bank Corporation and earn more than \$10,000.
- (4) Find all employees in the database who live in the same cities as the companies for which they work.
- (5) Find all employees in the database who live in the same cities and on the same streets as do their managers.
- (6) Find all employees in the database who do not work for First Bank Corporation.
- (7) Find all employees in the database who earn more than each employee of Small Bank Corporation.
- (8) Assume that the companies may be located in several cities. Find all companies located in every city in which Small Bank Corporation is located.
- (9) Find all employees who earn more than the average salary of all employees of their company.
- (10) Find the company that has the most employees.
- (11) Find the company that has the smallest payroll.
- (12) Find those companies whose employees earn a higher salary, on average, than the average salary at First Bank Corporation.

Solution of Queries:

- (1) select employee-name from works where company-name="First Bank Corporation";
- (2) select employee-name, city from employee natural join works where company-name="First Bank Corporation";
- (3) select employee-name, street, city from employee natural join works where company-name="First Bank Corporation" and salary>10000;

- (4)select employee-name from employee natural join works natural join company;
- (5)select employee-name from employee natural join manages where (manager-name,street,city) in (select manager-name,street,city from employee,manages where employee.employee-name=manages.manager-name)
- (6)select employee-name from works where company-name <> "First Bank Corporation"
- (7)select employee-name from works where salary >all (select salary from works where company-name="Small Bank Corporation");
- (8)select cname from company as S where not exists (select city from company where cname="Small Bank Corporation" except select city from company where company.cname=S.cname);
or
select company-name from company as S where not exists (select P.city from (select city from company where company-name="Small Bank Corporation") as P LEFT JOIN (select city from company as T where T.company-name=S.company-name) as R on P.city=R.city where R.city is NULL);
- (9)select employee-name from works as S where salary > (select avg(salary) from works as T where T.company-name=S.company-name);
- (10) select company-name from works group by company-name having count(employee-name) >=all (select count(employee-name) from works group by company-name);
- (11) select company-name from works group by company-name having sum(salary) <=all (select sum(salary) from works group by company-name);
- (12) select company-name from works group by company-name having avg(salary) > (select avg(salary) from works where company-name ="First Bank Corporation");

Experiment-4

Consider the following relation schemas:-

Branch-schema = (branch-name, branch-city, assets)

Customer-schema = (customer-name, customer-street, customer-city)

Loan-schema = (loan-number, branch-name, amount)

Borrower-schema = (customer-name, loan-number)

Account-schema = (account-number, branch-name, balance)

Depositor-schema = (customer-name, account-number)

Account table

<i>account-number</i>	<i>branch-name</i>	<i>balance</i>
A-101	Downtown	500
A-102	Perryridge	400
A-201	Brighton	900
A-215	Mianus	700
A-217	Brighton	750
A-222	Redwood	700
A-305	Round Hill	350

Branch table

<i>branch-name</i>	<i>branch-city</i>	<i>assets</i>
Brighton	Brooklyn	7100000
Downtown	Brooklyn	9000000
Mianus	Horseneck	400000
North Town	Rye	3700000
Perryridge	Horseneck	1700000
Pownal	Bennington	300000
Redwood	Palo Alto	2100000
Round Hill	Horseneck	8000000

Customer table

<i>customer-name</i>	<i>customer-street</i>	<i>customer-city</i>
Adams	Spring	Pittsfield
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
Williams	Nassau	Princeton

depositor table

<i>customer-name</i>	<i>account-number</i>
Hayes	A-102
Johnson	A-101
Johnson	A-201
Jones	A-217
Lindsay	A-222
Smith	A-215
Turner	A-305

Loan table**Borrower table**

<i>loan-number</i>	<i>branch-name</i>	<i>amount</i>
L-11	Round Hill	900
L-14	Downtown	1500
L-15	Perryridge	1500
L-16	Perryridge	1300
L-17	Downtown	1000
L-23	Redwood	2000
L-93	Mianus	500

<i>customer-name</i>	<i>loan-number</i>
Adams	L-16
Curry	L-93
Hayes	L-15
Jackson	L-14
Jones	L-17
Smith	L-11
Smith	L-23
Williams	L-17

Queries:

1. Find the customer names, loan numbers, and loan amounts for all loans at the Perryridge branch.
2. Find the names of all customers whose street address includes the substring 'Main'.
3. Find all customers in alphabetic order who have a loan at the Perryridge branch.
4. Find all customers having a loan, an account, or both at the bank.
5. Find all customers who have both a loan and an account at the bank.
6. Find all customers who have an account but no loan at the bank.
7. Find the average account balance at the Perryridge branch. ^
8. Find the average account balance at each branch.
9. Find the number of depositors for each branch.
10. Find the branches where the average account balance is more than \$1200.
11. Find the average balance for each customer who lives in Harrison and has at least three accounts.
12. Find all customers who have both an account and a loan at the Perryridge branch.
13. Find the names of all branches that have assets greater than those of at least one branch located in Brooklyn.
14. Find the names of all branches that have an asset value greater than that of each branch in Brooklyn.
15. Find those branches for which the average balance is greater than or equal to all average balances.
16. Find all customers who have an account at all the branches located in Brooklyn.
17. Find the average account balance of those branches where the average account balance is greater than \$1200.
18. Find the maximum across all branches of the total balance at each branch.