CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY FACULTY OF TECHNOLOGY & ENGINEERING

DEVANG PATEL INSTITUE OF ADVANCE TECHNOLOGY AND RESEARCH (DEPSTAR)

Subject: CE246 -Database Management System

Semester: IV Academic Year: 2021-22

Full name: Vaishnavi Bhalodi

Id: 20DCE008

Practical-1

Aim:

Evaluation of Database (File System, DBMS, RDBMS, DDBMS)

Theory:

File System:

A file system is a process that manages how and where data on a storage disk, typically a hard disk drive (HDD), is stored, accessed and managed. It is a logical disk component that manages a disk's internal operations as it relates to a computer and is abstract to a human user.
Examples: NTFS, HFS

DBMS:

A database management system (DBMS) is system software for creating and managing databases. A DBMS makes it possible for end users to create, protect, read, update and delete data in a database. The most prevalent type of data management platform, the DBMS essentially serves as an interface between databases and end users or application programs, ensuring that data is consistently organized and remains easily accessible.

DDBMS:

This chapter introduces the concept of DDBMS. In a distributed database, there are a number of databases that may be geographically distributed all over the world. A distributed DBMS manages the distributed database in a manner so that it appears as one single database to users.

RDBMS:

The software used to store, manage, query, and retrieve data stored in a relational database is called a relational database management system (RDBMS). The RDBMS provides an interface between users and applications and the database, as well as administrative functions for managing data storage, access, and performance.

Differences:

File System	DBMS
It manages and organizes files in a storage medium.	It manages the database.
It doesn't provide backup.	It provides backup and recovery of data.
It has less consistency.	It has less consistency.
It is less complex as compared to DBMS.	It is more complex as compared to DBMS.
It provides less security.	It provides more security.
It provides less expensive.	It provides more expensive.
There is no data independence.	There is data independence.

DBMS	RDBMS
It stores data as file format.	It stores data in tabular form.
Data elements are accessed individually.	Multiple data elements can be accessed at the same time.
There is no relationship between the data.	There is relationship between the data present in the tabular form.
It is does not support distributed database.	It supports distributed database.
It stores data in either a navigational or hierarchical form.	It uses a tabular structure where the headers are the column names, and the rows contain corresponding values.
It provides less expensive.	It provides more expensive.
It deals with small quantity of data	It deals with large amount of data.
It supports single user.	It supports multiple users.

Conclusion:

I learnt about the different database systems and their differences.

Practical-2

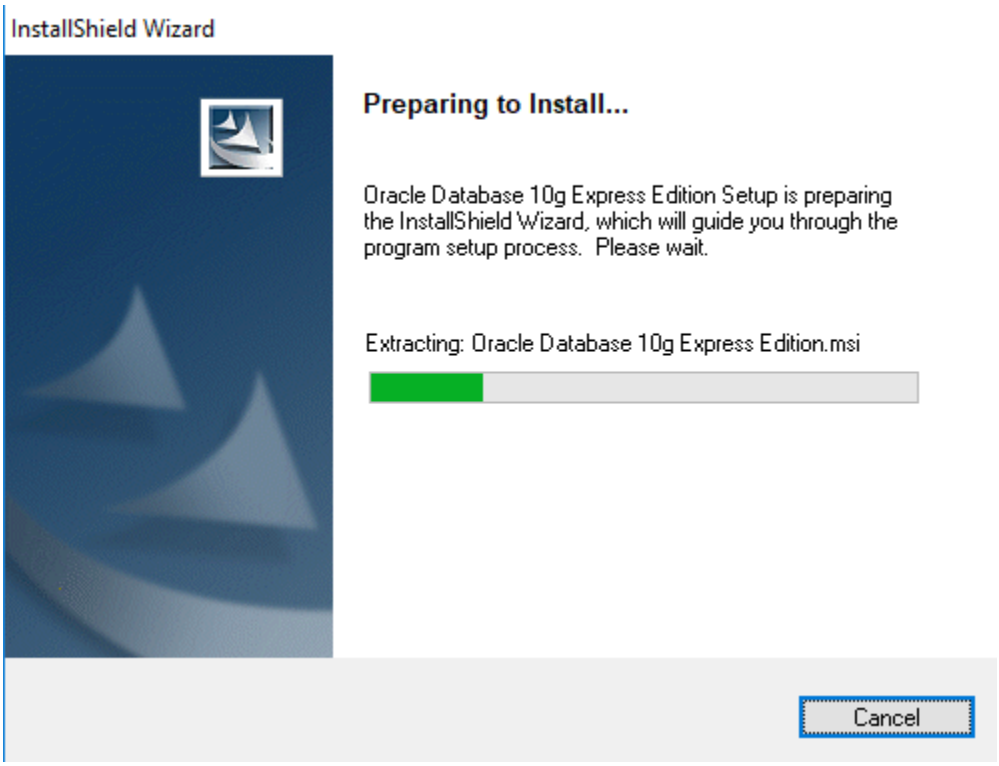
Aim:

Introduction to Oracle (step by step installation, introduction of sql, plsql).

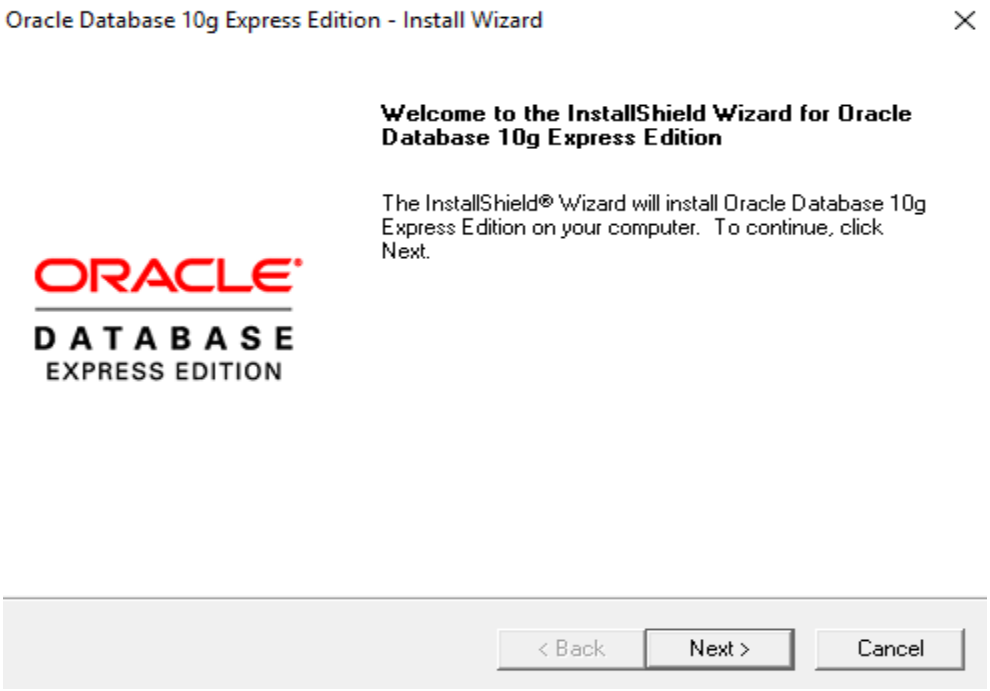
Installation:

1) Download Oracle 10g from below link: https://drive.google.com/file/d/1Y6ghDOEfVorTNrzWgF1UKaENHjeGgrHG/view

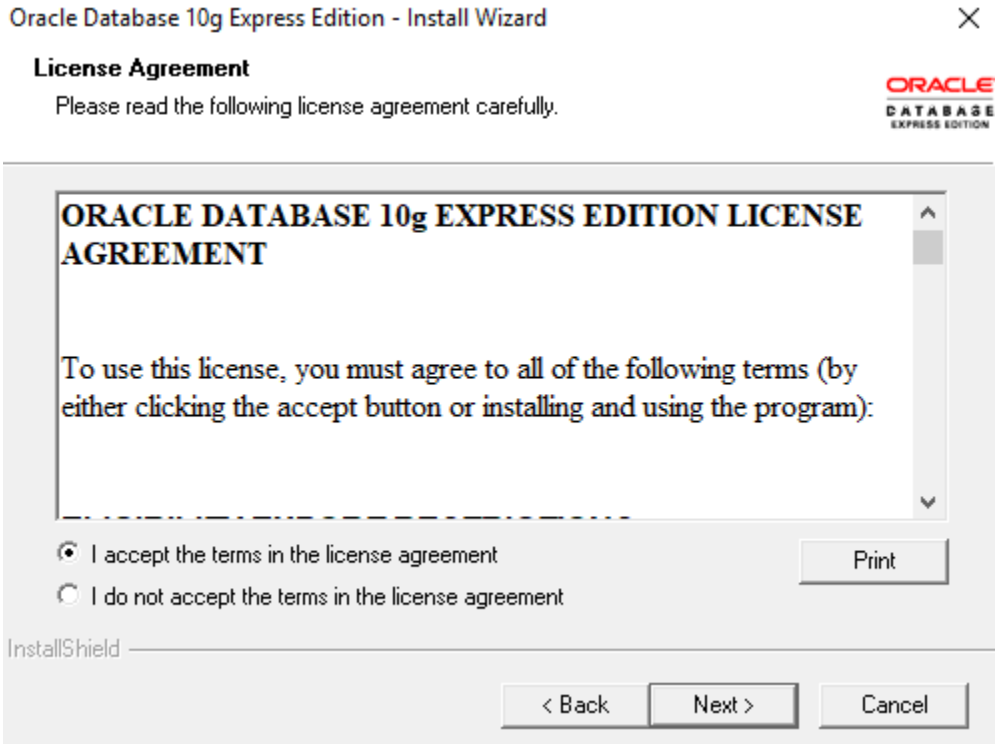
2) Install it by double clicking .exe which you have downloaded



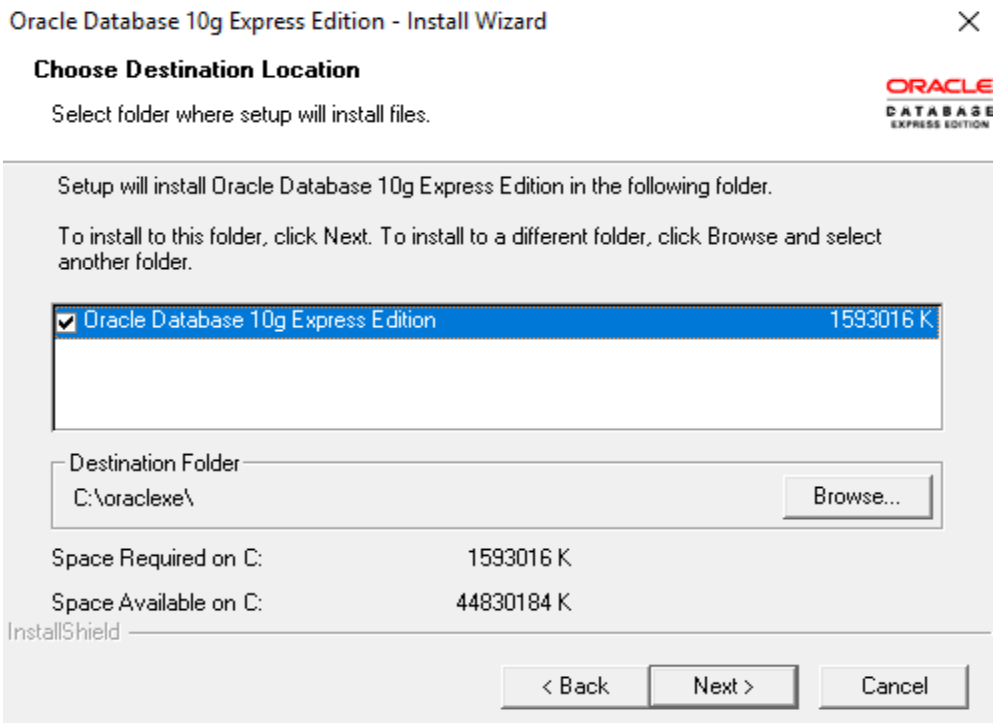
3) Click on Next button



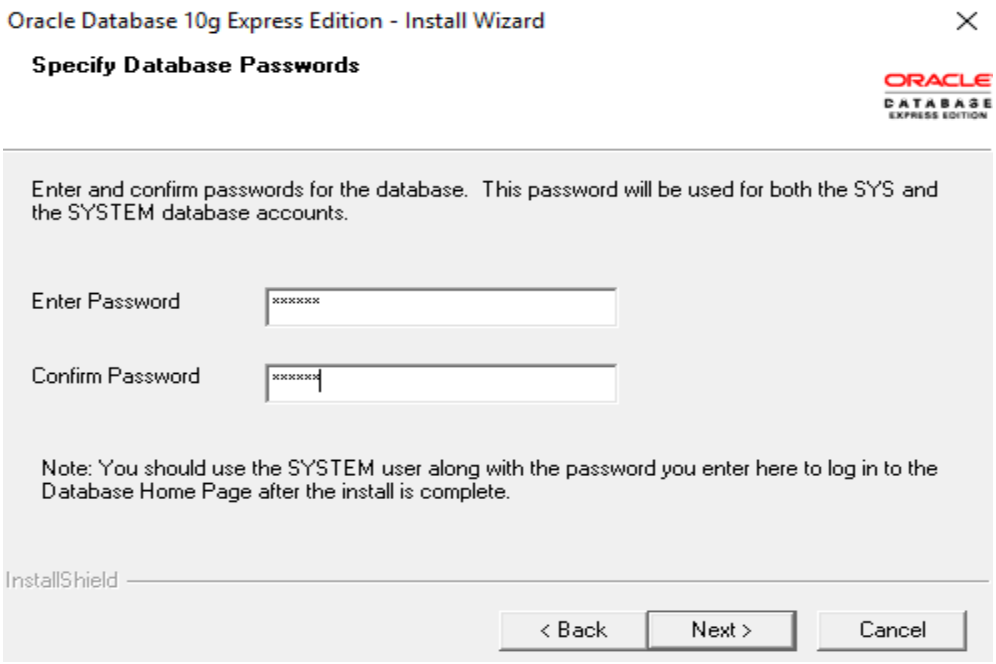
4) Accept license agreement and click on next button



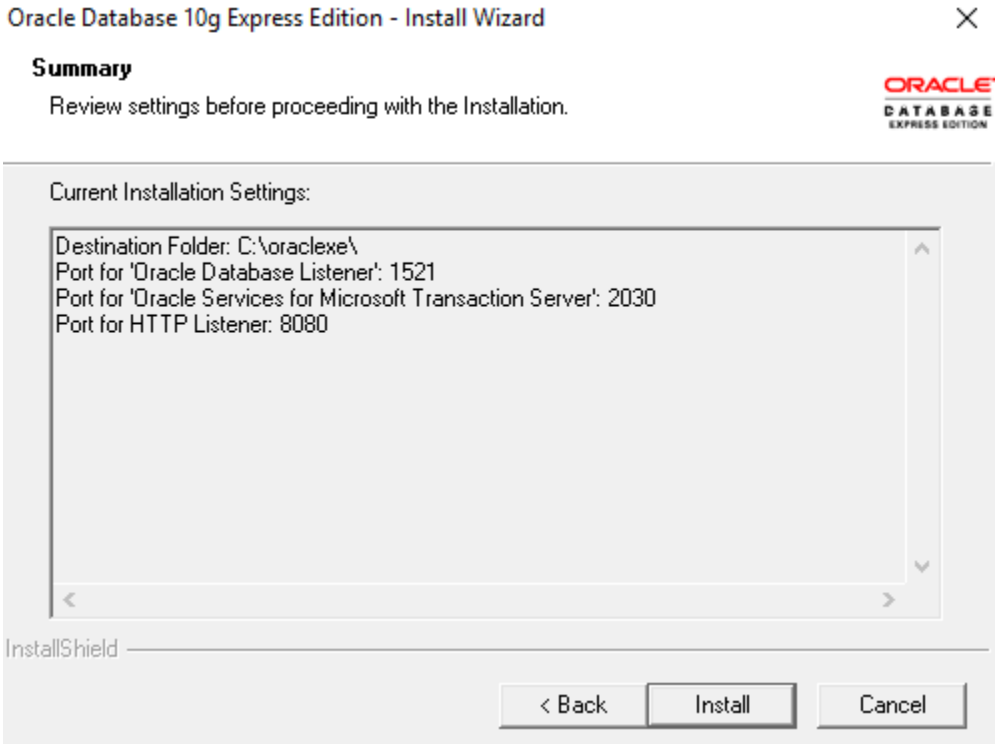
5) Click on next button



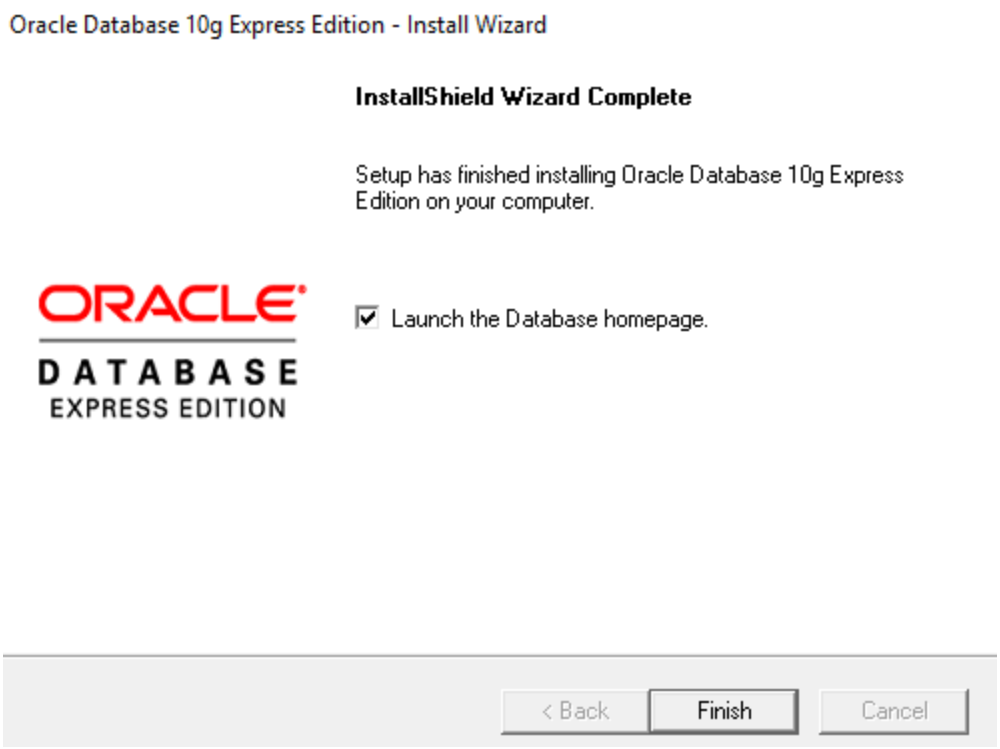
6) Enter password and confirm password for SYS and SYSTEM user. Please remember it because once installation will be over you have to enter it. To make it easy to remember give password as : “oracle”



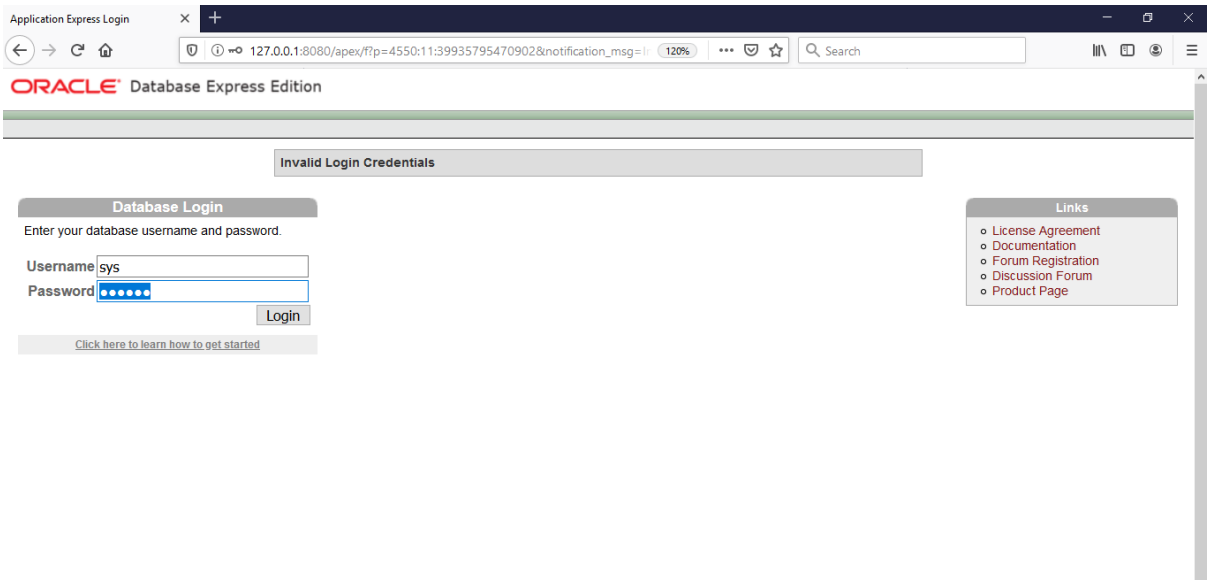
7) Click on install button



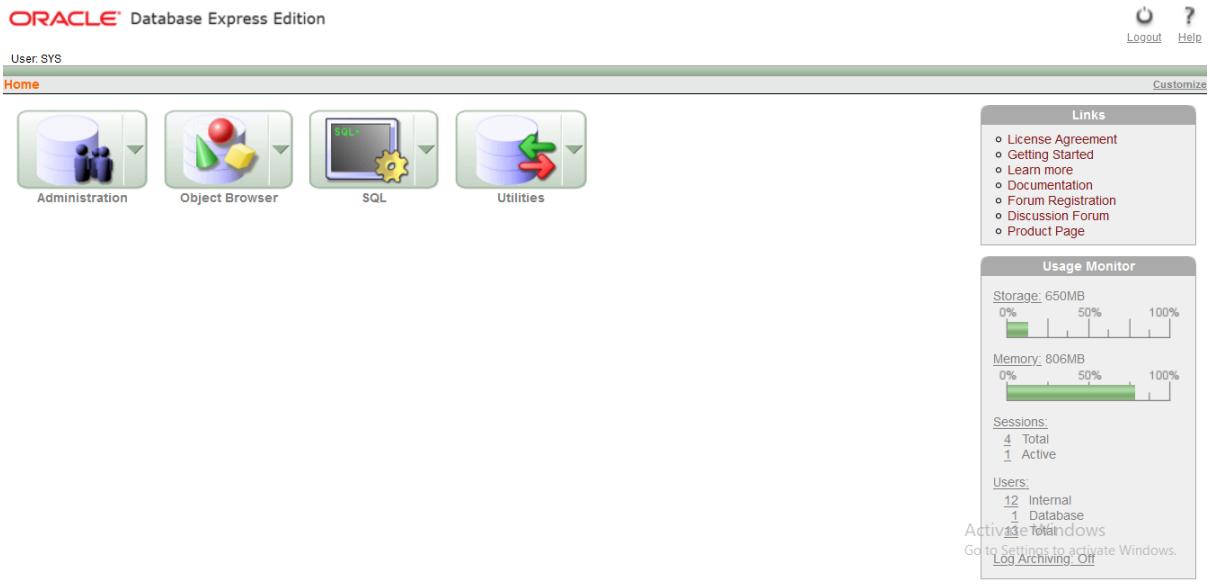
8) Click on finish button.



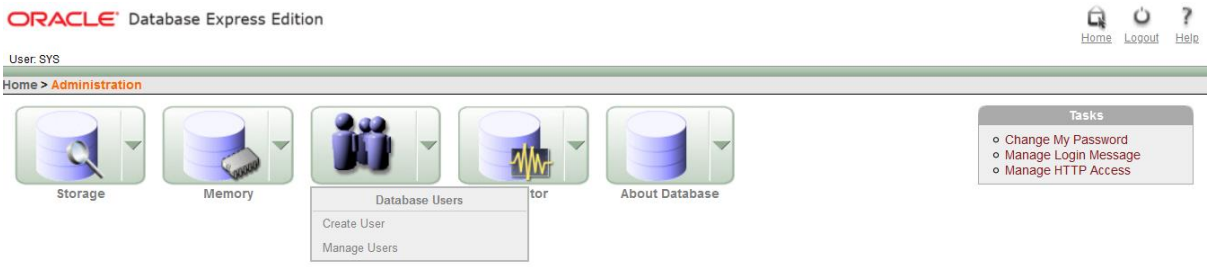
9) Enter username as SYS OR SYSTEM and enter your password (Entered in step: 6)



10) Click on Administration

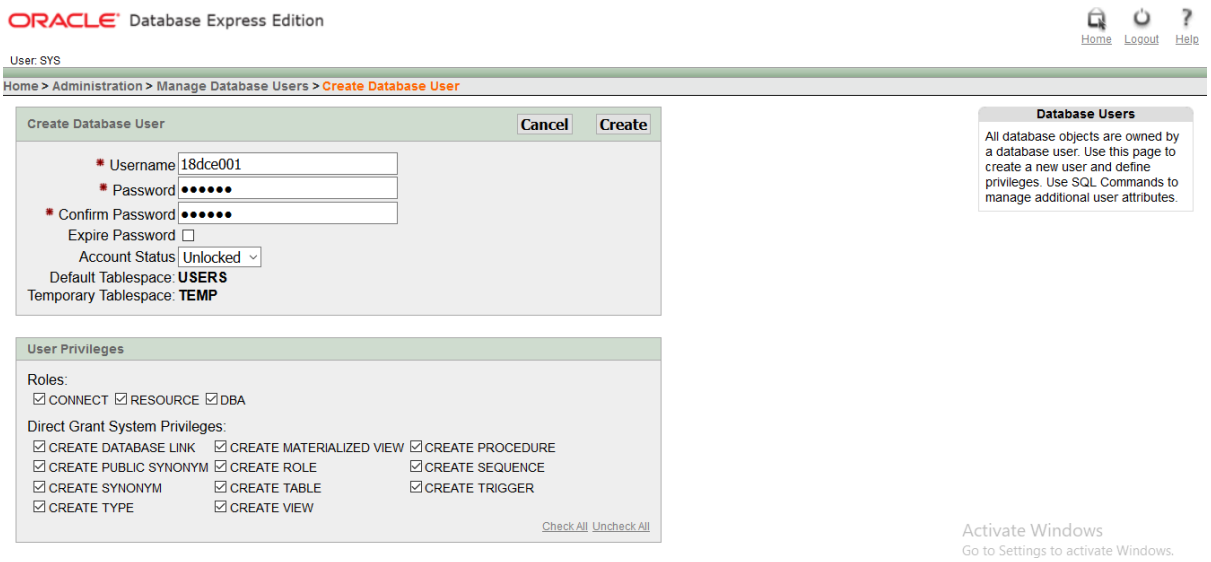


11) Now click on “database user drop down button”. From that click on “create user”.

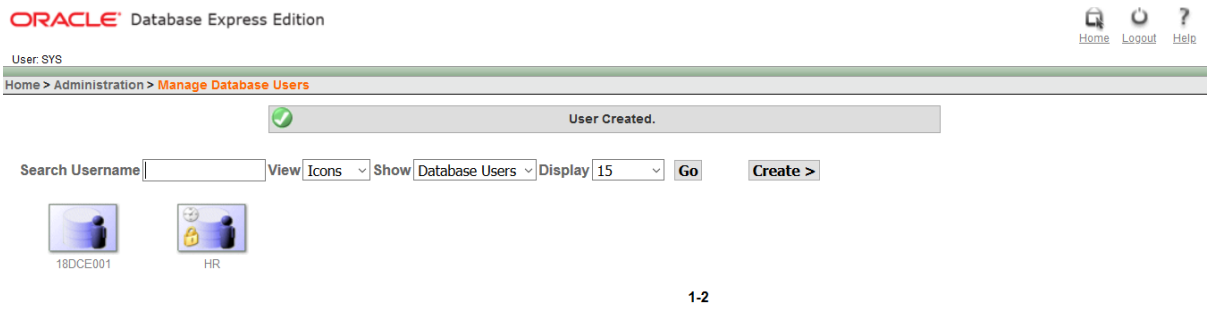


12) Enter your college roll no in username and give password (NEW) and confirm password. Don’t check expire password, make account status

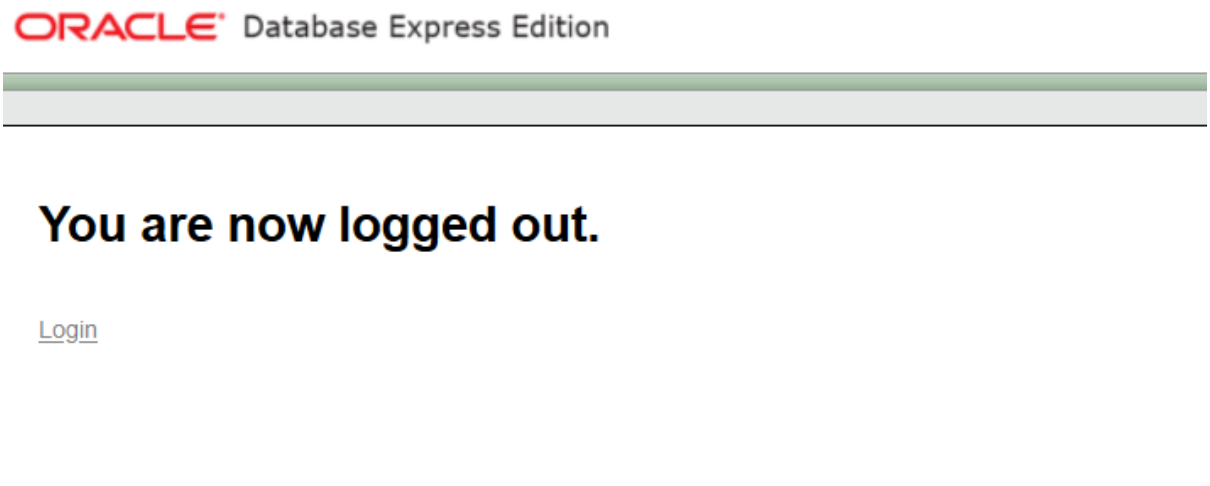
unblocked if it is not. Give all privileges to your user. Finally click on "create" button.



13) This page will be shown to you. Now click on "logout" button.

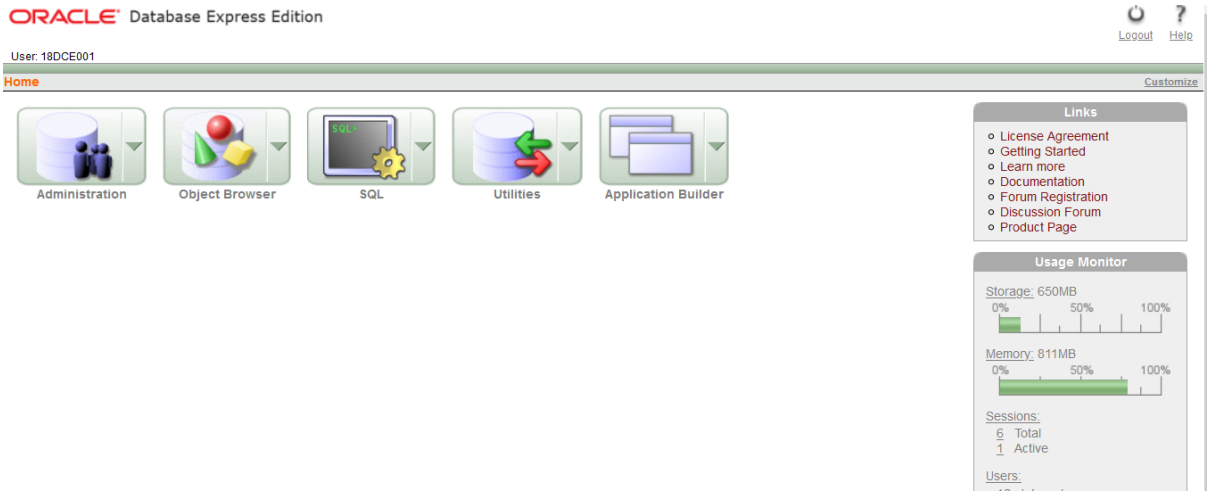


14) Click on login



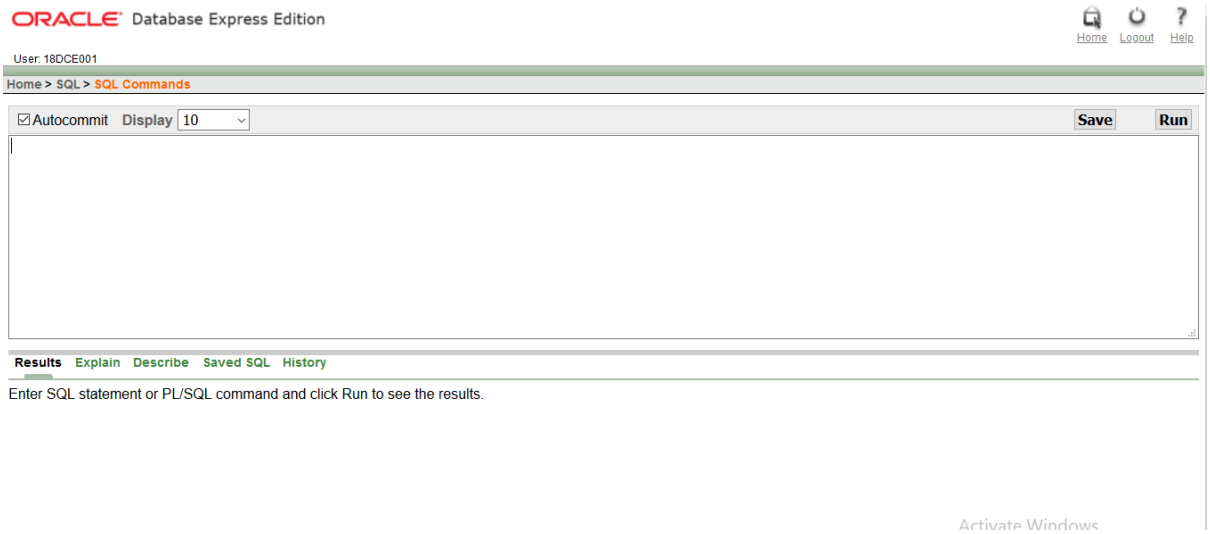
15) Enter username and password that you just created and click on "login" button

16) Click on SQL



17) Click on SQL Commands

18) Congratulation!!! Now you are ready to code SQL and PLSQL.



Conclusion: We learnt how to install oracle 10g and how to use it.

Practical-3

Aim:

(i) Create tables according to the following definition.

● CREATE TABLE DEPOSIT (ACTNO VARCHAR2(5), CNAME VARCHAR2(18), BNAME VARCHAR2(18), AMOUNT NUMBER (8,2), ADATE DATE);

● CREATE TABLE BRANCH (BNAME VARCHAR2(18), CITY VARCHAR2(18));

● CREATE TABLE CUSTOMERS (CNAME VARCHAR2(19), CITY VARCHAR2(18));

● CREATE TABLE BORROW (LOANNO VARCHAR2(5), CNAME VARCHAR2(18), BNAME VARCHAR2(18), AMOUNT NUMBER (8,2));

(ii) Insert the data as shown below.

DEPOSIT

ACTNO	CNAME	BNAME	AMOUNT	ADATE
100	ANIL	VRCE	1000.00	1-MAR-95
101	SUNIL	AJNI	5000.00	4-JAN-96
102	MEHUL	KAROLBAGH	3500.00	17-NOV-95
104	MADHURI	CHANDI	1200.00	17-DEC-95
105	PRMOD	M.G.ROAD	3000.00	27-MAR-96
106	SANDIP	ANDHERI	2000.00	31-MAR-96
107	SHIVANI	VIRAR	1000.00	5-SEP-95
108	KRANTI	NEHRU PLACE	5000.00	2-JUL-95
109	MINU	POWAI	7000.00	10-AUG-95

BRANCH

BNAME	CITY
VRCE	NAGPUR
AJNI	NAGPUR
KAROLBAGH	DELHI
CHANDI	DELHI
DHARAMPETH	NAGPUR
M.G.ROAD	BANGLORE
ANDHERI	BOMBAY
VIRAR	BOMBAY
NEHRU PLACE	DELHI
POWAI	BOMBAY

CUSTOMERS

CNAME	CITY
ANIL	CALCUTTA
SUNIL	DELHI
MEHUL	BARODA
MANDAR	PATNA
MADHURI	NAGPUR
PRAMOD	NAGPUR
SANDIP	SURAT
SHIVANI	BOMBAY
KRANTI	BOMBAY
NAREN	BOMBAY

BORROW

LOANNO	CNAME	BNAME	AMOUNT
201	ANIL	VRCE	1000.00
206	MEHUL	AJNI	5000.00
311	SUNIL	DHARAMPETH	3000.00
321	MADHURI	ANDHERI	2000.00
375	PRMOD	VIRAR	8000.00
481	KRANTI	NEHRU PLACE	3000.00

Theory:

DDL: A data definition language (DDL) is a computer language used to create and modify the structure of database objects in a database.

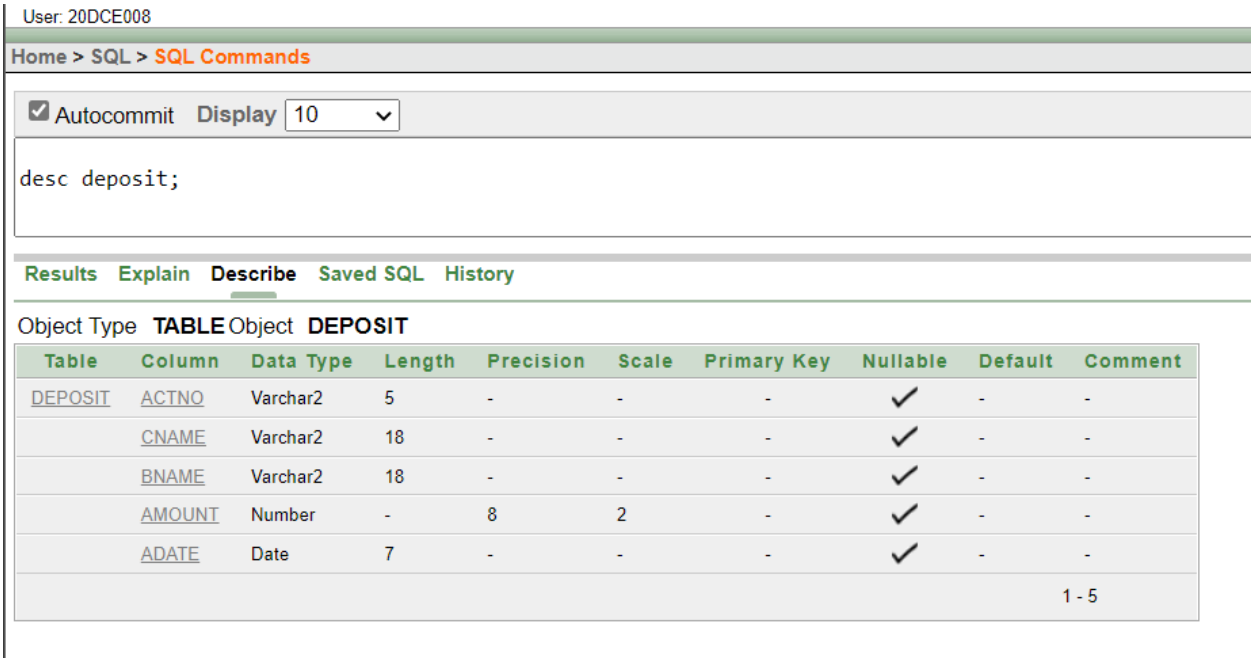
DDL-CREATE command: This command builds a new table and has a predefined syntax. The create command syntax is:
CREATE TABLE [table name] ([column definitions]) [table parameters];
For example:
CREATE TABLE DEPOSIT (ACTNO VARCHAR2(5), CNAME VARCHAR2(18), BNAME VARCHAR2(18), AMOUNT NUMBER (8,2), ADATE DATE);

DML: A data manipulation language (DML) is a family of computer languages including commands permitting users to manipulate data in a database. This manipulation involves inserting data into database tables, retrieving existing data, deleting data from existing tables and modifying existing data. DML is mostly incorporated in SQL databases.

DML_INSERT command: This command adds one or more records to a database table. The insert command syntax is:
INSERT INTO [table name] [column(s)] VALUES [value(s)].
For Example:
INSERT INTO DEPOSIT (ACTNO, CNAME, BNAME, AMOUNT, ADATE)
VALUES('101','SUNIL','AJNI',5000.00,'4-JAN-96');

Program:

1. Describe deposit, branch.



User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
desc branch;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **BRANCH**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
BRANCH	BNAME	Varchar2	18	-	-	-	✓	-	-
	CITY	Varchar2	18	-	-	-	✓	-	-

1 - 2

2. Describe borrow, customers.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
DESC BORROW
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **BORROW**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
BORROW	LOANNO	Varchar2	5	-	-	-	✓	-	-
	CNAME	Varchar2	18	-	-	-	✓	-	-
	BNAME	Varchar2	18	-	-	-	✓	-	-
	AMOUNT	Number	-	8	2	-	✓	-	-

1 - 4

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
DESC customers
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **CUSTOMERS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CUSTOMERS	CNAME	Varchar2	19	-	-	-	✓	-	-
	CITY	Varchar2	18	-	-	-	✓	-	-

1 - 2

3. List all data from table DEPOSIT.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
select *from deposit;
```

Results Explain Describe Saved SQL History

ACTNO	CNAME	BNAME	AMOUNT	ADATE
100	ANIL	VRCE	1000	01-MAR-95
101	SUNIL	AJNI	5000	04-JAN-96
102	MEHUL	KAROLBAGH	3500	17-NOV-95
104	MADHURI	CHANDI	1200	17-DEC-95
105	PRMOD	M.G.ROAD	3000	27-MAR-96
106	SANDIP	ANDHERI	2000	31-MAR-96
107	SHIVANI	VIRAR	1000	05-SEP-95
108	KRANTI	NEHRUPLACE	5000	02-JUL-95
109	MINU	POWAI	7000	10-AUG-95

9 rows returned in 0.01 seconds CSV Export

4. List all data from table BORROW.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
select *from borrow;
```

Results Explain Describe Saved SQL History

LOANNO	CNAME	BNAME	AMOUNT
201	ANIL	VRCE	1000
206	MEHUL	AJNI	5000
311	SUNIL	DHARAMPETH	3000
321	MADHURI	ANDHERI	2000
375	PRMOD	VIRAR	8000
481	KRANTI	NEHRU PALACE	3000

6 rows returned in 0.00 seconds CSV Export

5. List all data from table CUSTOMERS.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

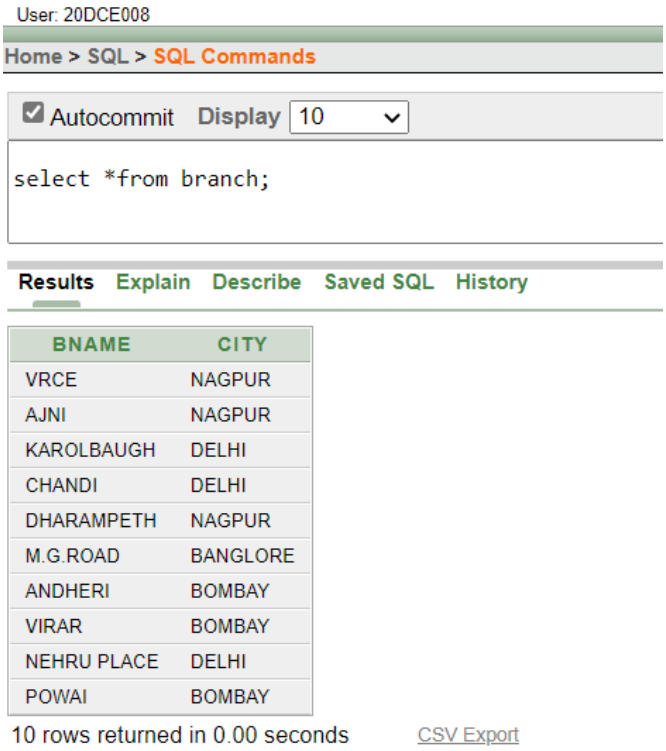
```
select *from customers;
```

Results Explain Describe Saved SQL History

CNAME	CITY
ANIL	CALCUTTA
SUNIL	DELHI
MEHUL	BARODA
MANDAR	PATNA
MADHURI	NAGPUR
PRAMOD	NAGPUR
SANDIP	SURAT
SHIVANI	BOMBAY
KRANTI	BOMBAY
NAREN	BOMBAY

10 rows returned in 0.00 seconds CSV Export

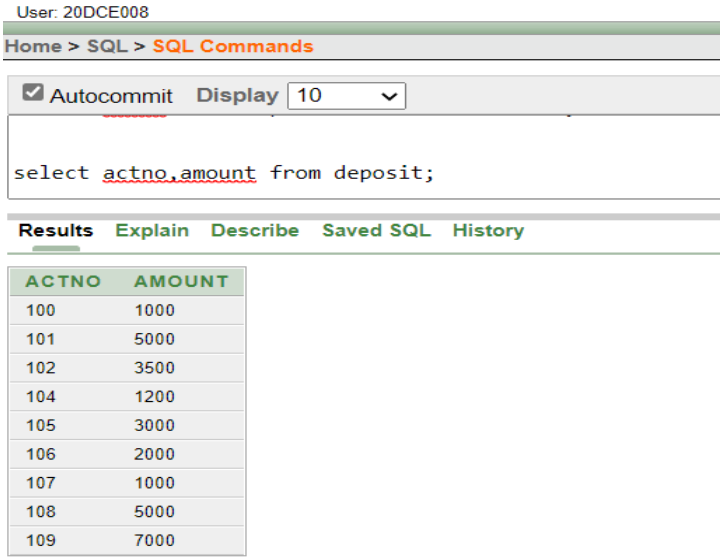
6. List all data from table BRANCH.



BNAME	CITY
VRCE	NAGPUR
AJNI	NAGPUR
KAROLBAUGH	DELHI
CHANDI	DELHI
DHARAMPETH	NAGPUR
M.G.ROAD	BANGLORE
ANDHERI	BOMBAY
VIRAR	BOMBAY
NEHRU PLACE	DELHI
POWAI	BOMBAY

10 rows returned in 0.00 seconds CSV Export

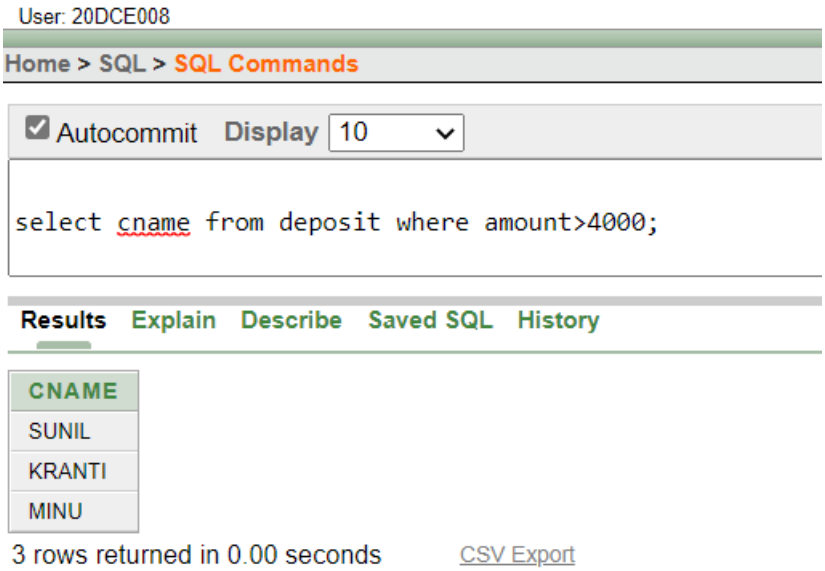
7. Give account no and amount of depositors.



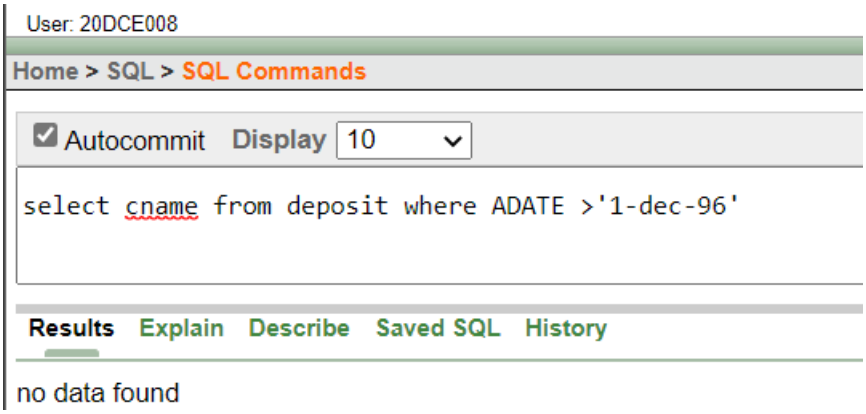
ACTNO	AMOUNT
100	1000
101	5000
102	3500
104	1200
105	3000
106	2000
107	1000
108	5000
109	7000

9 rows returned in 0.02 seconds CSV Export

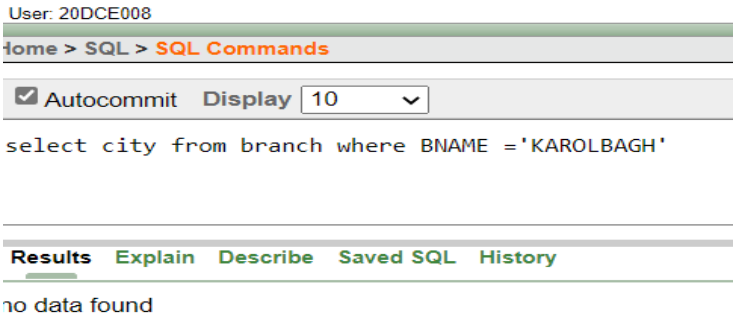
8. Give name of depositors having amount greater than 4000.



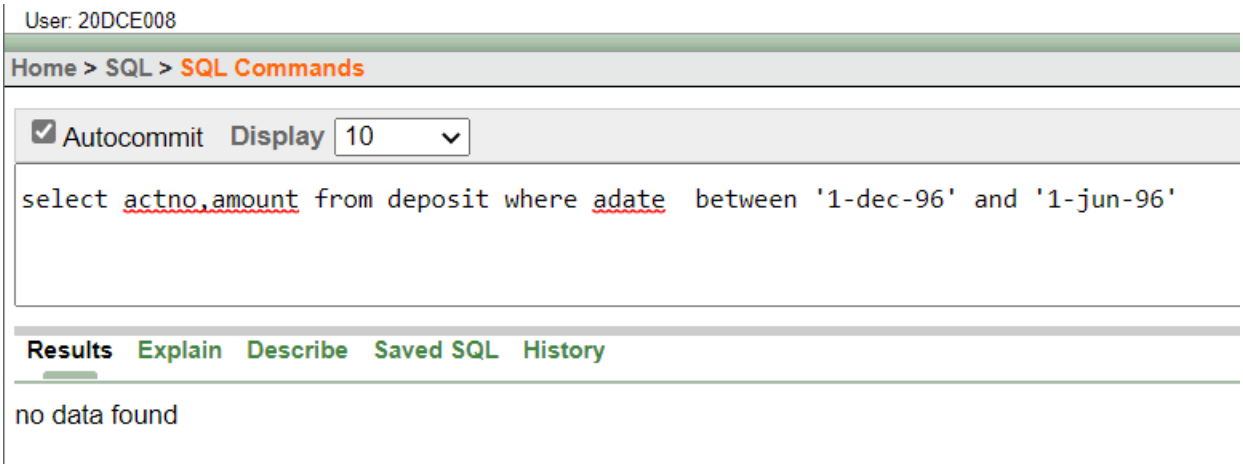
9. Give name of customers who opened account after date '1-12-96'.



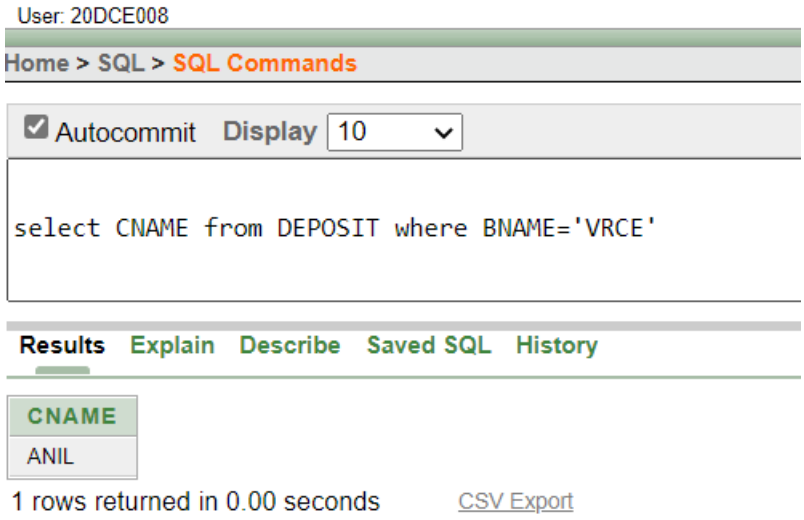
10. Give name of city where branch karolbagh is located.



11. Give account no and amount of customer having account opened between date 1-12-96 and 1-6- 96.



12. Give names of depositors having account at VRCE.



Conclusion:

We learnt some basic functions like creating tables, entering values into it and performing some operations on it.

Practical-4

Aim: Create the below given table and insert the data accordingly.

Create Table Job (job_id, job_title, min_sal, max_sal)

COLUMN NAME	DATA TYPE
job_id	Varchar2(15)
job_title	Varchar2(30)
min_sal	Number(7,2)
max_sal	Number(7,2)

Create table Employee (emp_no, emp_name, emp_sal, emp_comm, dept_no, l_name, dept_name, job_id, location, manager_id, hiredate)

COLUMN NAME	DATA TYPE
emp_no	Number(3)
emp_name	Varchar2(30)
emp_sal	Number(8,2)
emp_comm	Number(6,1)
dept_no	Number(3)
l_name	Varchar2(30)
dept_name	Varchar2(30)
job_id	Varchar2(15)
location	Varchar2(15)
manager_id	Number(5)
hiredate	Date

Create table deposit(a_no,cname,bname,amount,a_date).

COLUMN NAME	DATA TYPE
a_no	Varchar2(5)
cname	Varchar2(15)
bname	Varchar2(10)
amount	Number(7,2)
a_date	Date

Create table borrow (loanno, cname, bname, amount)

COLUMN NAME	DATA TYPE
loanno	Varchar2(5)
cname	Varchar2(15)
bname	Varchar2(10)
amount	Varchar2(7,2)

Insert following values in the table Employee.

emp_	emp_nam	emp_sal	emp_com	dept_	l_name	dept_name	job_id	location	manag id	hiredate
101	Smith	800		20	shah	machine learning	fi_mgr	toronto	105	09-aug-96
102	Snehal	1600	300	25	gupta	data science	lec	las vegas		14-mar-96
103	Adama	1100	0	20	wales	machine learning	mk_mgr	ontario	105	30-nov-95
104	Aman	3000		15	sharma	virtual reality	comp_op	mexico	12	02-oct-97
105	Anita	5000	50,000	10	patel	big data analytics	comp_op	germany	107	01-jan-98
106	Sneha	2450	24,500	10	joseph	big data analytics	fi_acc	melbourne	105	26-sep-97
107	Anamika	2975		30	jha	artificial intelligence	it_prog	new york		15jul-97

Insert following values in the table Job.

job_id	job_name	min_sal	max_sal
it_prog	Programmer	4000	10000
mk_mgr	Marketing manager	9000	15000
fi_mgr	Finance manager	8200	12000
fi_acc	Account	4200	9000
lec	Lecturer	6000	17000
comp_op	Computer Operator	1500	3000

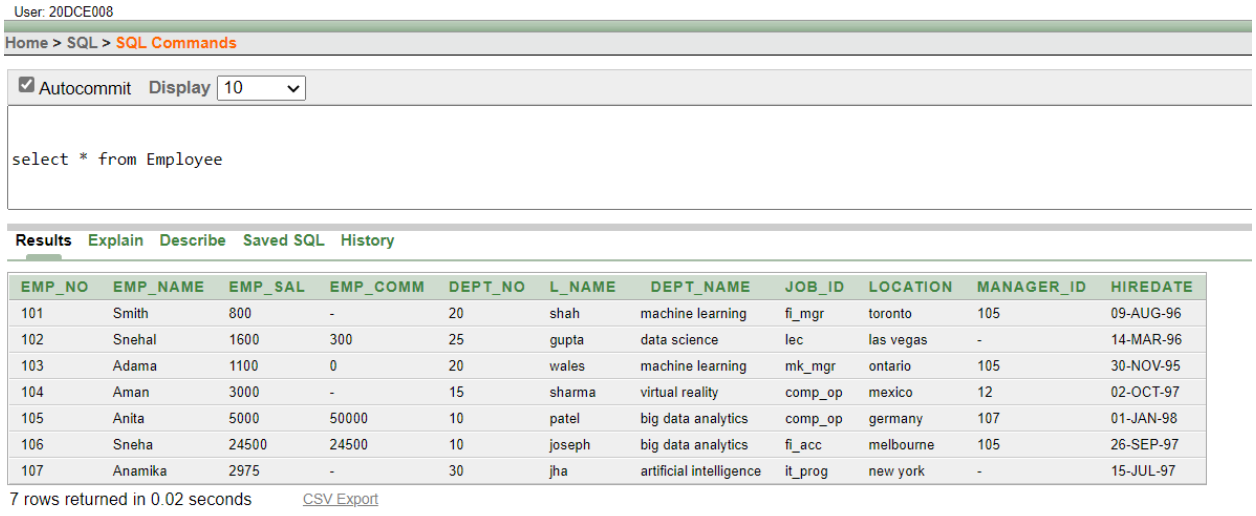
Insert following values in the table deposit

A_no	cname	Bname	Amount	date
101	Anil	andheri	7000	01-jan-06
102	sunil	virar	5000	15-jul-06
103	jay	villeparle	6500	12-mar-06
104	vijay	andheri	8000	17-sep-06
105	keyur	dadar	7500	19-nov-06
106	mayur	borivali	5500	21-dec-06

Program:

Perform following queries :

(1) Retrieve all data from employee, jobs and deposit.



EMP_NO	EMP_NAME	EMP_SAL	EMP_COMM	DEPT_NO	L_NAME	DEPT_NAME	JOB_ID	LOCATION	MANAGER_ID	HIREDATE
101	Smith	800	-	20	shah	machine learning	fi_mgr	toronto	105	09-AUG-96
102	Snehal	1600	300	25	gupta	data science	lec	las vegas	-	14-MAR-96
103	Adama	1100	0	20	wales	machine learning	mk_mgr	ontario	105	30-NOV-95
104	Aman	3000	-	15	sharma	virtual reality	comp_op	mexico	12	02-OCT-97
105	Anita	5000	50000	10	patel	big data analytics	comp_op	germany	107	01-JAN-98
106	Sneha	24500	24500	10	joseph	big data analytics	fi_acc	melbourne	105	26-SEP-97
107	Anamika	2975	-	30	jha	artificial intelligence	it_prog	new york	-	15-JUL-97

7 rows returned in 0.02 seconds CSV Export

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
select * from deposit_1
```

Results Explain Describe Saved SQL History

A_NO	CNAME	BNAME	AMOUNT	A_DATE
101	ANIL	ANDHERI	7000	01-JAN-06
102	SUNIL	VIRAR	5000	15-JUL-06
103	JAY	VILLEPARLE	6500	12-MAR-06
104	VIJAY	ANDHERI	8000	17-SEP-06
105	KEYUR	DADAR	7500	19-NOV-06
106	MAYUR	BORIVALI	5500	21-DEC-06

6 rows returned in 0.00 seconds CSV Export

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

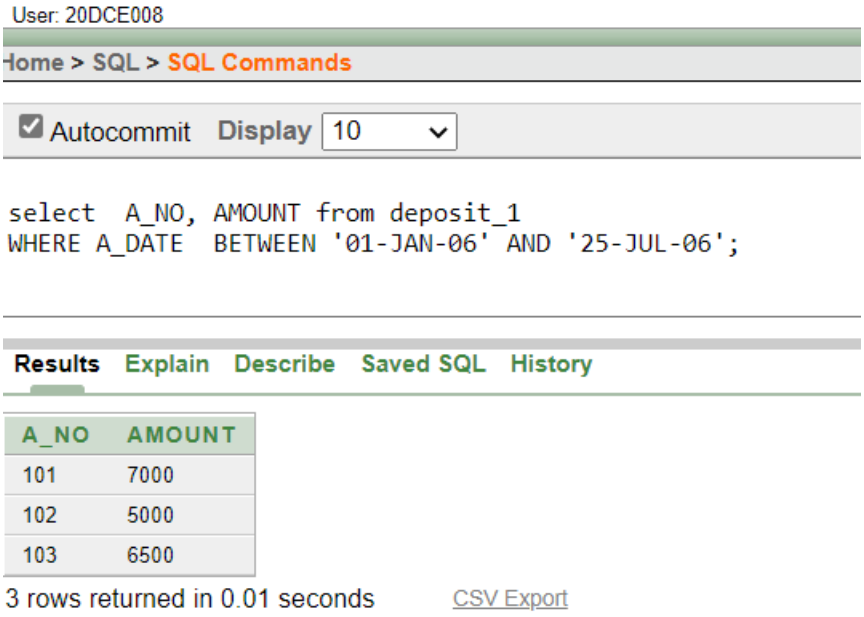
```
select * from job
```

Results Explain Describe Saved SQL History

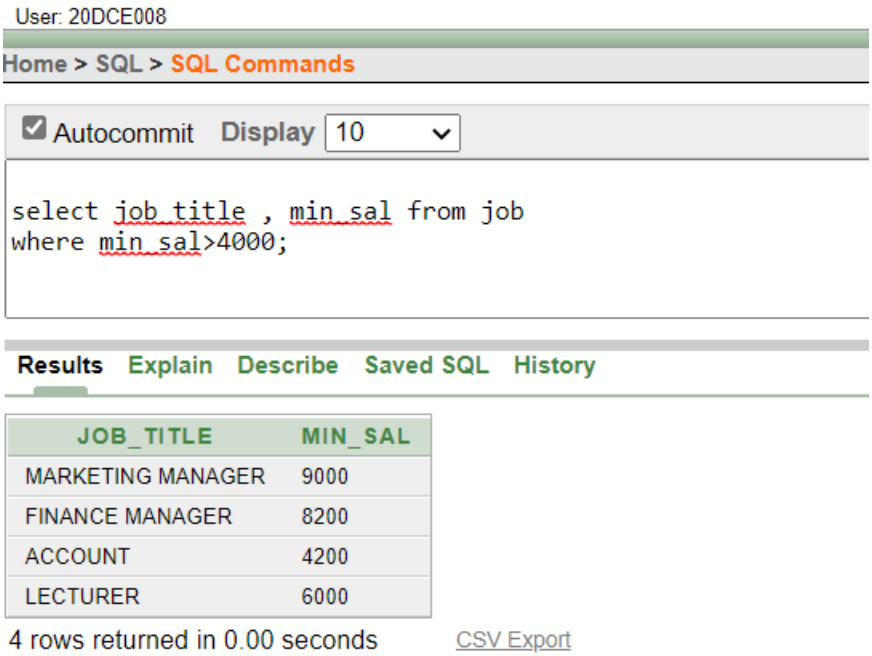
JOB_ID	JOB_TITLE	MIN_SAL	MAX_SAL
IT_PROG	PROGRAMMER	4000	10000
MK_MGR	MARKETING MANAGER	9000	15000
FI_MGR	FINANCE MANAGER	8200	12000
FI_ACC	ACCOUNT	4200	9000
LEC	LECTURER	6000	17000
COMP_OP	COMPUTER OPERATOR	1500	3000

6 rows returned in 0.00 seconds CSV Export

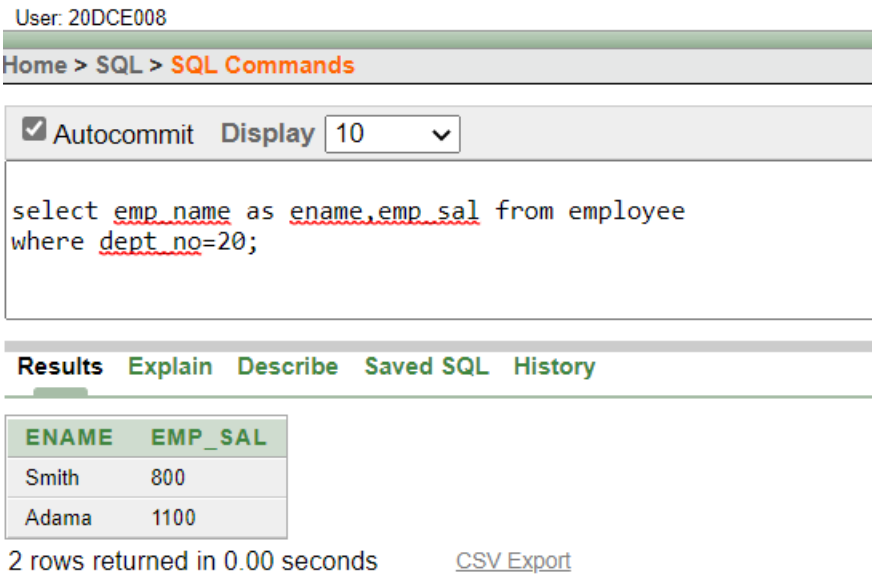
(2) Give details of account no. and deposited rupees of customers having account opened between dates 01-01-06 and 25-07-06.



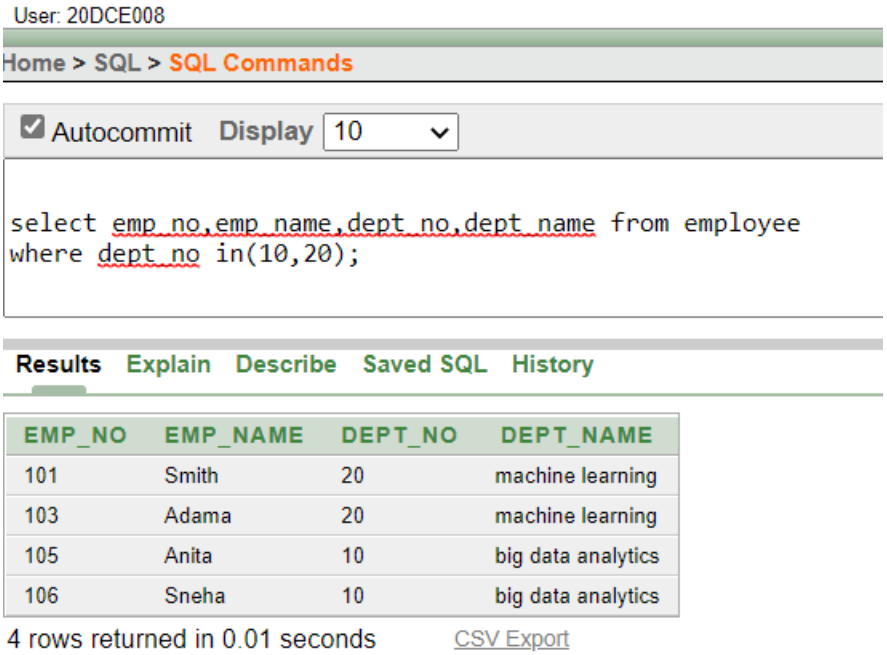
(3) Display all jobs with minimum salary is greater than 4000.



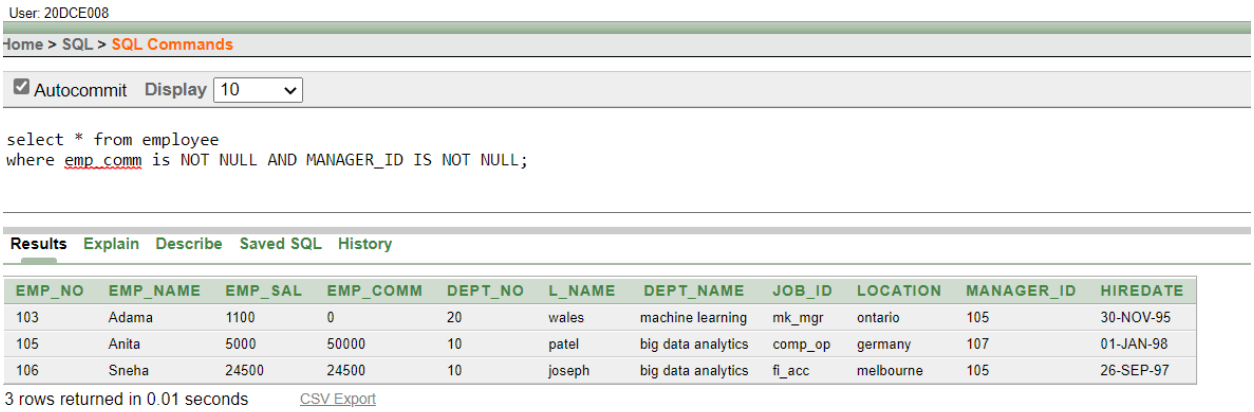
(4) Display name and salary of employee whose department no is 20. Give alias name to name of employee.



(5) Display employee no, name and department details of those employee whose department lies in (10,20).



(6) Display the non-null values of employees.



User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

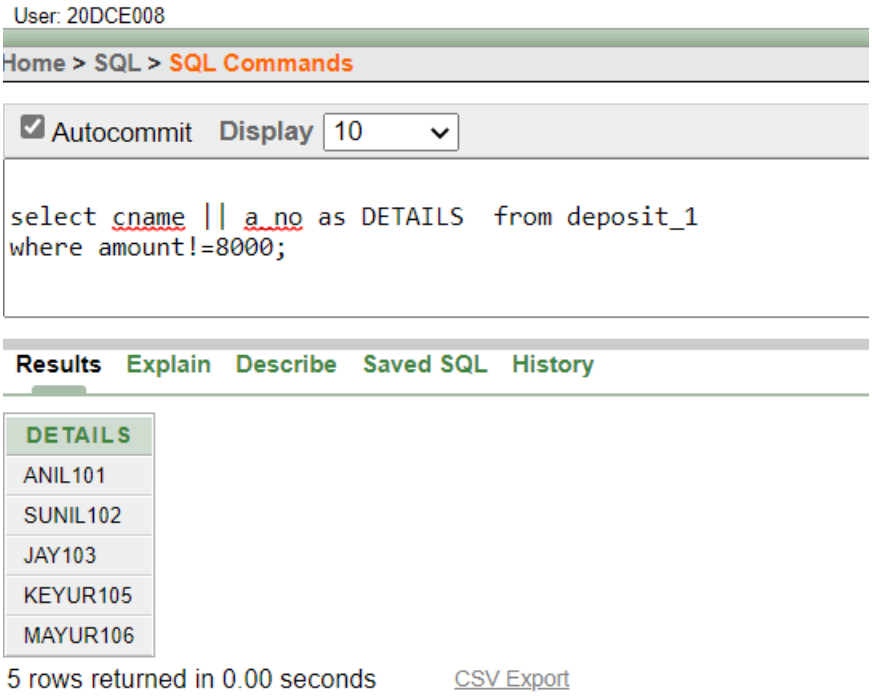
```
select * from employee
where emp_comm is NOT NULL AND MANAGER_ID IS NOT NULL;
```

Results Explain Describe Saved SQL History

EMP_NO	EMP_NAME	EMP_SAL	EMP_COMM	DEPT_NO	L_NAME	DEPT_NAME	JOB_ID	LOCATION	MANAGER_ID	HIREDATE
103	Adama	1100	0	20	wales	machine learning	mk_mgr	ontario	105	30-NOV-95
105	Anita	5000	50000	10	patel	big data analytics	comp_op	germany	107	01-JAN-98
106	Sneha	24500	24500	10	joseph	big data analytics	fi_acc	melbourne	105	26-SEP-97

3 rows returned in 0.01 seconds CSV Export

(7) Display name of customer along with its account no (both column should be displayed as one) whose amount is not equal to 8000 Rs.



User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

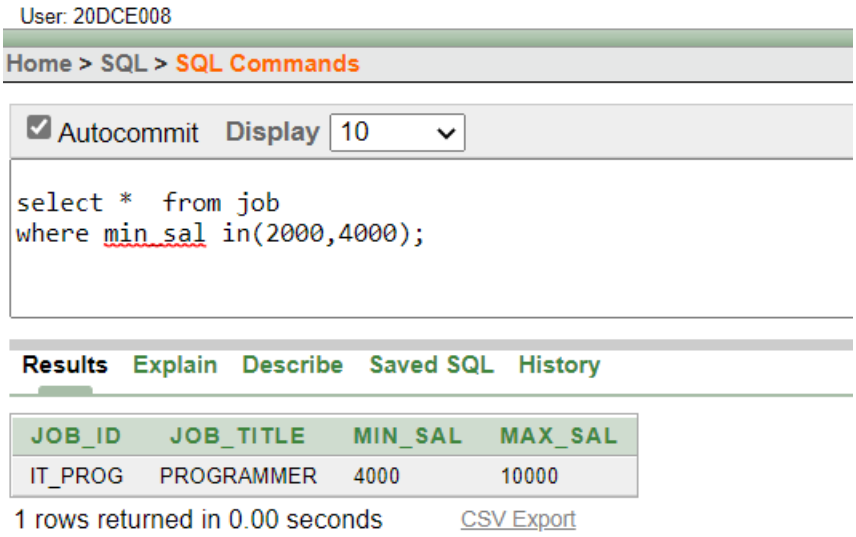
```
select cname || a_no as DETAILS  from deposit_1
where amount!=8000;
```

Results Explain Describe Saved SQL History

DETAILS
ANIL101
SUNIL102
JAY103
KEYUR105
MAYUR106

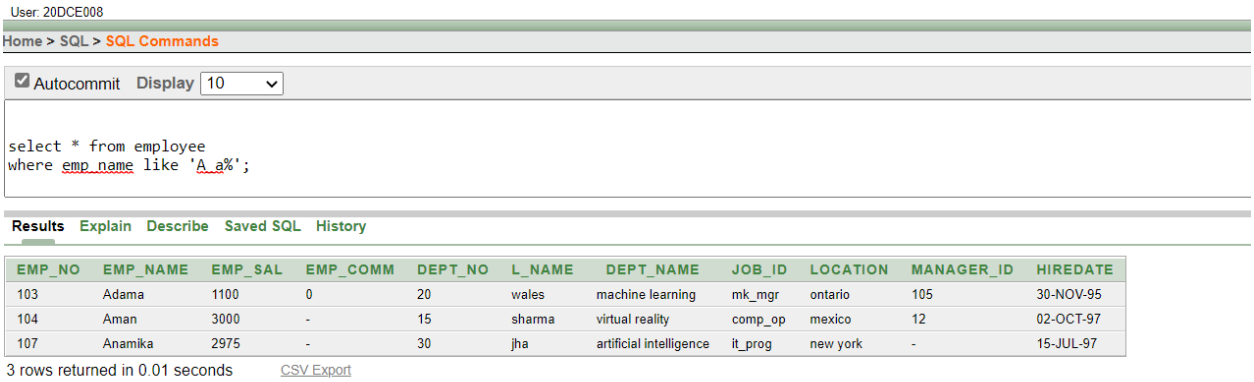
5 rows returned in 0.00 seconds CSV Export

(8)Display the content of job details with minimum salary either 2000 or 4000.

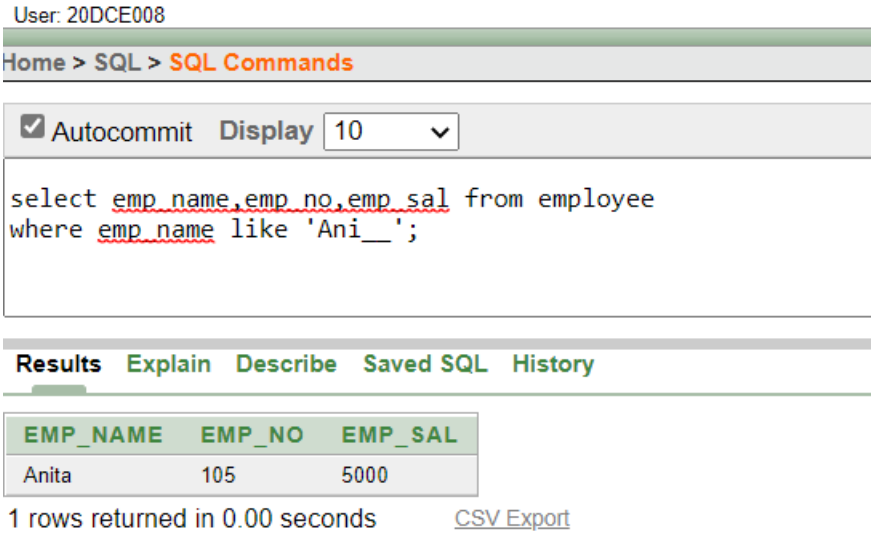


To study various options of LIKE predicate

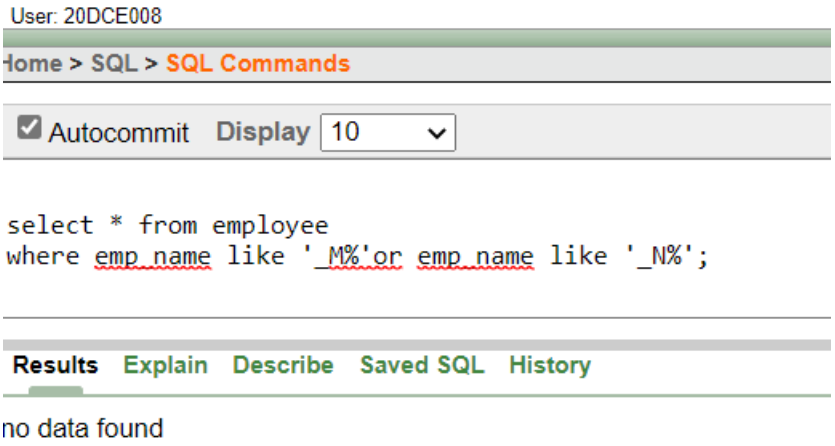
(1) Display all employee whose name start with ‘A’ and third character is ‘‘a’.



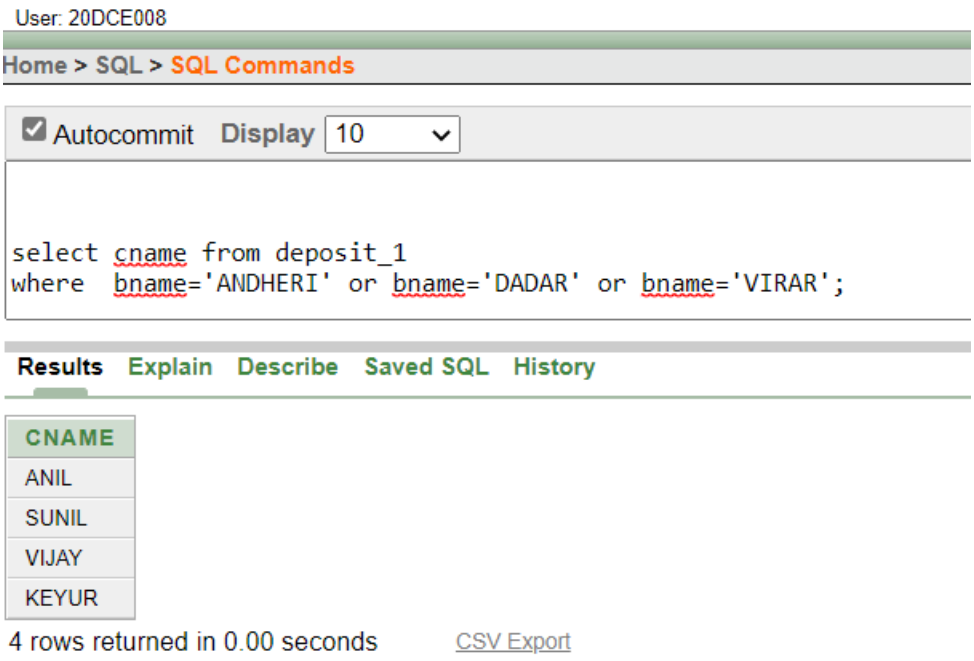
(2) Display name, number and salary of those employees whose name is 5 characters long and first three characters are ‘Ani’.



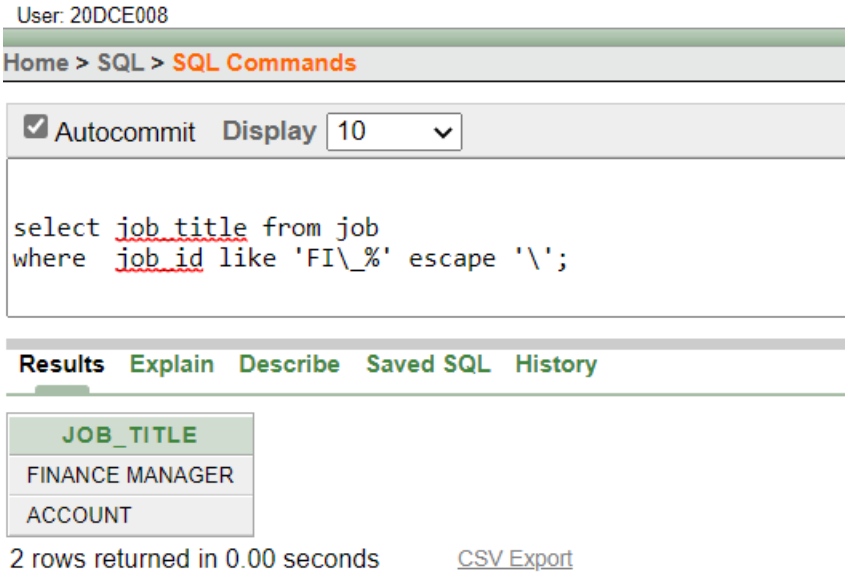
(3) Display all information of employee whose second character of name is either ‘M’ or ‘N’.



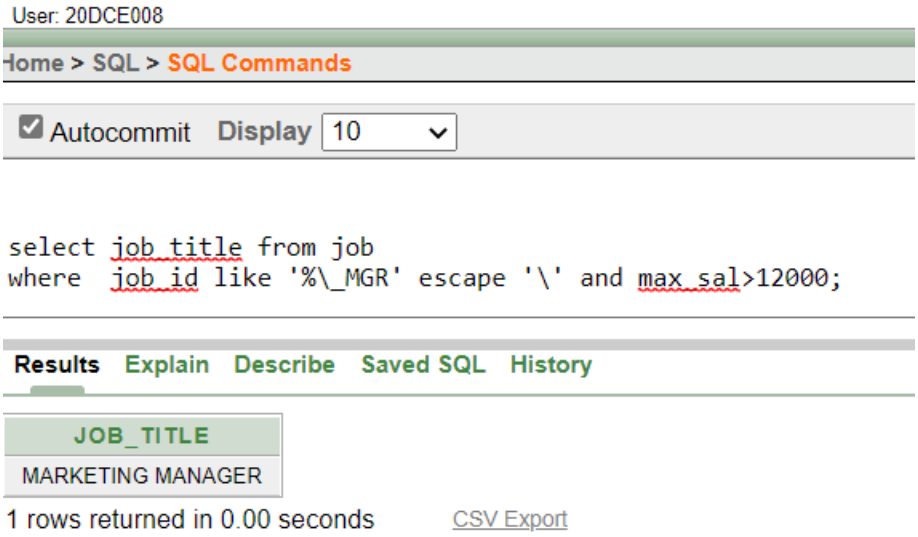
(4) Find the list of all customer name whose branch is in ‘andheri’ or ‘dadar’ or ‘virar’.



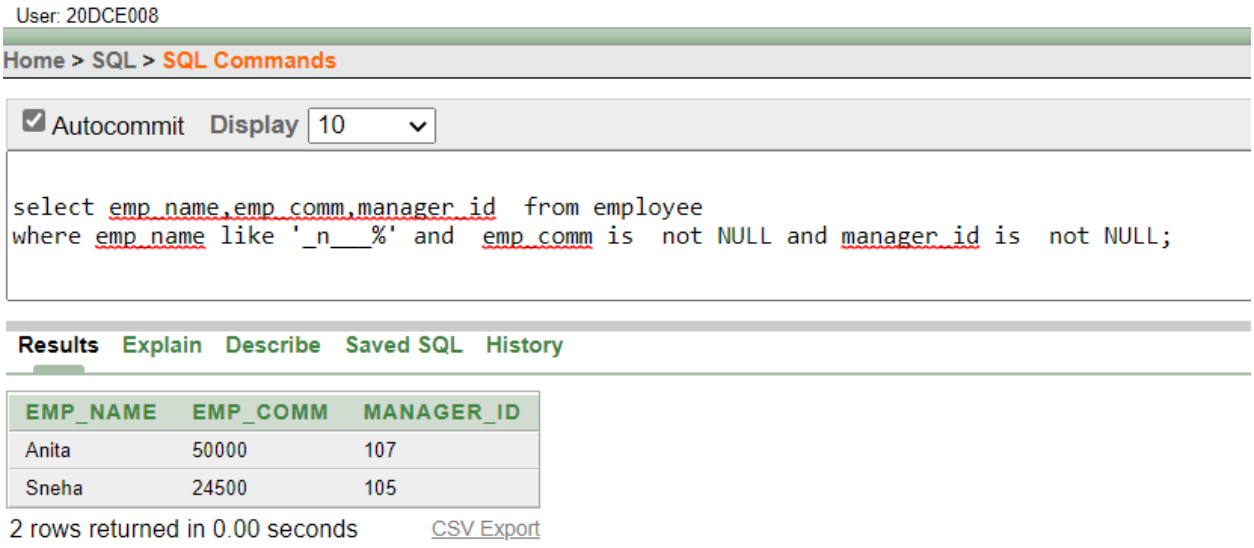
(5) Display the job name whose first three character in job id field is ‘FI_’.



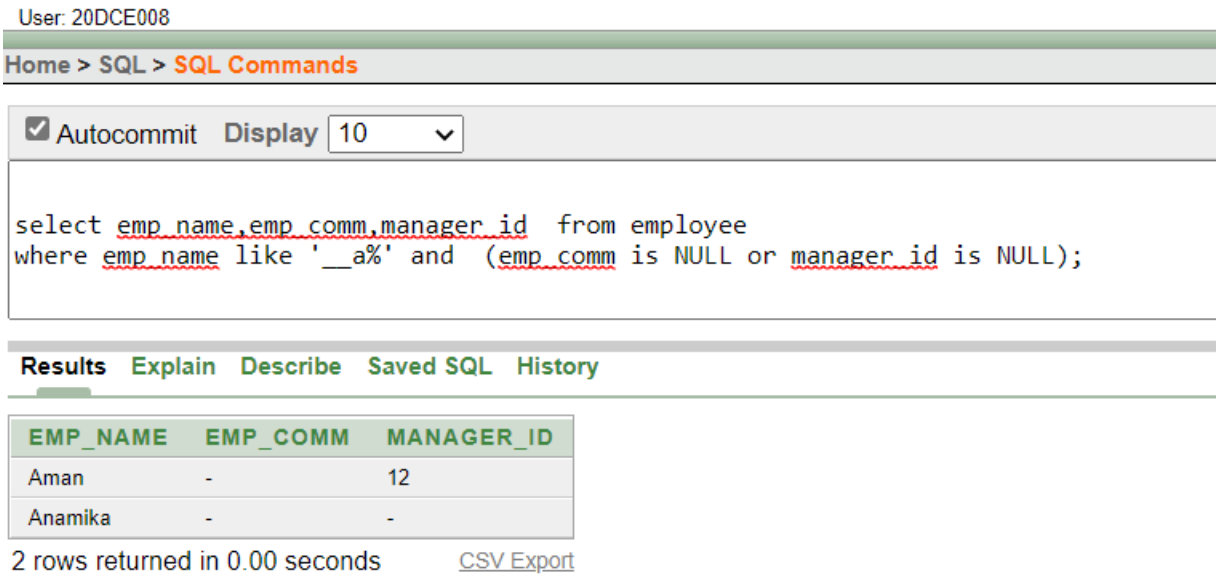
(6) Display the title/name of job who's last three character are '_MGR' and their maximum salary is greater than Rs 12000.



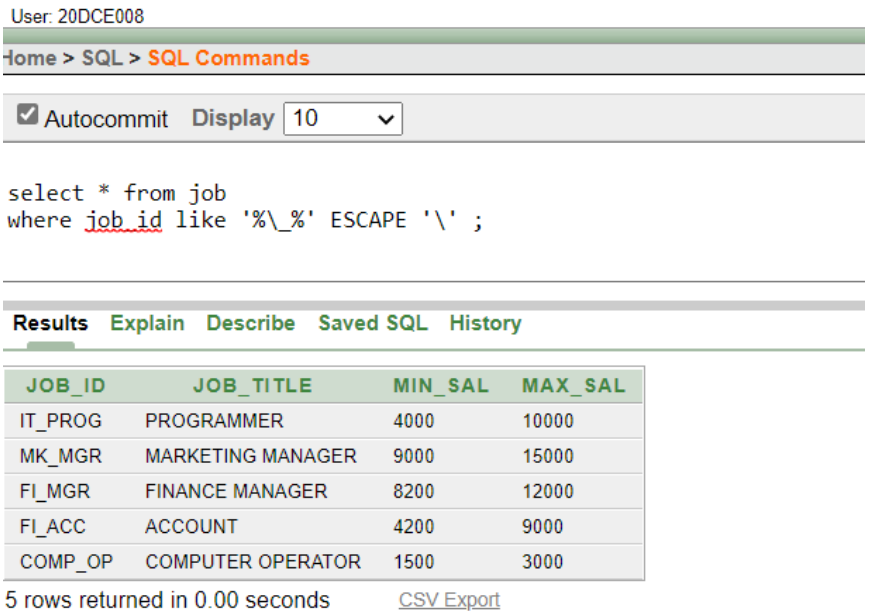
(7) Display the non-null values of employees and also employee name second character should be 'n' and string should be 5-character long.



(8) Display the null values of employee and also employee name's third character should be 'a'.



(9) What will be output if you are giving LIKE predicate as '%_%' ESCAPE '\'



Conclusion:

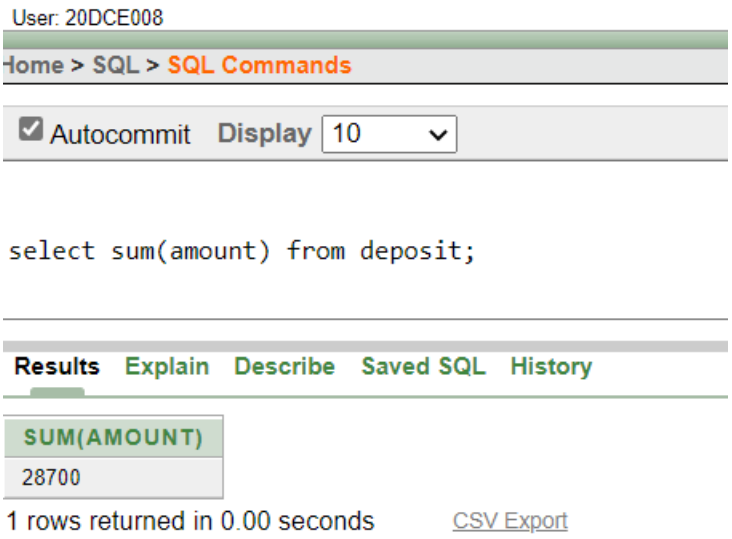
Through this practical we learnt about the Like predicate.

Practical-5

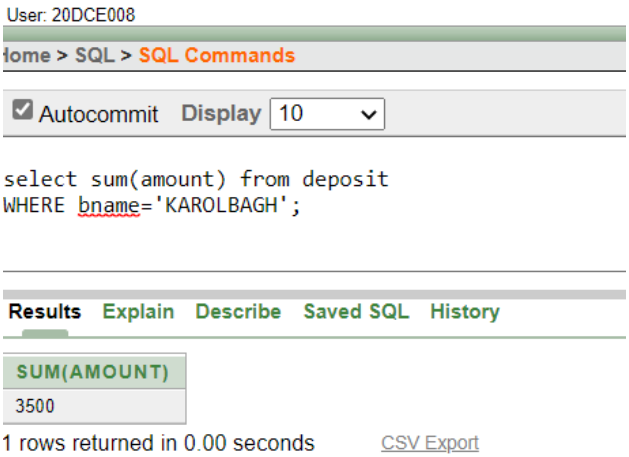
Aim: To Perform various data manipulation commands, aggregate functions and sorting concept on all created tables.

Program:

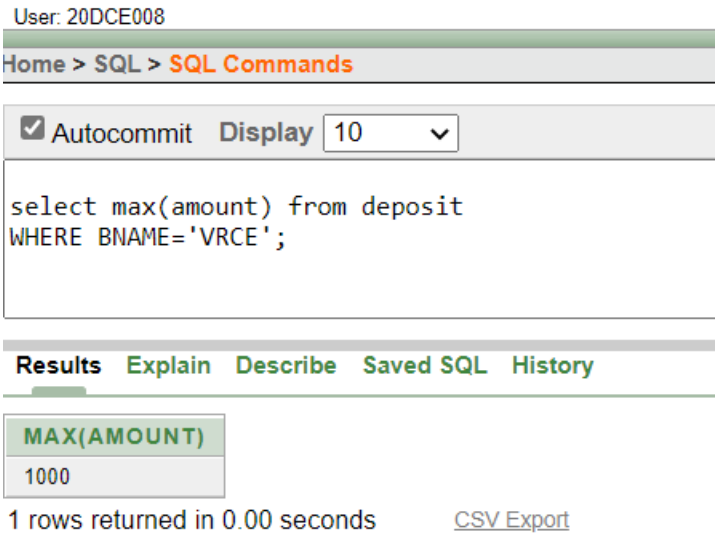
(1) List total deposit from deposit.



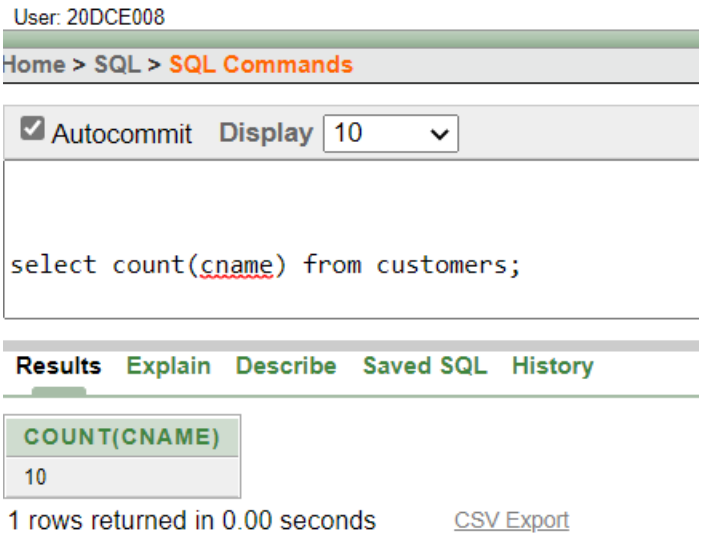
(2) List total loan from karolbagh branch



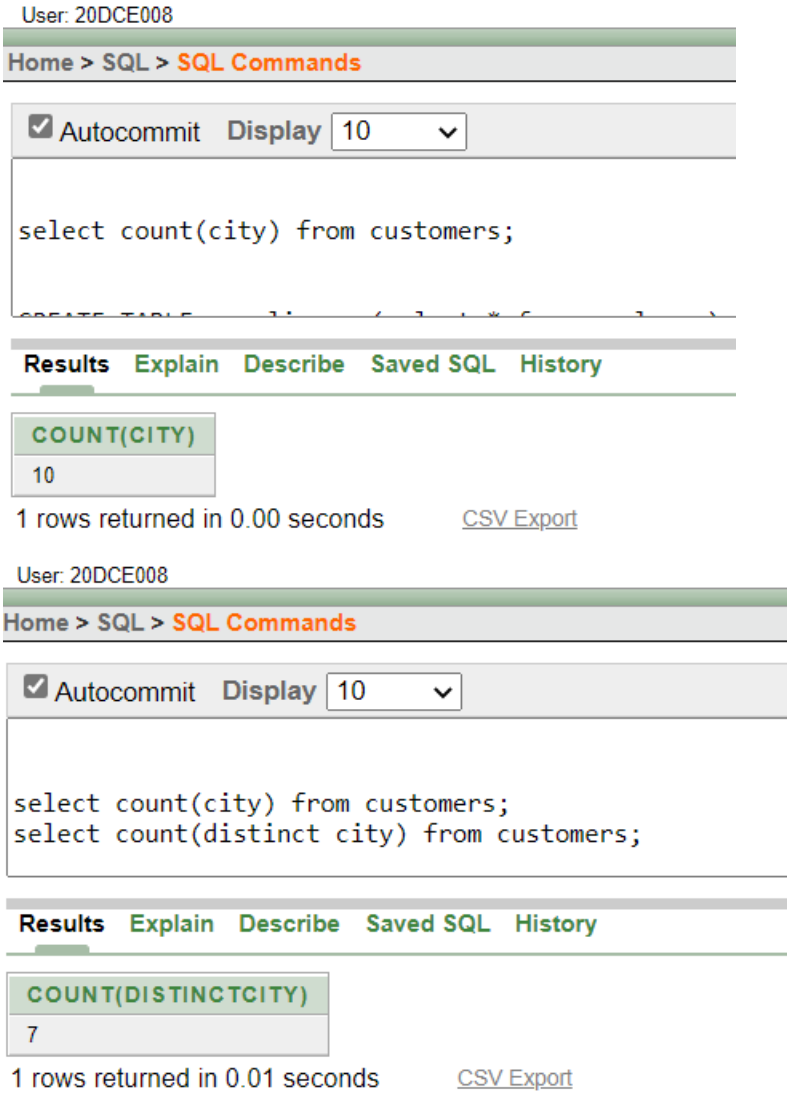
(3) Give maximum loan from branch vrce.



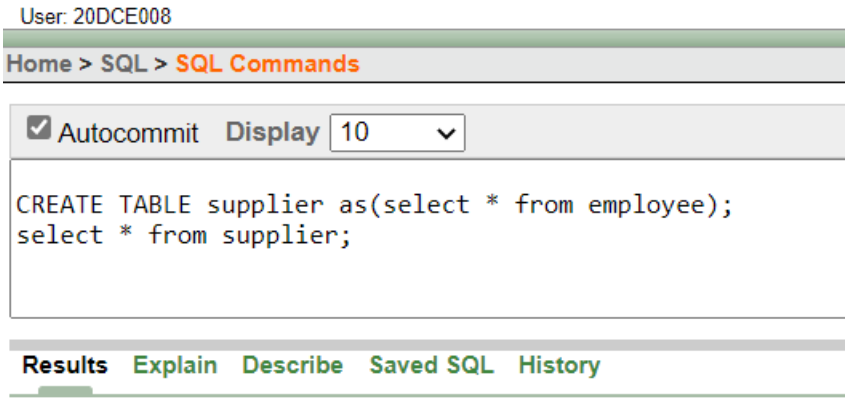
(4) Count total number of customers

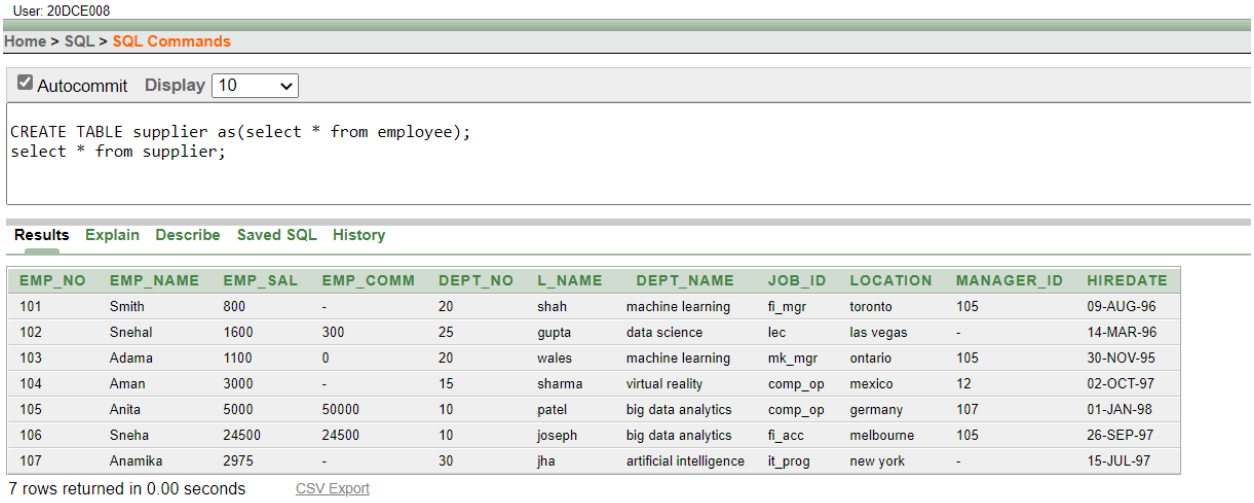


(5) Count total number of customer's cities.



(6) Create table supplier from employee with all the columns.





EMP_NO	EMP_NAME	EMP_SAL	EMP_COMM	DEPT_NO	L_NAME	DEPT_NAME	JOB_ID	LOCATION	MANAGER_ID	HIREDATE
101	Smith	800	-	20	shah	machine learning	fi_mgr	toronto	105	09-AUG-96
102	Snehal	1600	300	25	gupta	data science	lec	las vegas	-	14-MAR-96
103	Adama	1100	0	20	wales	machine learning	mk_mgr	ontario	105	30-NOV-95
104	Aman	3000	-	15	sharma	virtual reality	comp_op	mexico	12	02-OCT-97
105	Anita	5000	50000	10	patel	big data analytics	comp_op	germany	107	01-JAN-98
106	Sneha	24500	24500	10	joseph	big data analytics	fi_acc	melbourne	105	26-SEP-97
107	Anamika	2975	-	30	jha	artificial intelligence	it_prog	new york	-	15-JUL-97

7 rows returned in 0.00 seconds CSV Export

(7) Create table sup1 from employee with first two columns.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
CREATE TABLE sup1 as(select emp_no,emp_name from employee);
select * from sup1;
```

Results Explain Describe Saved SQL History

Table created.

0.01 seconds

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

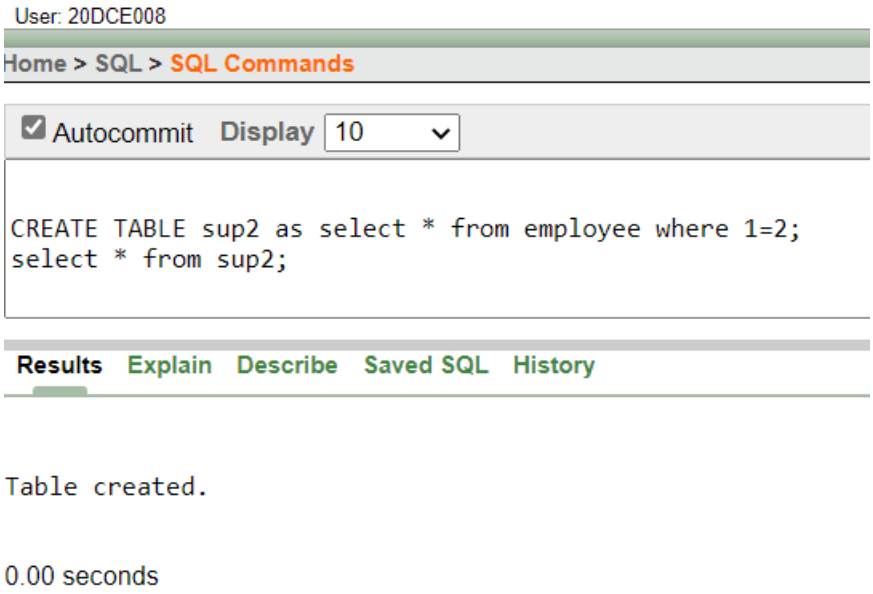
```
CREATE TABLE sup1 as(select emp_no,emp_name from employee);
select * from sup1;
```

Results Explain Describe Saved SQL History

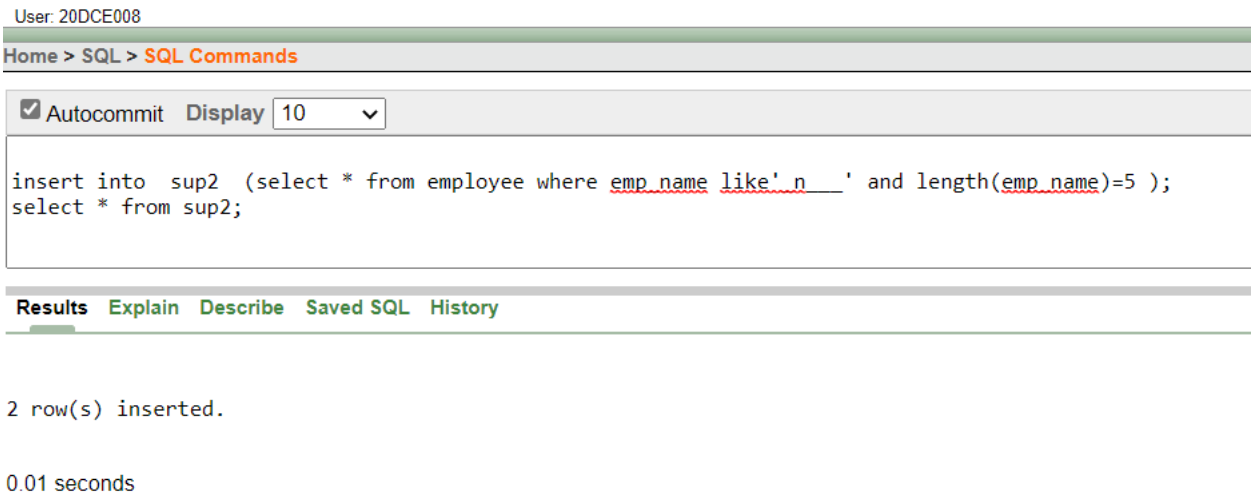
EMP_NO	EMP_NAME
101	Smith
102	Snehal
103	Adama
104	Aman
105	Anita
106	Sneha
107	Anamika

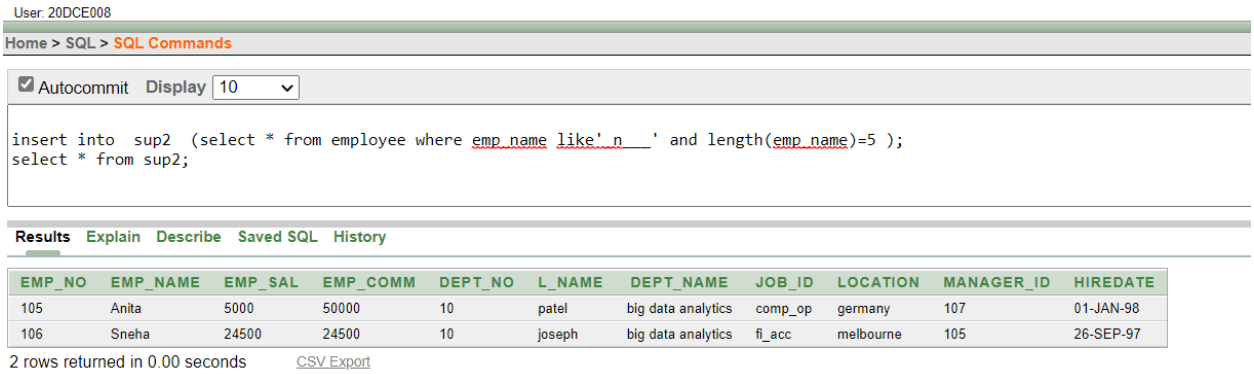
7 rows returned in 0.02 seconds CSV Export

(8) Create table sup2 from employee with no data

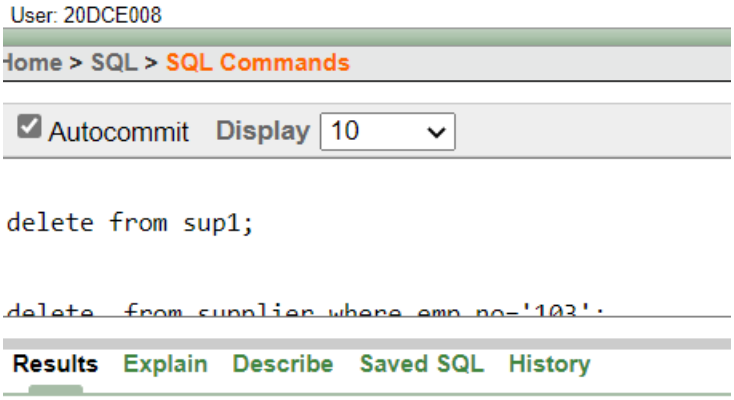


(9) Insert the data into sup2 from employee whose second character should be 'n' and string should be 5 characters long in employee name field.

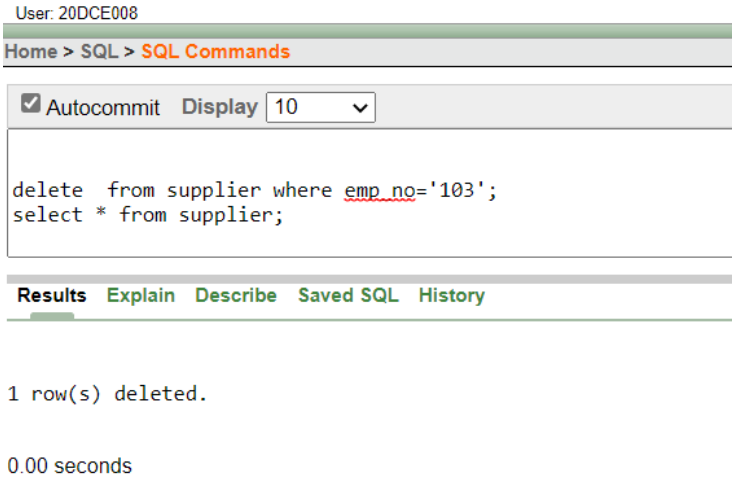


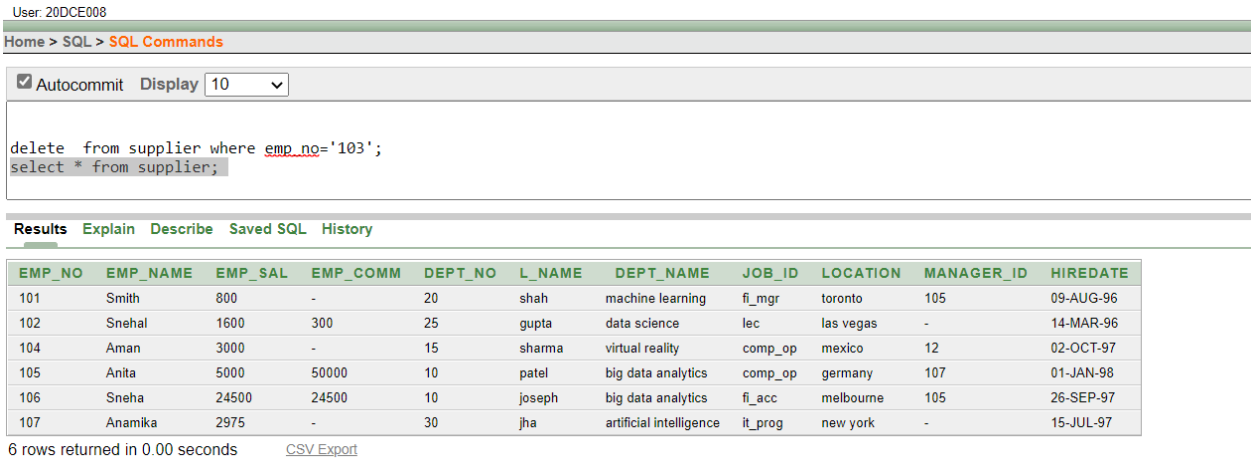


(10) Delete all the rows from sup1.



(11) Delete the detail of supplier whose sup_no is 103.




User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

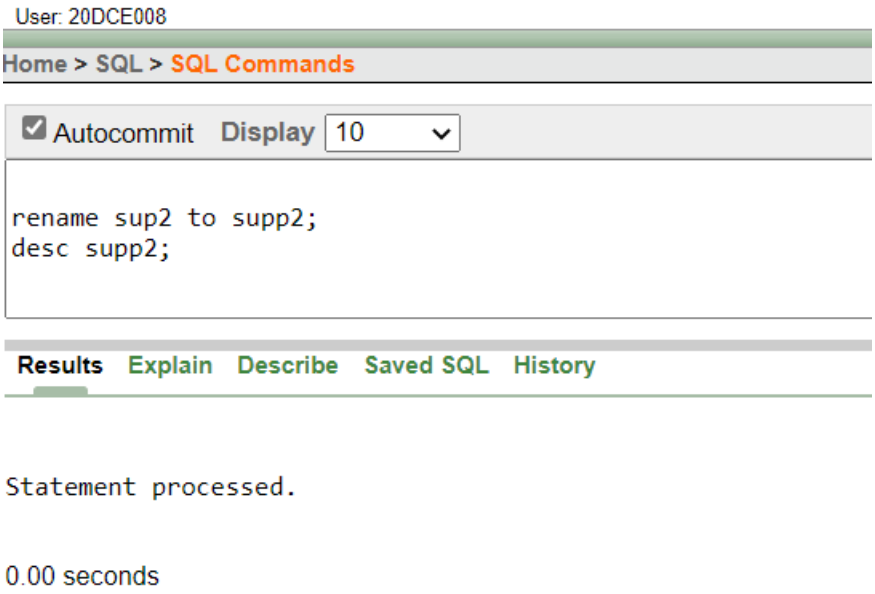
```
delete  from supplier where emp_no='103';
select * from supplier;
```

Results Explain Describe Saved SQL History

EMP_NO	EMP_NAME	EMP_SAL	EMP_COMM	DEPT_NO	L_NAME	DEPT_NAME	JOB_ID	LOCATION	MANAGER_ID	HIREDATE
101	Smith	800	-	20	shah	machine learning	fi_mgr	toronto	105	09-AUG-96
102	Snehal	1600	300	25	gupta	data science	lec	las vegas	-	14-MAR-96
104	Aman	3000	-	15	sharma	virtual reality	comp_op	mexico	12	02-OCT-97
105	Anita	5000	50000	10	patel	big data analytics	comp_op	germany	107	01-JAN-98
106	Sneha	24500	24500	10	joseph	big data analytics	fi_acc	melbourne	105	26-SEP-97
107	Anamika	2975	-	30	jha	artificial intelligence	it_prog	new york	-	15-JUL-97

6 rows returned in 0.00 seconds CSV Export

(12) Rename the table sup2.


User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
rename sup2 to supp2;
desc supp2;
```

Results Explain Describe Saved SQL History

```
Statement processed.

0.00 seconds
```

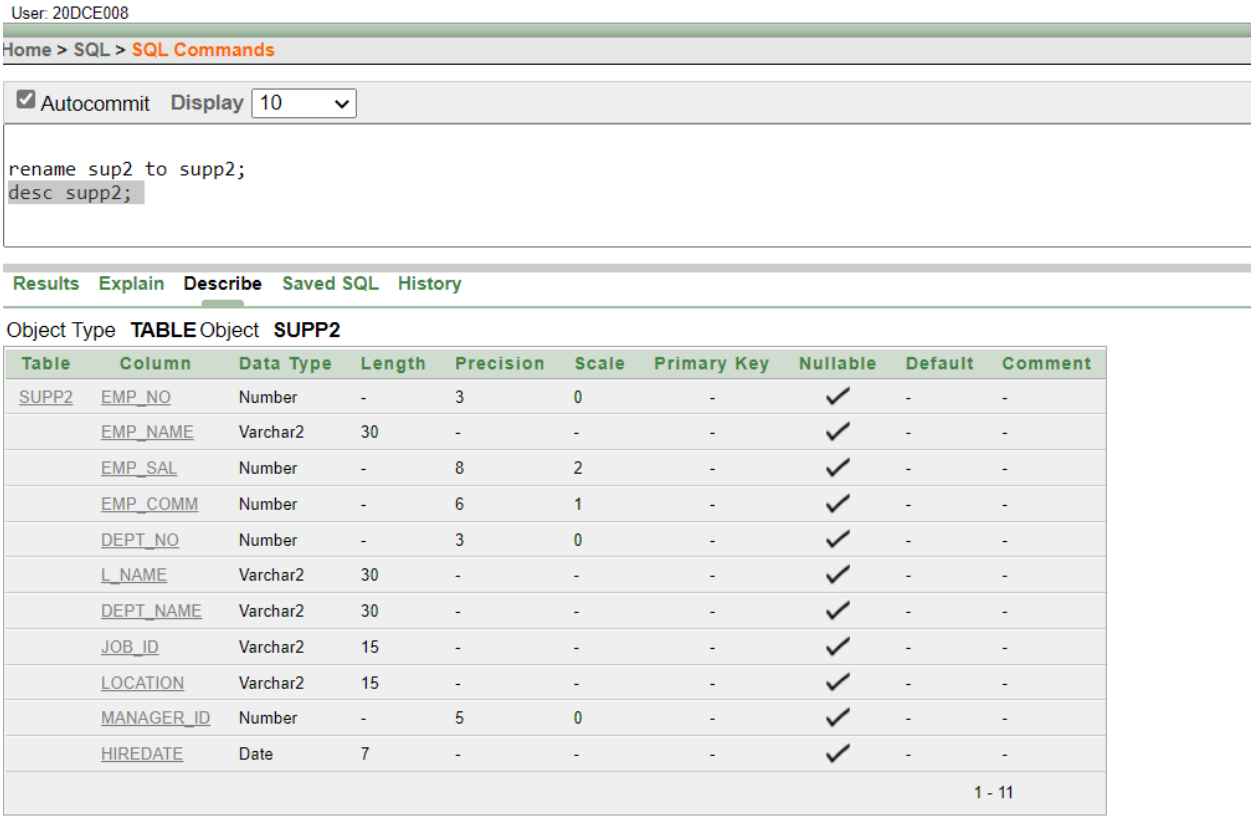
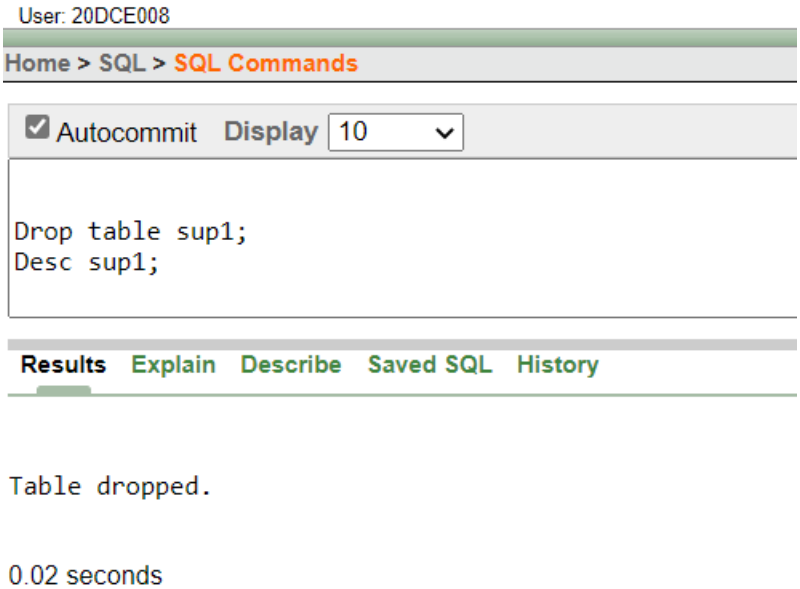


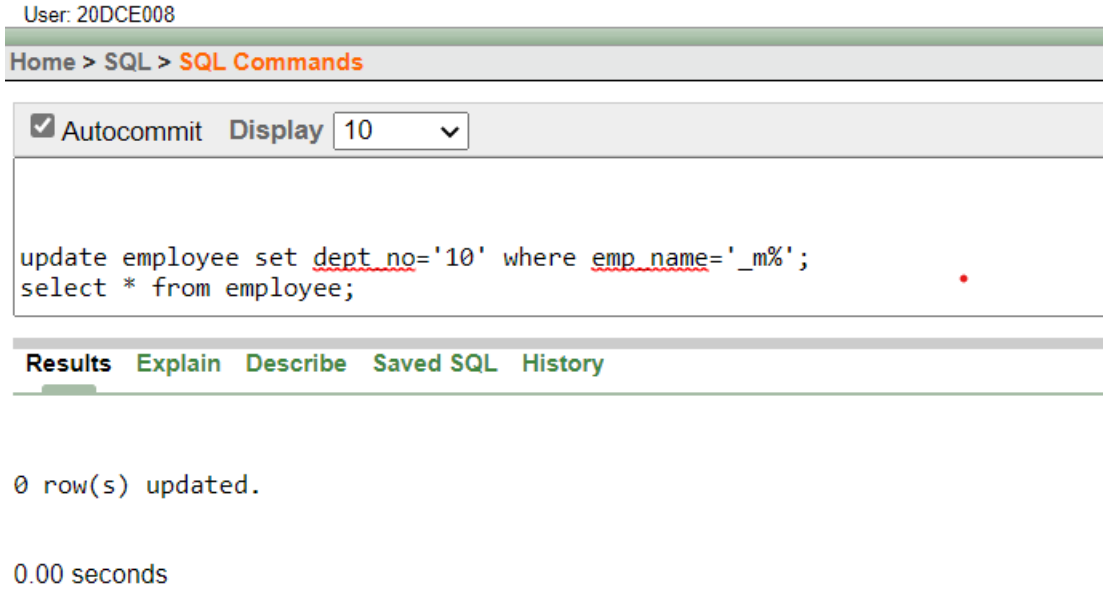
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
SUPP2	EMP_NO	Number	-	3	0	-	✓	-	-
	EMP_NAME	Varchar2	30	-	-	-	✓	-	-
	EMP_SAL	Number	-	8	2	-	✓	-	-
	EMP_COMM	Number	-	6	1	-	✓	-	-
	DEPT_NO	Number	-	3	0	-	✓	-	-
	L_NAME	Varchar2	30	-	-	-	✓	-	-
	DEPT_NAME	Varchar2	30	-	-	-	✓	-	-
	JOB_ID	Varchar2	15	-	-	-	✓	-	-
	LOCATION	Varchar2	15	-	-	-	✓	-	-
	MANAGER_ID	Number	-	5	0	-	✓	-	-
	HIREDATE	Date	7	-	-	-	✓	-	-

1 - 11

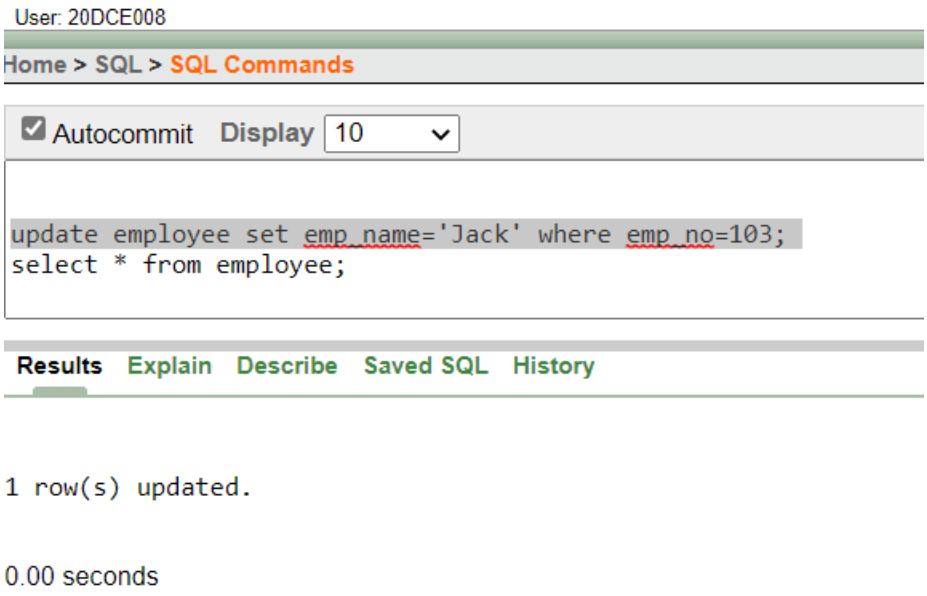
(13) Destroy table sup1 with all the data.

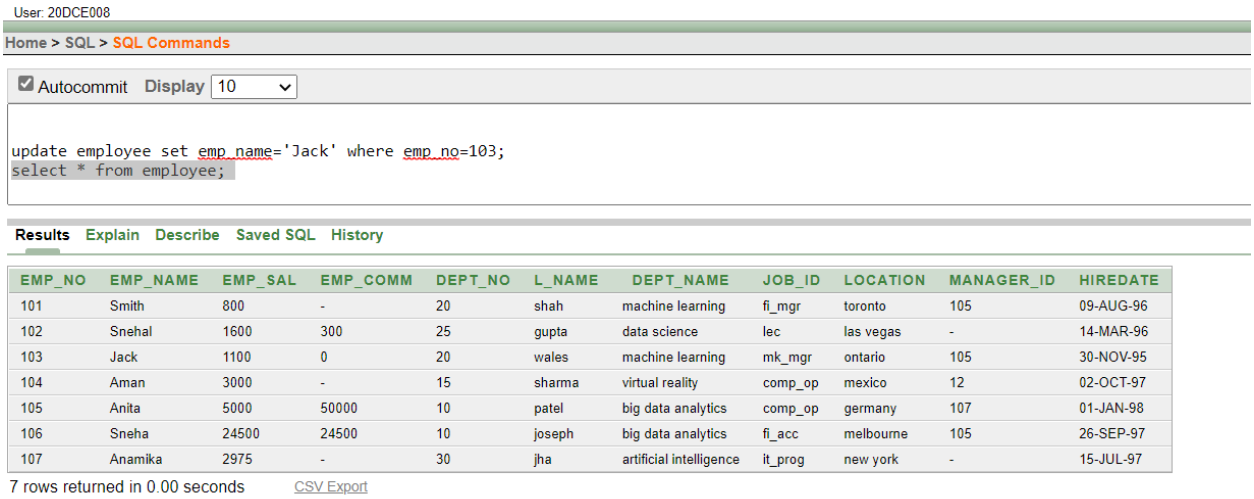


(14) Update the value dept_no to 10 where second character of emp. name is ‘m’.



(15) Update the value of employee name whose employee number is 103.

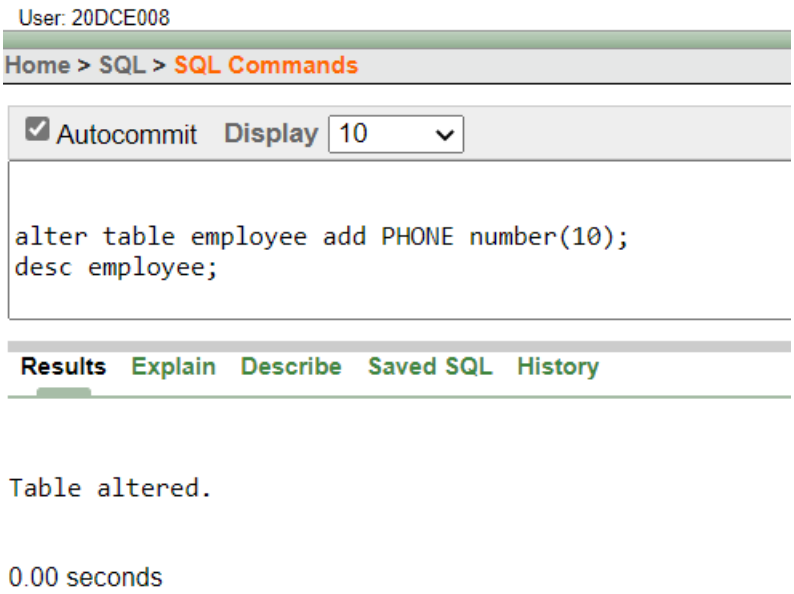




EMP_NO	EMP_NAME	EMP_SAL	EMP_COMM	DEPT_NO	L_NAME	DEPT_NAME	JOB_ID	LOCATION	MANAGER_ID	HIREDATE
101	Smith	800	-	20	shah	machine learning	fi_mgr	toronto	105	09-AUG-96
102	Snehal	1600	300	25	gupta	data science	lec	las vegas	-	14-MAR-96
103	Jack	1100	0	20	wales	machine learning	mk_mgr	ontario	105	30-NOV-95
104	Aman	3000	-	15	sharma	virtual reality	comp_op	mexico	12	02-OCT-97
105	Anita	5000	50000	10	patel	big data analytics	comp_op	germany	107	01-JAN-98
106	Sneha	24500	24500	10	joseph	big data analytics	fi_acc	melbourne	105	26-SEP-97
107	Anamika	2975	-	30	jha	artificial intelligence	it_prog	new york	-	15-JUL-97

7 rows returned in 0.00 seconds CSV Export

(16) Add one column phone to employee with size of column is 10.



User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
alter table employee add PHONE number(10);
desc employee;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **EMPLOYEE**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEE	EMP_NO	Number	-	3	0	-	✓	-	-
	EMP_NAME	Varchar2	30	-	-	-	✓	-	-
	EMP_SAL	Number	-	8	2	-	✓	-	-
	EMP_COMM	Number	-	6	1	-	✓	-	-
	DEPT_NO	Number	-	3	0	-	✓	-	-
	L_NAME	Varchar2	30	-	-	-	✓	-	-
	DEPT_NAME	Varchar2	30	-	-	-	✓	-	-
	JOB_ID	Varchar2	15	-	-	-	✓	-	-
	LOCATION	Varchar2	15	-	-	-	✓	-	-
	MANAGER_ID	Number	-	5	0	-	✓	-	-
	HIREDATE	Date	7	-	-	-	✓	-	-
	PHONE	Number	-	10	0	-	✓	-	-

1 - 12

(17) Modify the column emp_name to hold maximum of 30 characters.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
alter table employee modify emp_name varchar2(30);
desc employee;
```

Results Explain Describe Saved SQL History

```
Table altered.

0.02 seconds
```

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
alter table employee modify emp_name varchar2(30);
desc employee;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **EMPLOYEE**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEE	EMP_NO	Number	-	3	0	-	✓	-	-
	EMP_NAME	Varchar2	30	-	-	-	✓	-	-
	EMP_SAL	Number	-	8	2	-	✓	-	-
	EMP_COMM	Number	-	6	1	-	✓	-	-
	DEPT_NO	Number	-	3	0	-	✓	-	-
	L_NAME	Varchar2	30	-	-	-	✓	-	-
	DEPT_NAME	Varchar2	30	-	-	-	✓	-	-
	JOB_ID	Varchar2	15	-	-	-	✓	-	-
	LOCATION	Varchar2	15	-	-	-	✓	-	-
	MANAGER_ID	Number	-	5	0	-	✓	-	-
	HIREDATE	Date	7	-	-	-	✓	-	-
	PHONE	Number	-	10	0	-	✓	-	-

1 - 12

(18) Count the total no as well as distinct rows in dept_no column with a condition of salary greater than 1000 of employee

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
select count(*) from employee where emp_sal>1000;
select count(distinct " " dept_no) from employee where emp_sal>1000;
```

Results Explain Describe Saved SQL History

COUNT(*)
6

1 rows returned in 0.00 seconds CSV Export

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
select count(*) from employee where emp_sal>1000;
select count(distinct dept_name) from employee where emp_sal>1000;
```

Results Explain Describe Saved SQL History

COUNT(DISTINCTDEPT_NAME)
5

1 rows returned in 0.00 seconds CSV Export

(19) Display the detail of all employees in ascending order, descending order of their name and no.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
select * from employee order by emp_name asc,emp_no desc
```

Results Explain Describe Saved SQL History

EMP_NO	EMP_NAME	EMP_SAL	EMP_COMM	DEPT_NO	L_NAME	DEPT_NAME	JOB_ID	LOCATION	MANAGER_ID	HIREDATE	PHONE
104	Aman	3000	-	15	sharma	virtual reality	comp_op	mexico	12	02-OCT-97	-
107	Anamika	2975	-	30	jha	artificial intelligence	it_prog	new york	-	15-JUL-97	-
105	Anita	5000	50000	10	patel	big data analytics	comp_op	germany	107	01-JAN-98	-
103	Jack	1100	0	20	wales	machine learning	mk_mgr	ontario	105	30-NOV-95	-
101	Smith	800	-	20	shah	machine learning	fi_mgr	toronto	105	09-AUG-96	-
106	Sneha	24500	24500	10	joseph	big data analytics	fi_acc	melbourne	105	26-SEP-97	-
102	Snehal	1600	300	25	gupta	data science	lec	las vegas	-	14-MAR-96	-

7 rows returned in 0.00 seconds CSV Export

(20) Display the dept_no in ascending order and accordingly display emp_comm in descending
order.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
select dept_no,emp_comm from employee order by emp_comm desc,dept_no asc;
select dept_no,emp_comm from employee order by emp_comm desc;
```

Results Explain Describe Saved SQL History

DEPT_NO	EMP_COMM
15	-
20	-
30	-
10	50000
10	24500
25	300
20	0

7 rows returned in 0.00 seconds CSV Export

21) Update the value of emp_comm to 500 where dept_no is 20.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

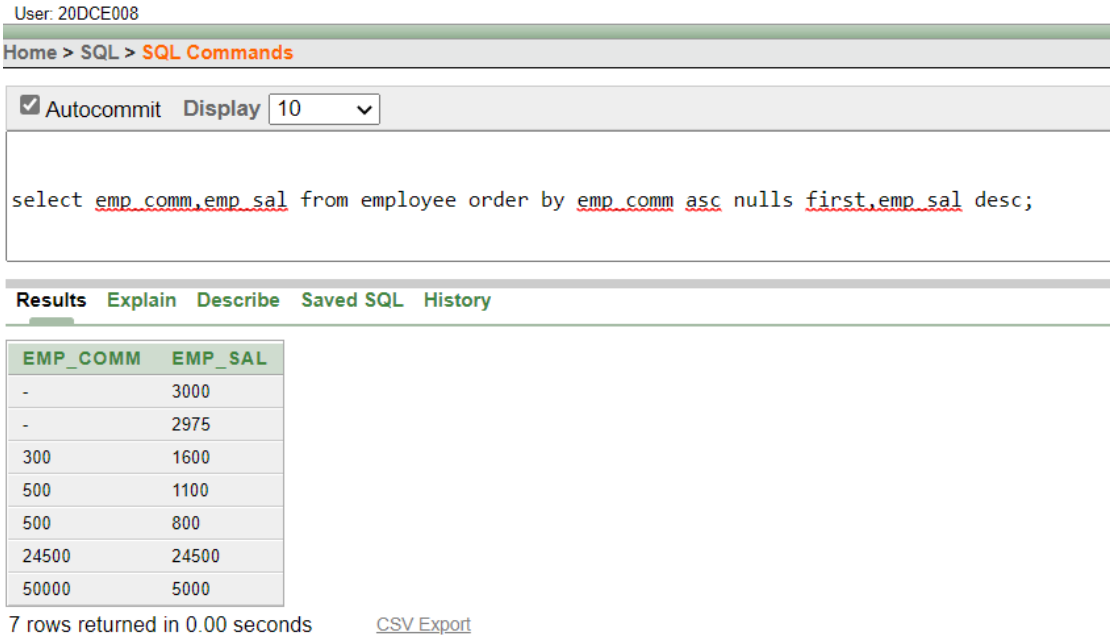
```
update employee set emp_comm=500 where dept_no=20 ;
select * from employee;
```

Results Explain Describe Saved SQL History

EMP_NO	EMP_NAME	EMP_SAL	EMP_COMM	DEPT_NO	L_NAME	DEPT_NAME	JOB_ID	LOCATION	MANAGER_ID	HIREDATE	PHONE
101	Smith	800	500	20	shah	machine learning	fi_mgr	toronto	105	09-AUG-96	-
102	Snehal	1600	300	25	gupta	data science	lec	las vegas	-	14-MAR-96	-
103	Jack	1100	500	20	wales	machine learning	mk_mgr	ontario	105	30-NOV-95	-
104	Aman	3000	-	15	sharma	virtual reality	comp_op	mexico	12	02-OCT-97	-
105	Anita	5000	50000	10	patel	big data analytics	comp_op	germany	107	01-JAN-98	-
106	Sneha	24500	24500	10	joseph	big data analytics	fi_acc	melbourne	105	26-SEP-97	-
107	Anamika	2975	-	30	jha	artificial intelligence	it_prog	new york	-	15-JUL-97	-

7 rows returned in 0.00 seconds CSV Export

(22) Display the emp_comm in ascending order with null value first and accordingly sort employee salary in descending order.



User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

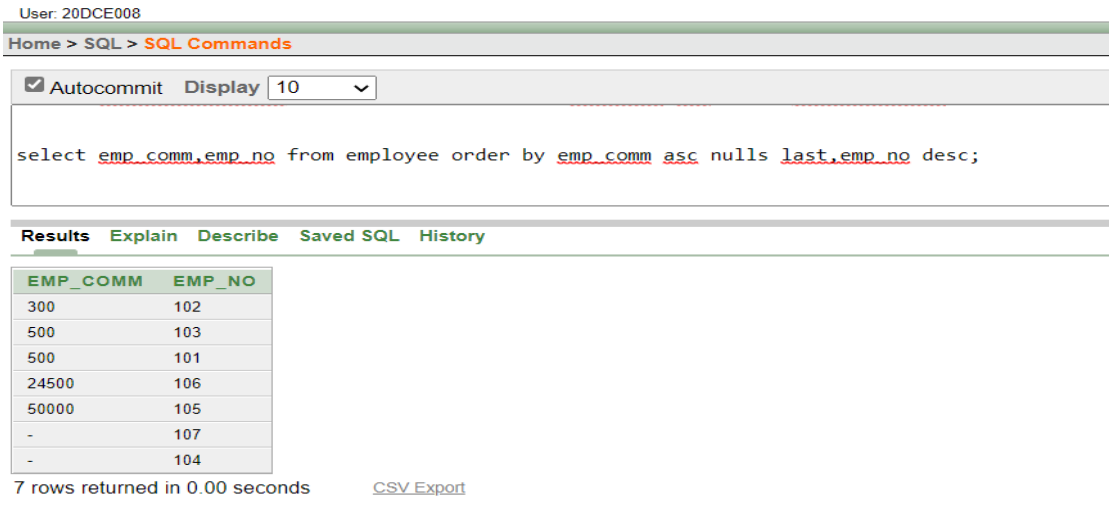
```
select emp_comm,emp_sal from employee order by emp_comm asc nulls first,emp_sal desc;
```

Results Explain Describe Saved SQL History

EMP_COMM	EMP_SAL
-	3000
-	2975
300	1600
500	1100
500	800
24500	24500
50000	5000

7 rows returned in 0.00 seconds CSV Export

(23) Display the emp_comm in ascending order with null value last and accordingly sort emp_no in descending order.



User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
select emp_comm,emp_no from employee order by emp_comm asc nulls last,emp_no desc;
```

Results Explain Describe Saved SQL History

EMP_COMM	EMP_NO
300	102
500	103
500	101
24500	106
50000	105
-	107
-	104

7 rows returned in 0.00 seconds CSV Export

Conclusion:

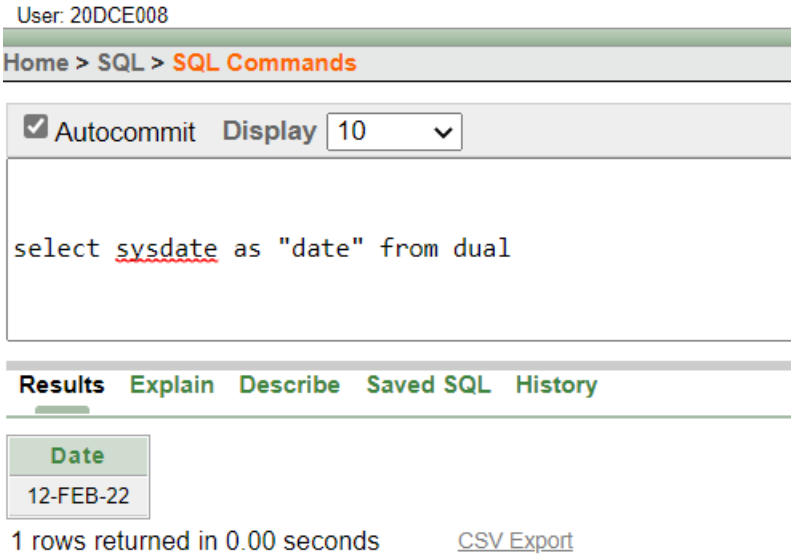
In this practical we learnt about various data manipulation commands, aggregate functions and sorting concept on all created tables.

Practical-6

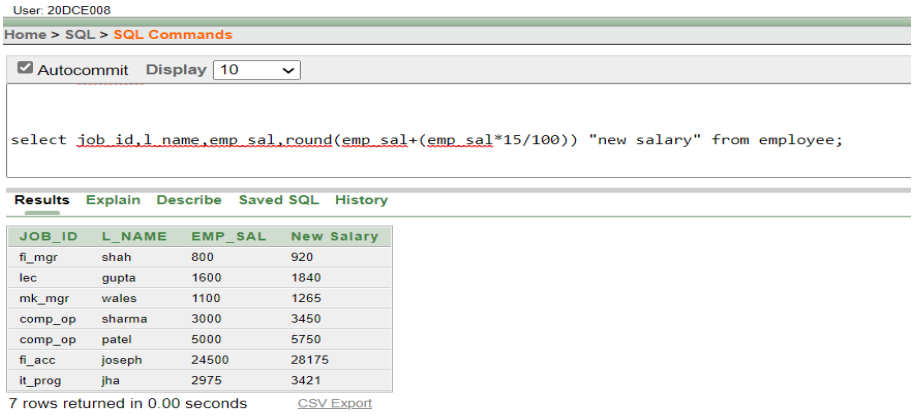
Aim: To study Single-row functions.

Program:

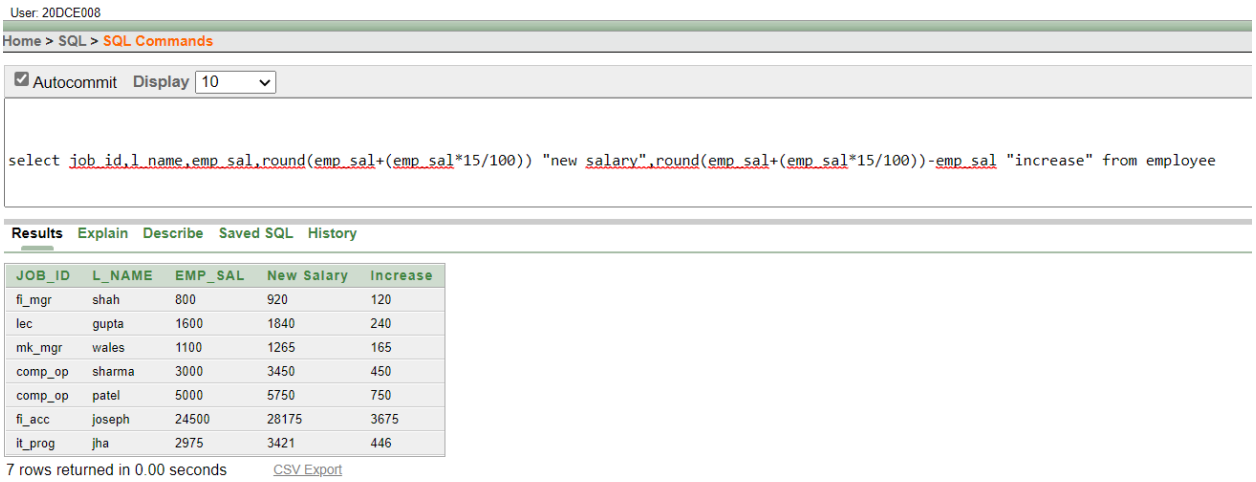
1. Write a query to display the current date. Label the column Date.



2. For each employee, display the employee number, salary, and salary increased by 15% and expressed as a whole number. Label the column New Salary.



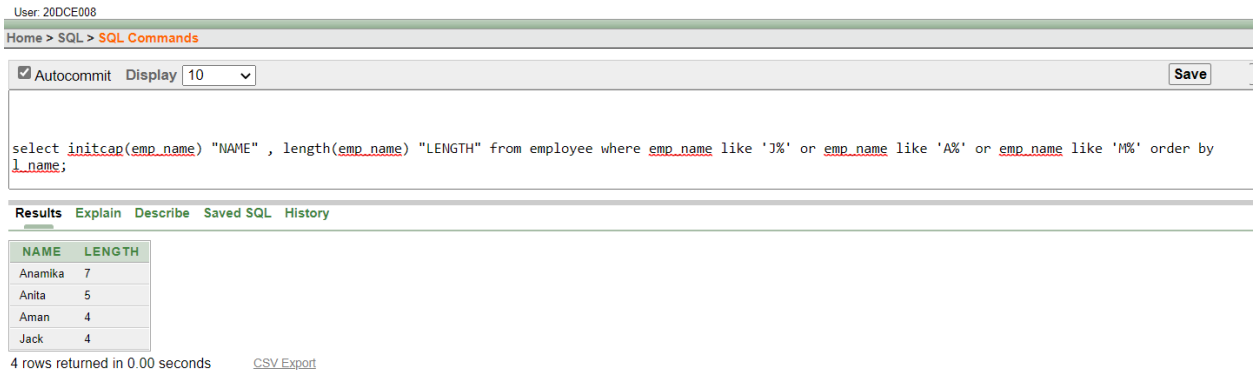
3. Modify your query no (2) to add a column that subtracts the old salary from the new salary. Label the column Increase.



JOB_ID	L_NAME	EMP_SAL	New Salary	Increase
fi_mgr	shah	800	920	120
lec	gupta	1600	1840	240
mk_mgr	wales	1100	1265	165
comp_op	sharma	3000	3450	450
comp_op	patel	5000	5750	750
fi_acc	joseph	24500	28175	3675
it_prog	jha	2975	3421	446

7 rows returned in 0.00 seconds CSV Export

4. Write a query that displays the employee's names with the first letter capitalized and all other letters lowercase, and the length of the names, for all employees whose name starts with J, A, or M. Give each column an appropriate label. Sort the results by the employees' last names.



NAME	LENGTH
Anamika	7
Anita	5
Aman	4
Jack	4

4 rows returned in 0.00 seconds CSV Export

5. Write a query that produces the following for each employee:
<employee last name> earns <salary> monthly

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

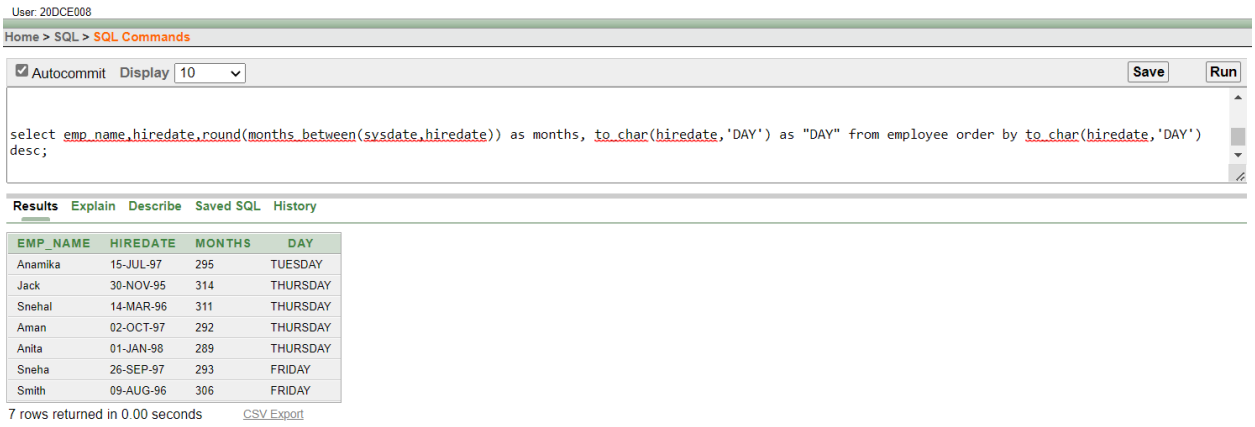
```
select emp_name || ' EARNS ' || emp_sal || ' MONTHLY ' as "SENTENCE" from employee
```

Results Explain Describe Saved SQL History

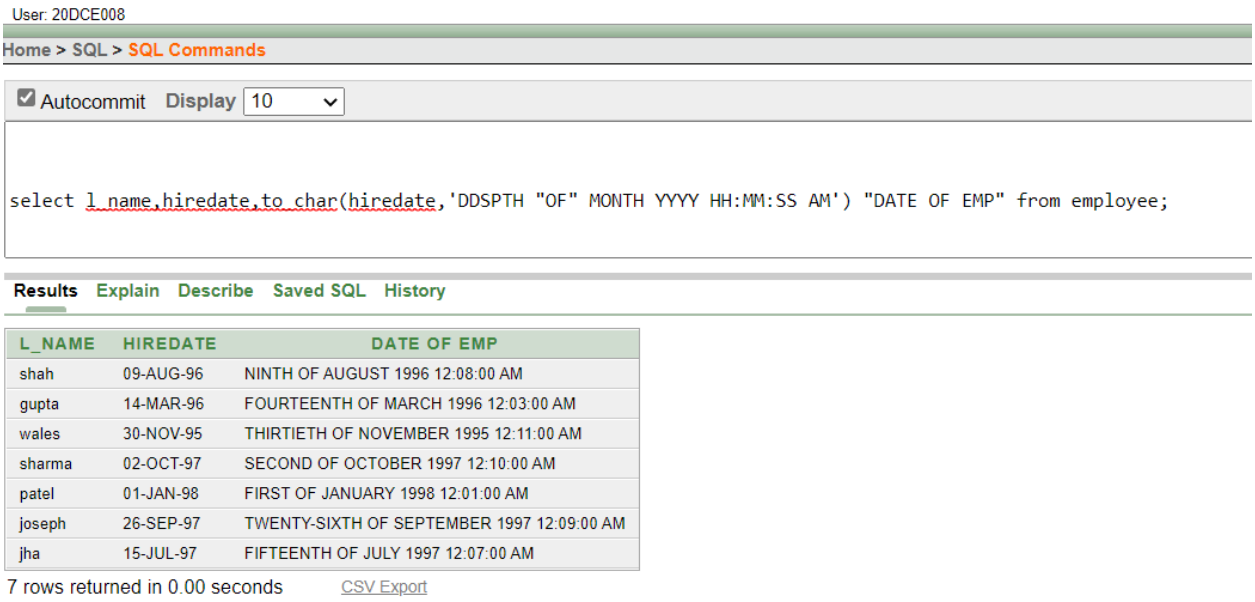
SENTENCE
Smith EARNS 800 MONTHLY
Snehal EARNS 1600 MONTHLY
Jack EARNS 1100 MONTHLY
Aman EARNS 3000 MONTHLY
Anita EARNS 5000 MONTHLY
Sneha EARNS 24500 MONTHLY
Anamika EARNS 2975 MONTHLY

7 rows returned in 0.00 seconds CSV Export

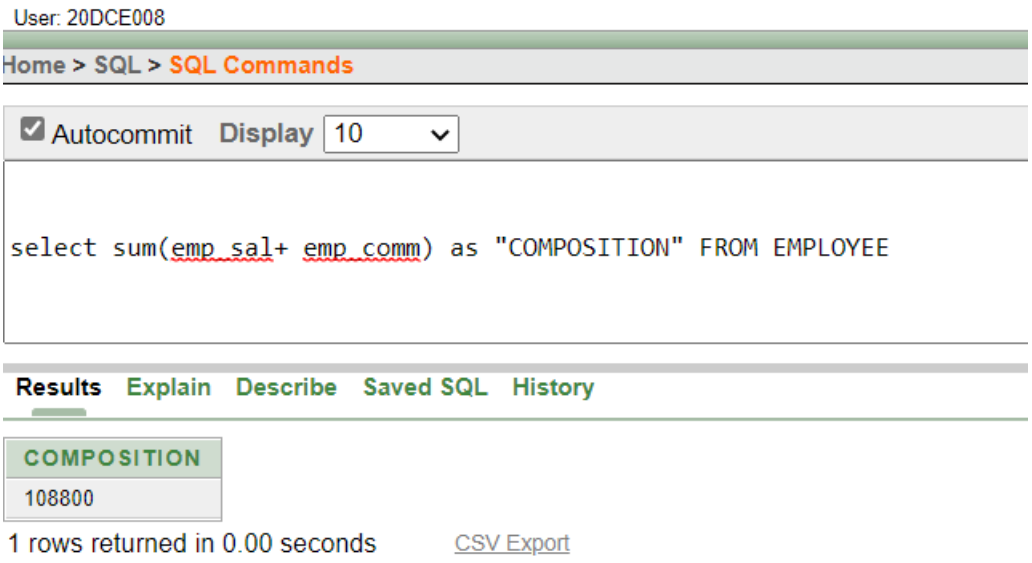
6. Display the name, date, number of months employed and day of the week on which the employee has started. Order the results by the day of the week starting with Monday.



7. Display the date of emp in a format that appears as Seventh of June 1994 12:00:00 AM.



8. Write a query to calculate the annual compensation of all employees (sal +comm.).



Conclusion:

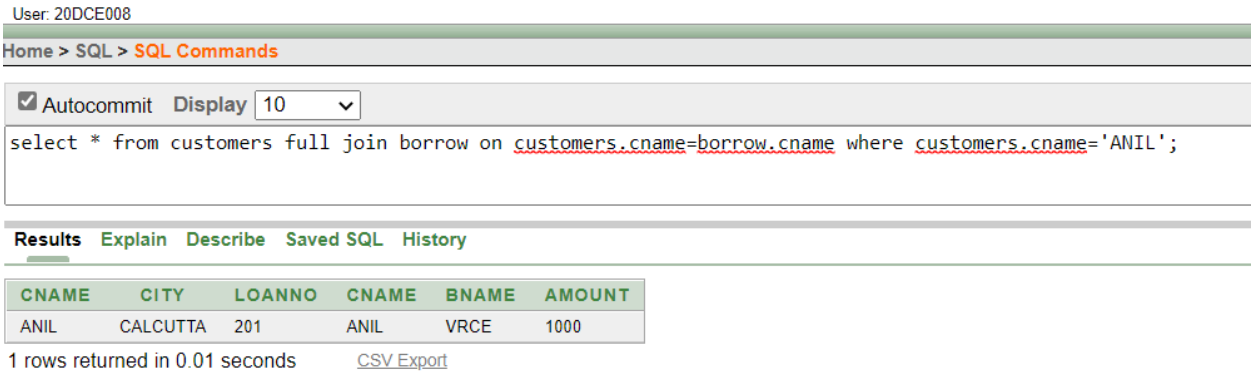
We learnt some single row function in this practical.

Practical-7

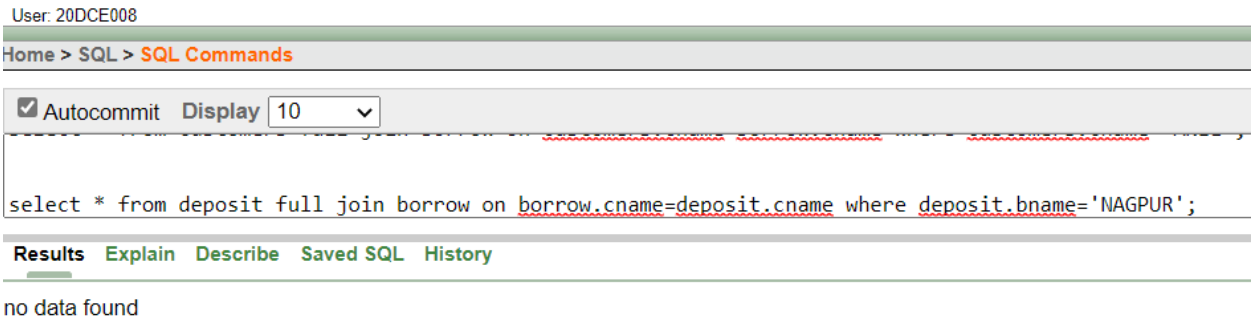
Aim: Displaying data from Multiple Tables (join)

Program:

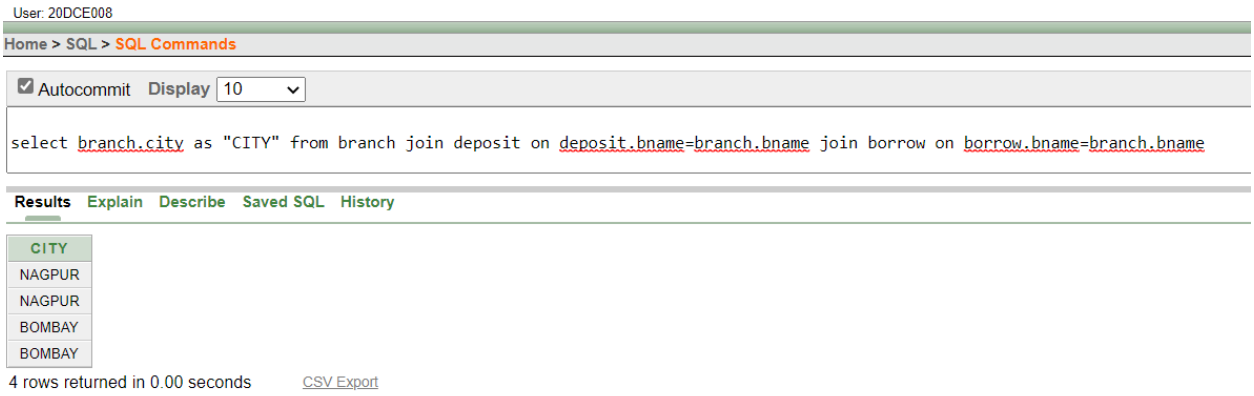
1. Give details of customers ANIL.



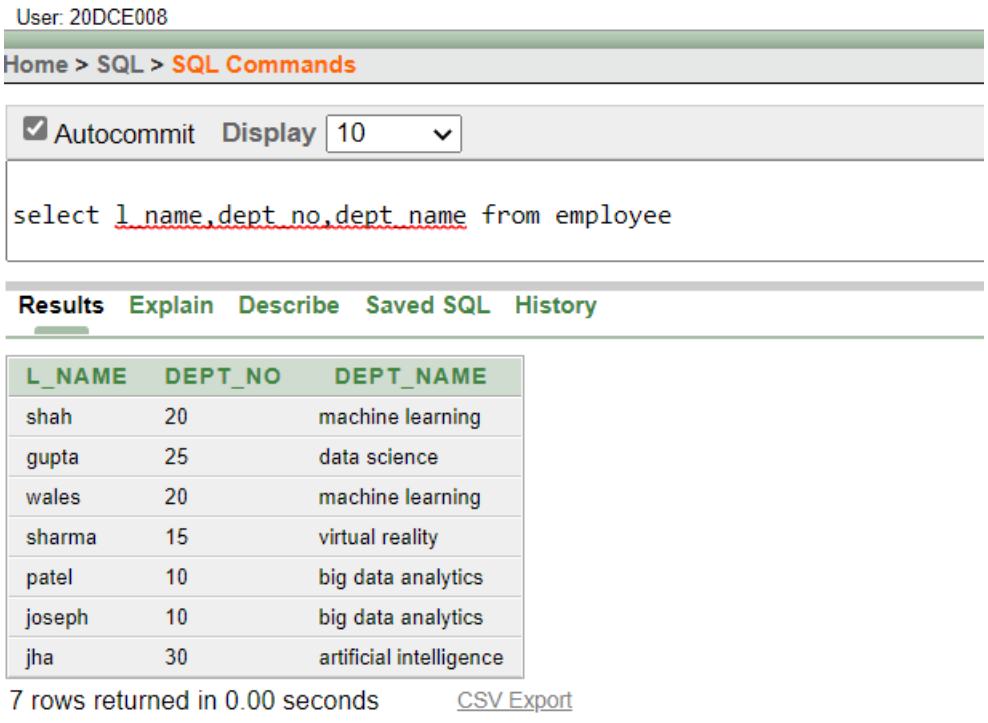
2. Give name of customer who are borrowers and depositors and having living city Nagpur



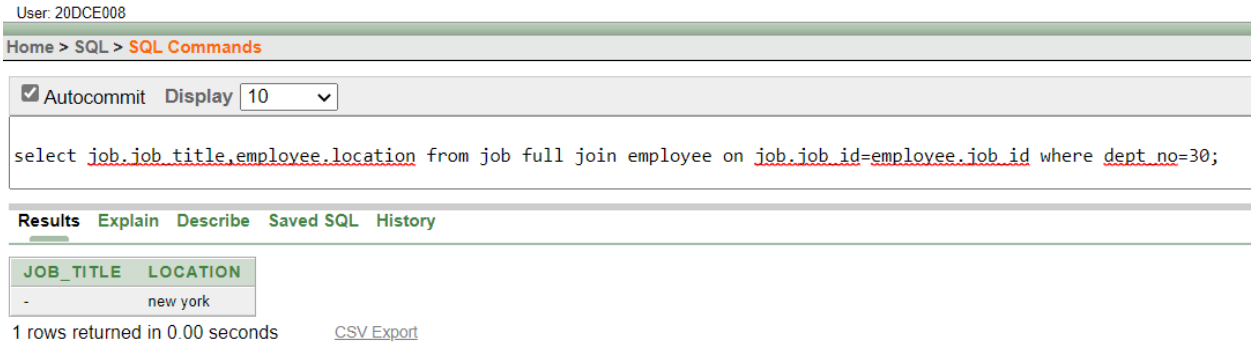
3. Give city as their city name of customers having same living branch.



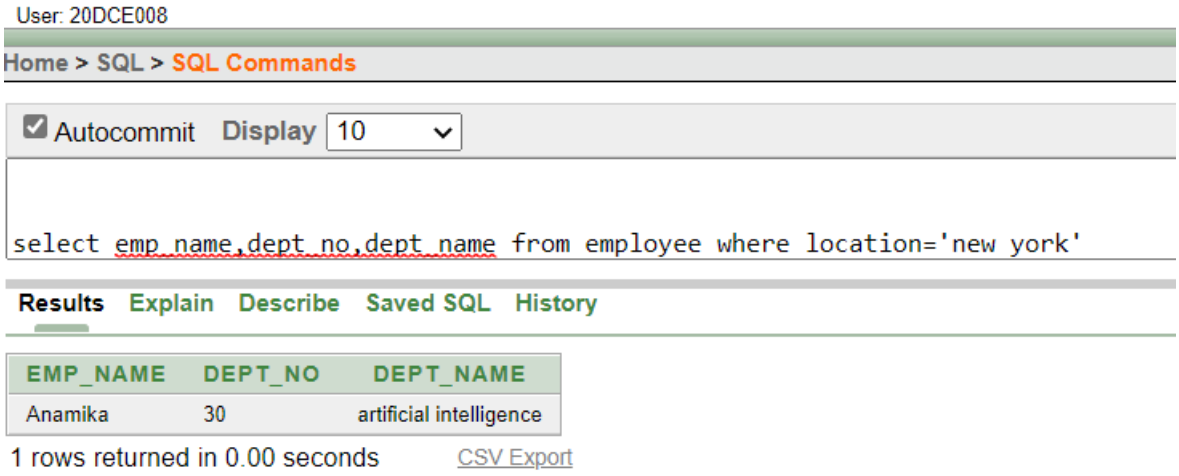
4. Write a query to display the last name, department number, and department name for all employees.



5. Create a unique listing of all jobs that are in department 30. Include the location of the department in the output



6. Write a query to display the employee name, department number, and department name for all employees who work in NEW YORK.



7. Display the employee last name and employee number along with their manager's last name and manager number. Label the columns Employee, Emp#, Manager, and Mgr#, respectively.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
select l_name "Employee",emp_no "Emp#", manager_id "Mgr#" from employee;
```

Results Explain Describe Saved SQL History

Employee	Emp#	Mgr#
shah	101	105
gupta	102	-
wales	103	105
sharma	104	12
patel	105	107
joseph	106	105
jha	107	-

7 rows returned in 0.00 seconds CSV Export

8. Create a query to display the name and hire date of any employee hired after employee “smith”.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
select emp_name,hiredate from employee where hiredate>'09-AUG-96'
```

Results Explain Describe Saved SQL History

EMP_NAME	HIREDATE
Aman	02-OCT-97
Anita	01-JAN-98
Sneha	26-SEP-97
Anamika	15-JUL-97

4 rows returned in 0.00 seconds CSV Export

Conclusion:

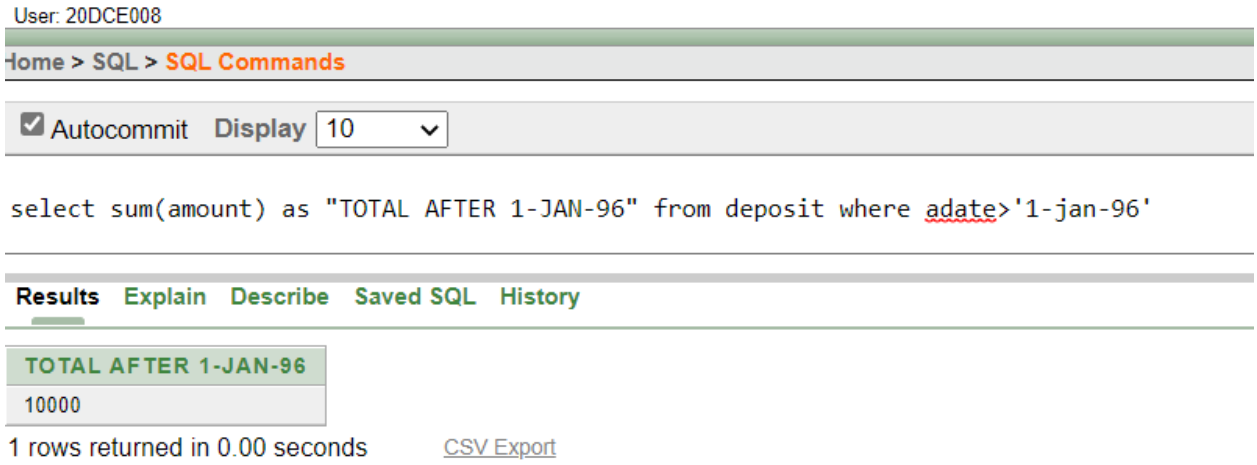
We learnt some operations that can be performed on table using different types of join query.

Practical-8

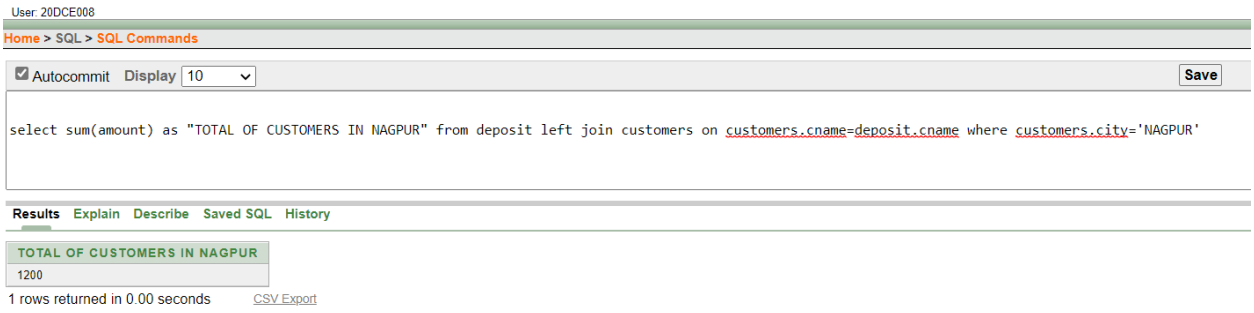
Aim: To apply the concept of Aggregating Data using Group functions.

Program:

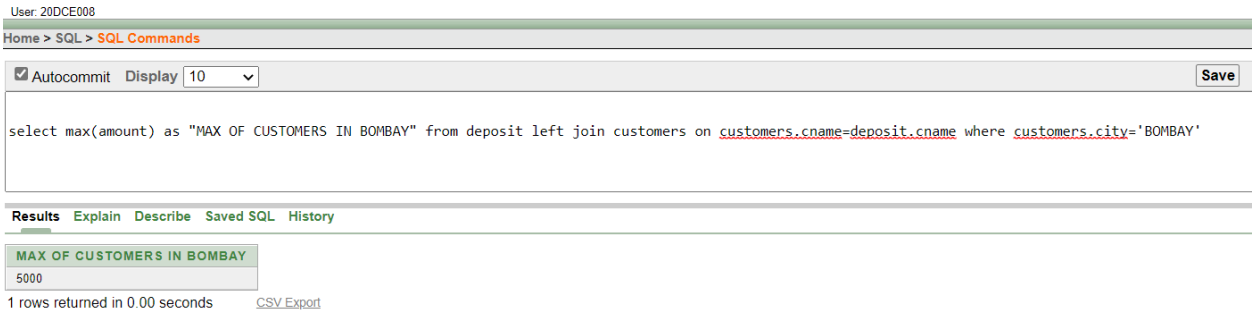
(1) List total deposit of customer having account date after 1-jan-96.



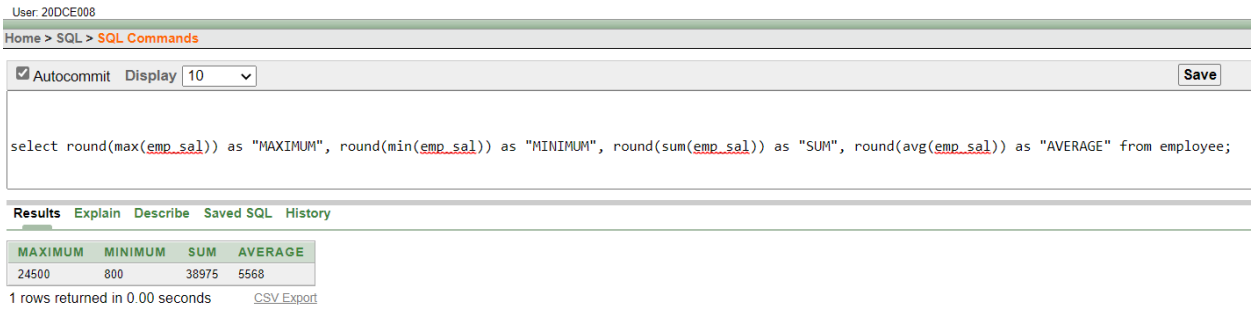
(2) List total deposit of customers living in city Nagpur.



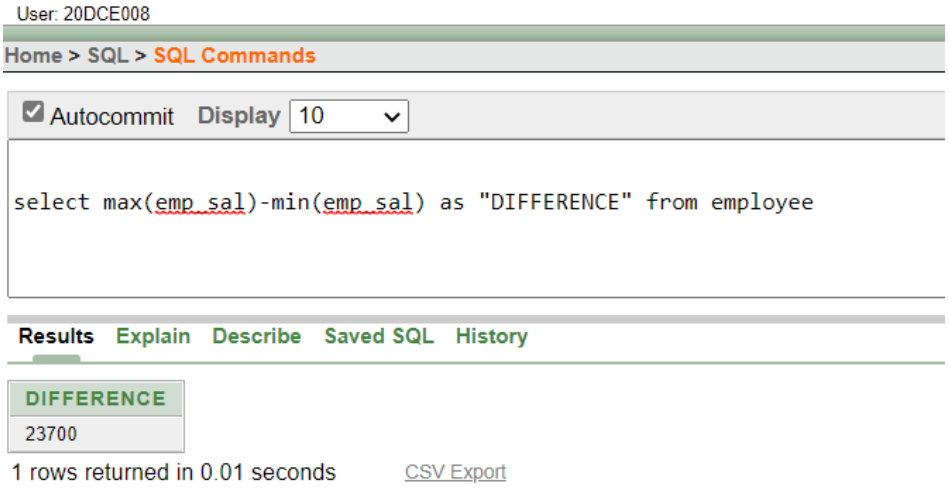
(3) List maximum deposit of customers living in bombay.



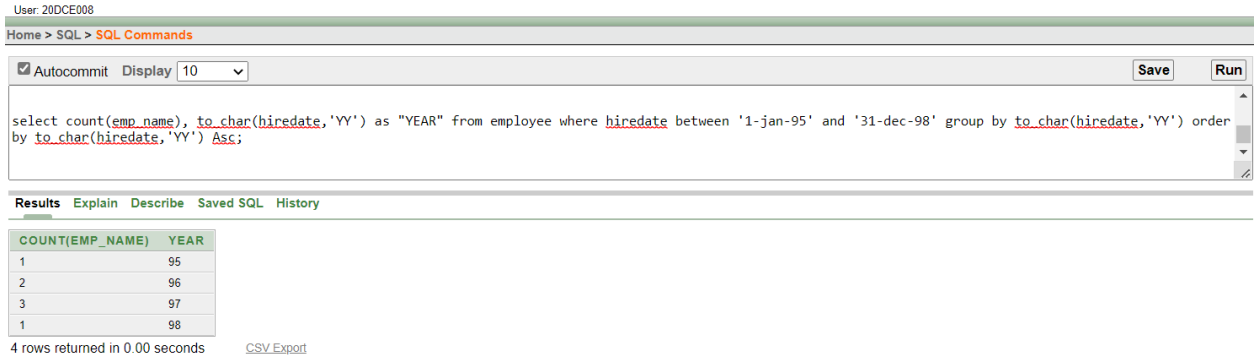
(4) Display the highest, lowest, sum, and average salary of all employees. Label the columns Maximum, Minimum, Sum, and Average, respectively. Round your results to the nearest whole number.



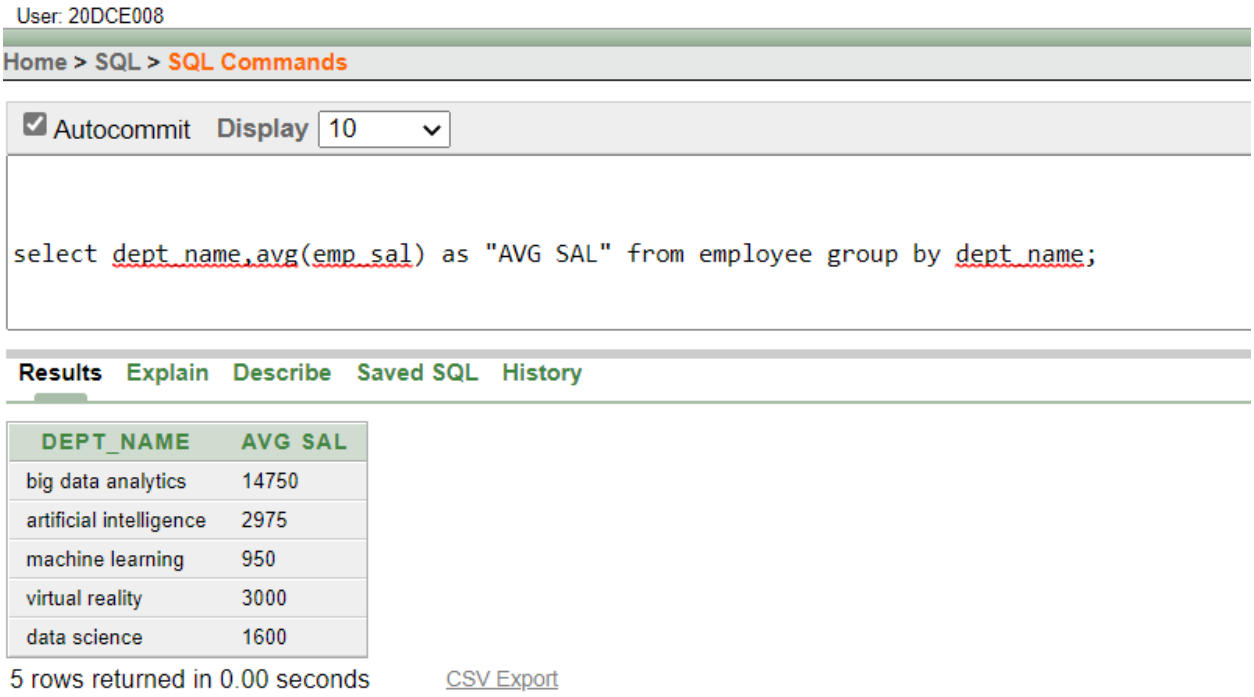
(5) Write a query that displays the difference between the highest and lowest salaries. Label the column DIFFERENCE.



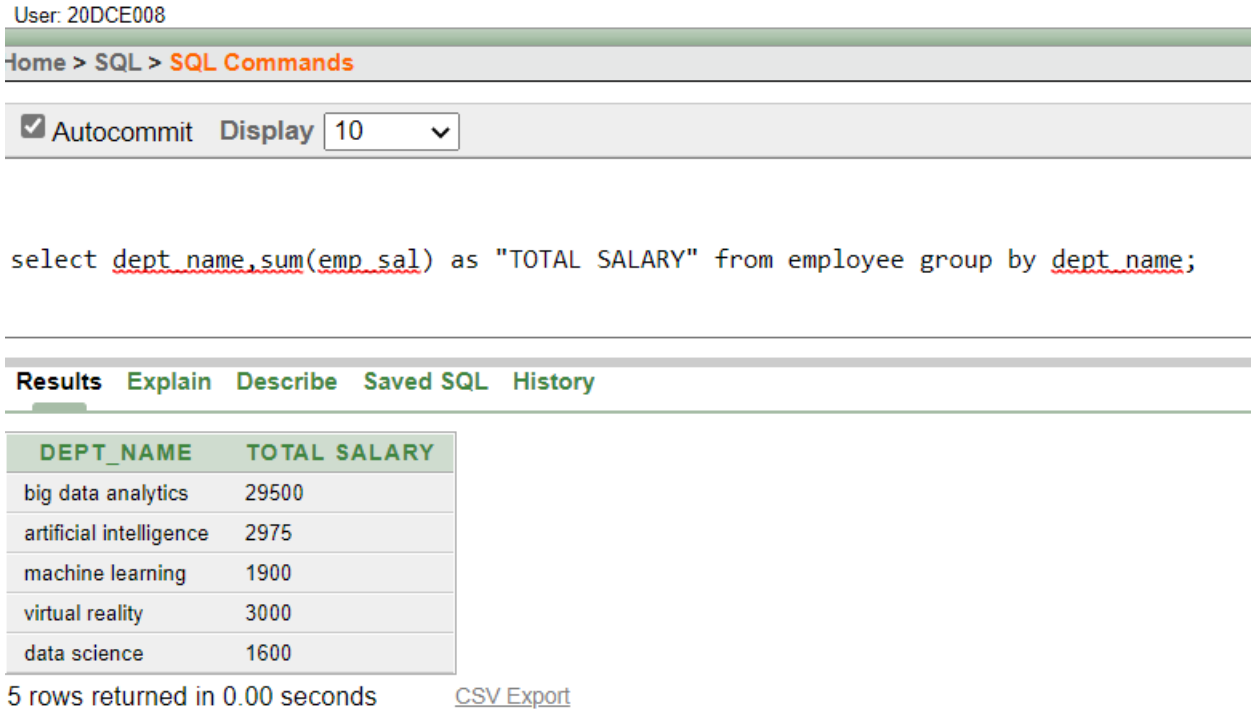
(6) Create a query that will display the total number of employees and, of that total, the number of employees hired in 1995, 1996, 1997, and 1998



(7) Find the average salaries for each department without displaying the respective department numbers.



(8) Write a query to display the total salary being paid to each job title, within each department.


User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

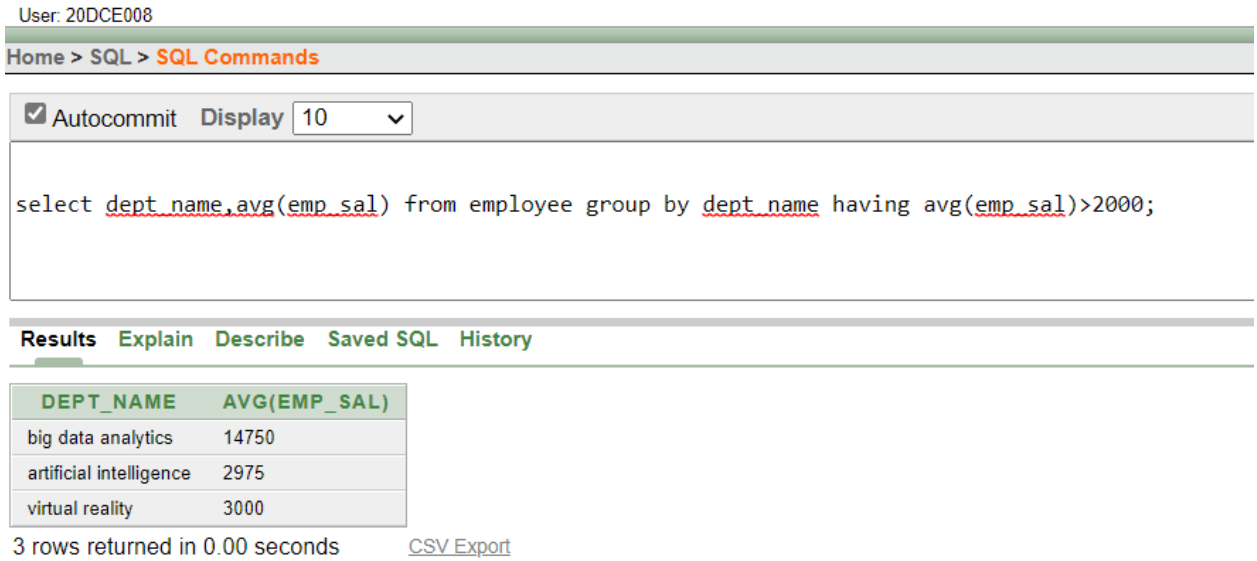
```
select dept_name,sum(emp_sal) as "TOTAL SALARY" from employee group by dept_name;
```

Results Explain Describe Saved SQL History

DEPT_NAME	TOTAL SALARY
big data analytics	29500
artificial intelligence	2975
machine learning	1900
virtual reality	3000
data science	1600

5 rows returned in 0.00 seconds CSV Export

(9) Find the average salaries > 2000 for each department without displaying the respective department numbers.


User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

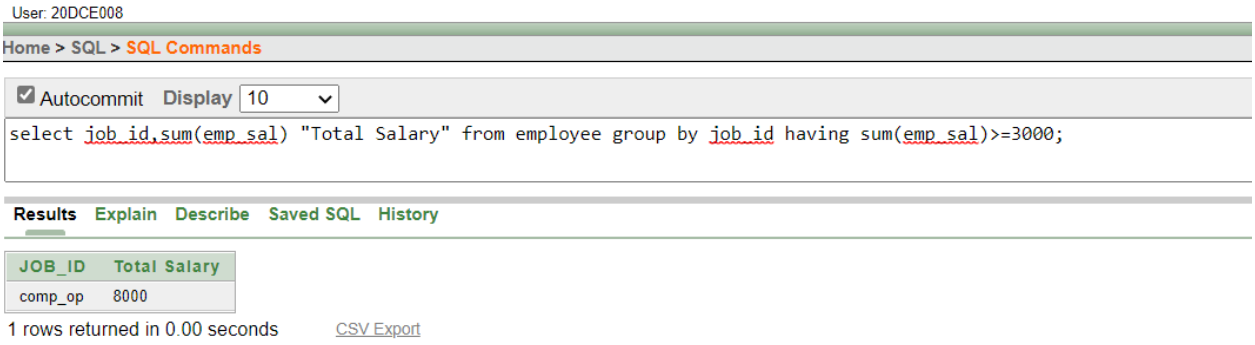
```
select dept_name,avg(emp_sal) from employee group by dept_name having avg(emp_sal)>2000;
```

Results Explain Describe Saved SQL History

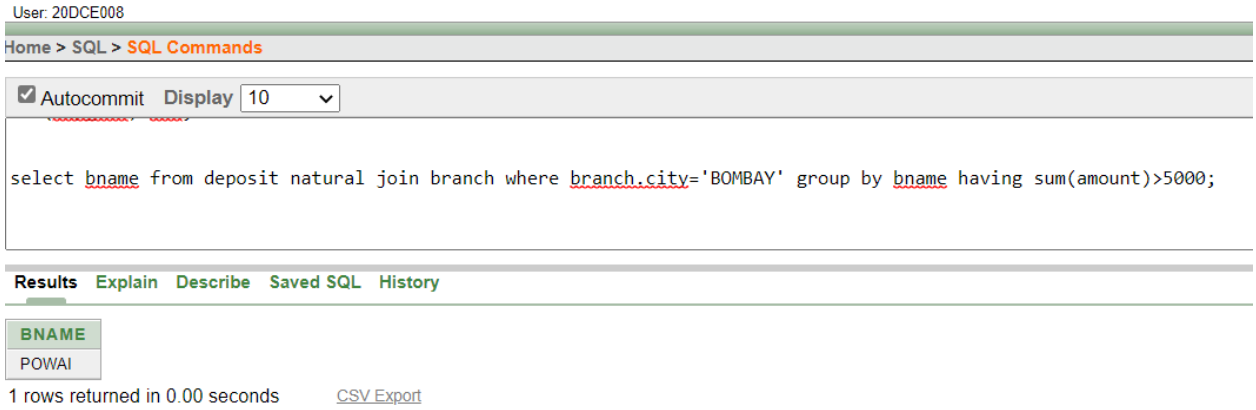
DEPT_NAME	AVG(EMP_SAL)
big data analytics	14750
artificial intelligence	2975
virtual reality	3000

3 rows returned in 0.00 seconds CSV Export

(10) Display the job and total salary for each job with a total salary amount exceeding 3000 and sorts the list by the total salary.



(11) List the branches having sum of deposit more than 5000 and located in city bombay.



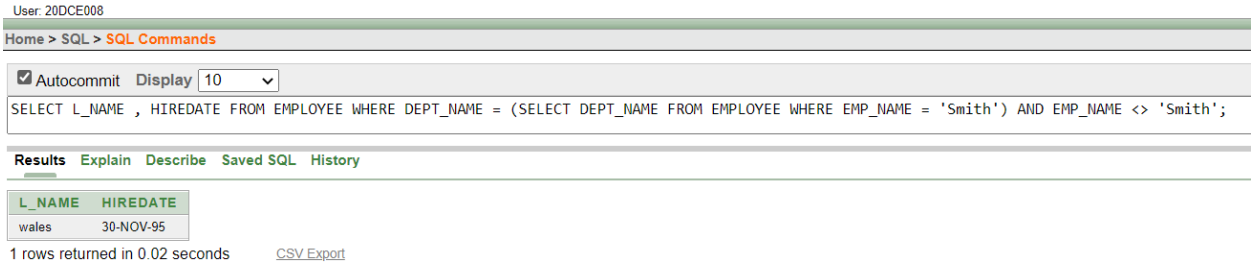
Conclusion: We learnt how to apply the concept of aggregating data using group function.

Practical-9

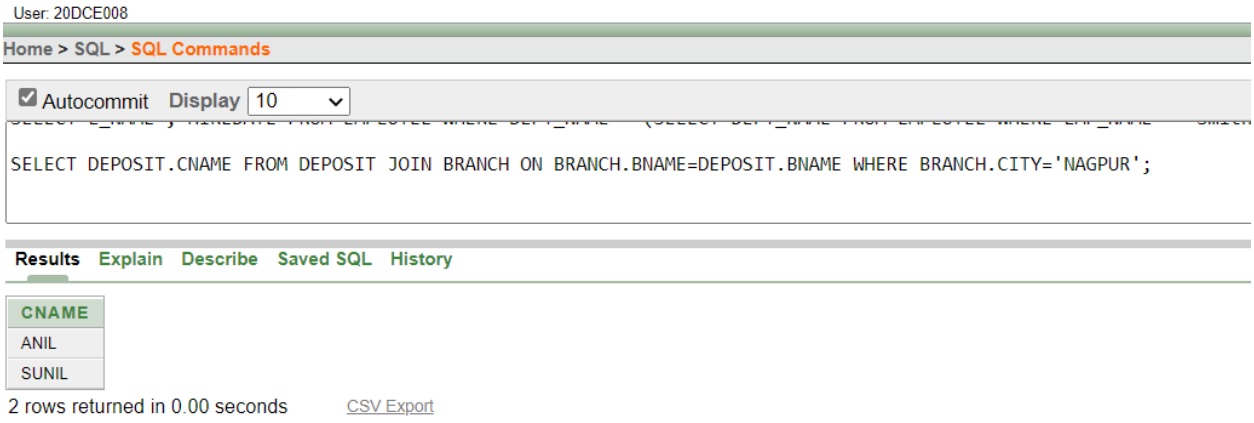
Aim: To solve queries using the concept of sub query.

Program:

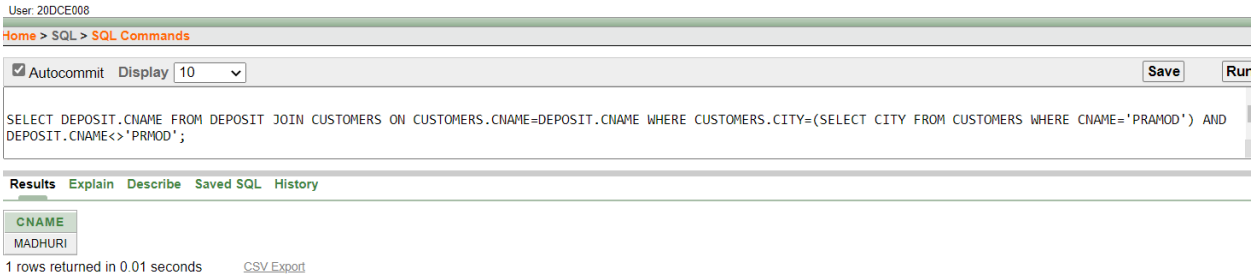
(1) Write a query to display the last name and hire date of any employee in the same department as smith. Exclude smith



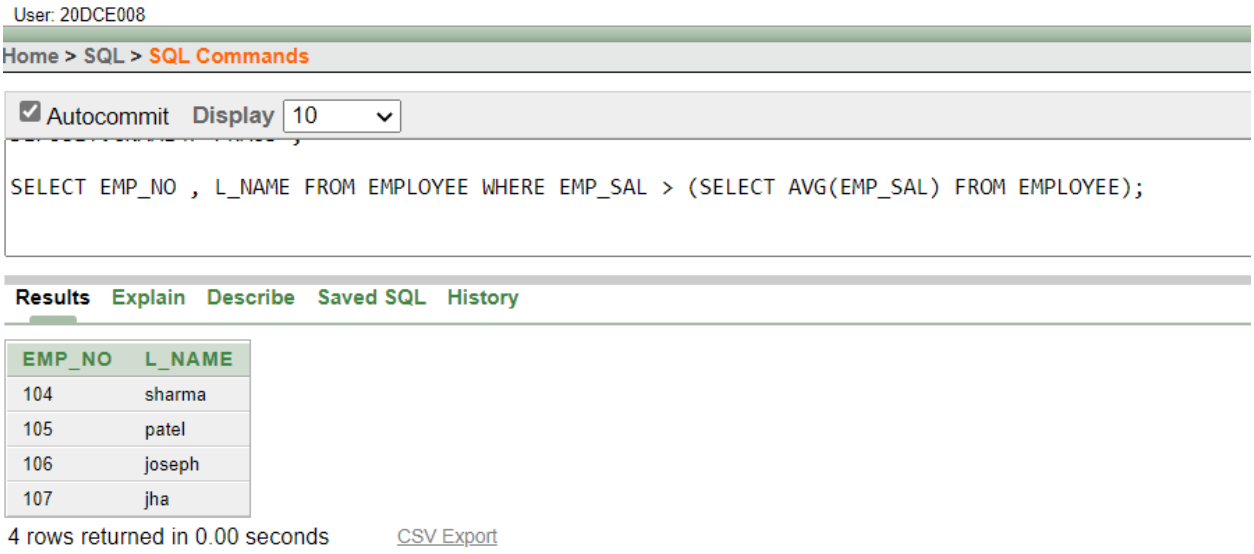
(2) Give name of customers who are depositors having same branch city of mr. sunil.



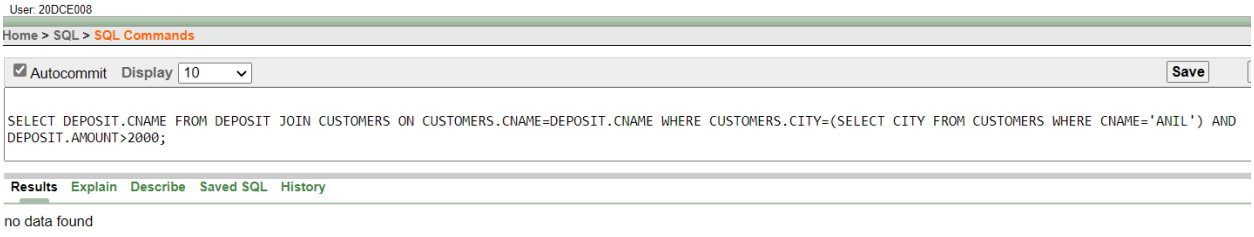
(3) Give deposit details and loan details of customer in same city where pramod is living.



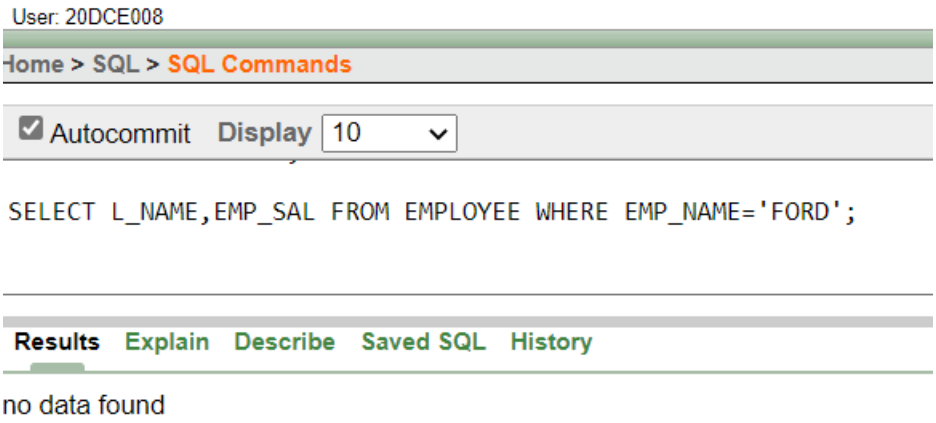
(4) Create a query to display the employee numbers and last names of all employees who earn more than the average salary. Sort the results in ascending order of salary.



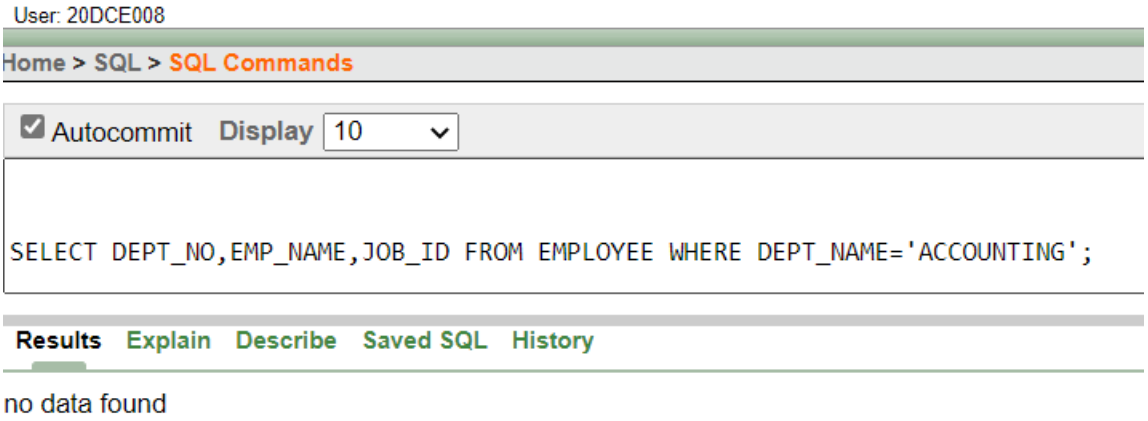
(5) Give names of depositors having same living city as mr. anil and having deposit amount greater than 2000



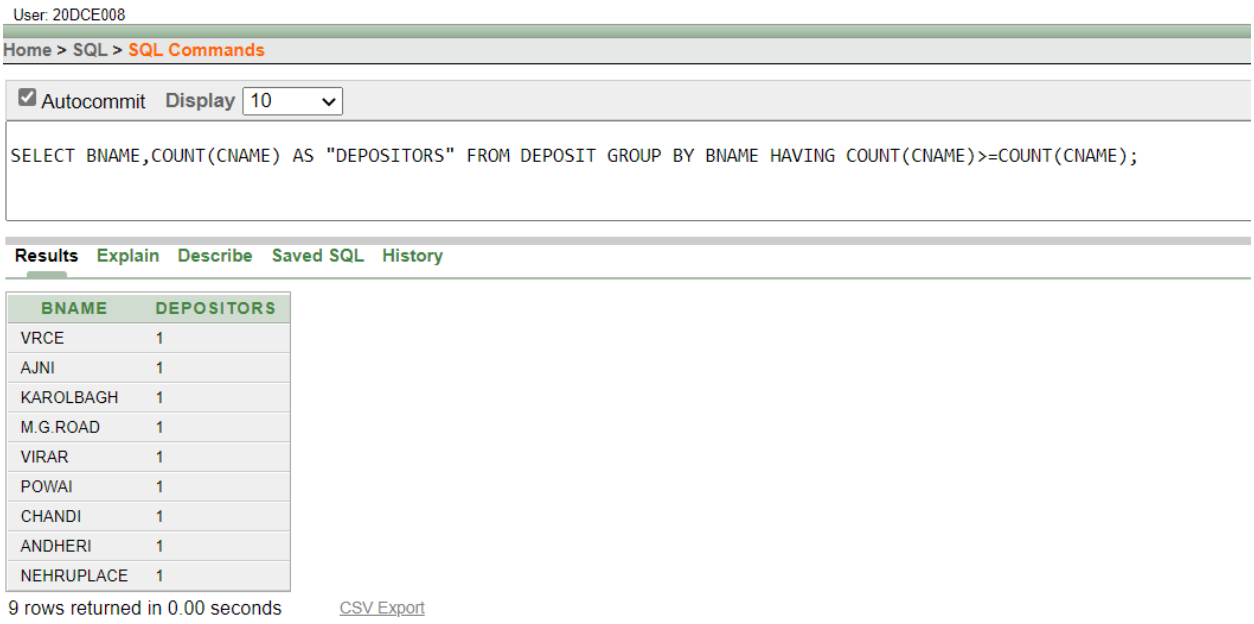
(6) Display the last name and salary of every employee who reports to ford.



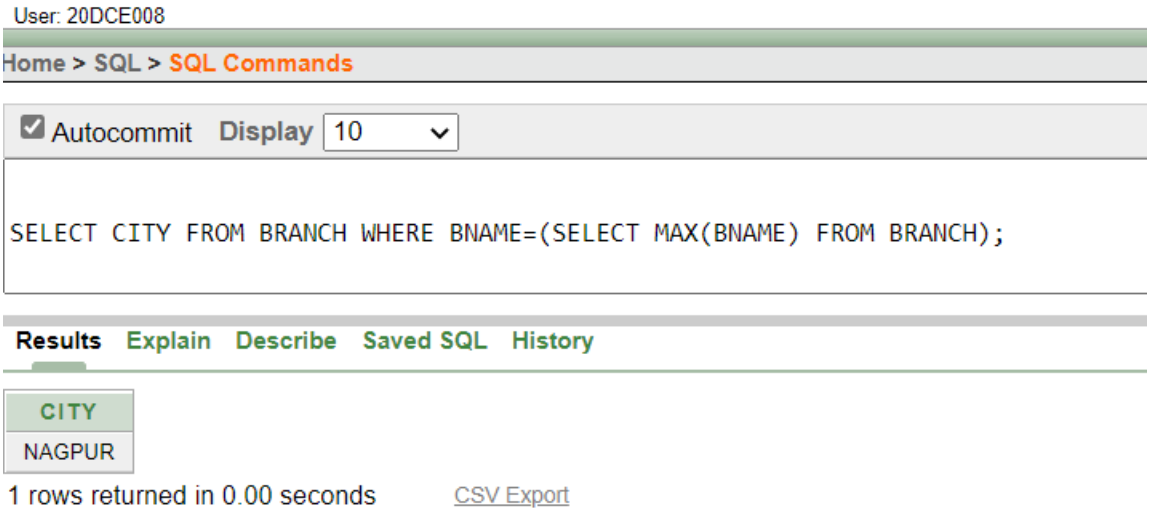
(7) Display the department number, name, and job for every employee in the Accounting department.



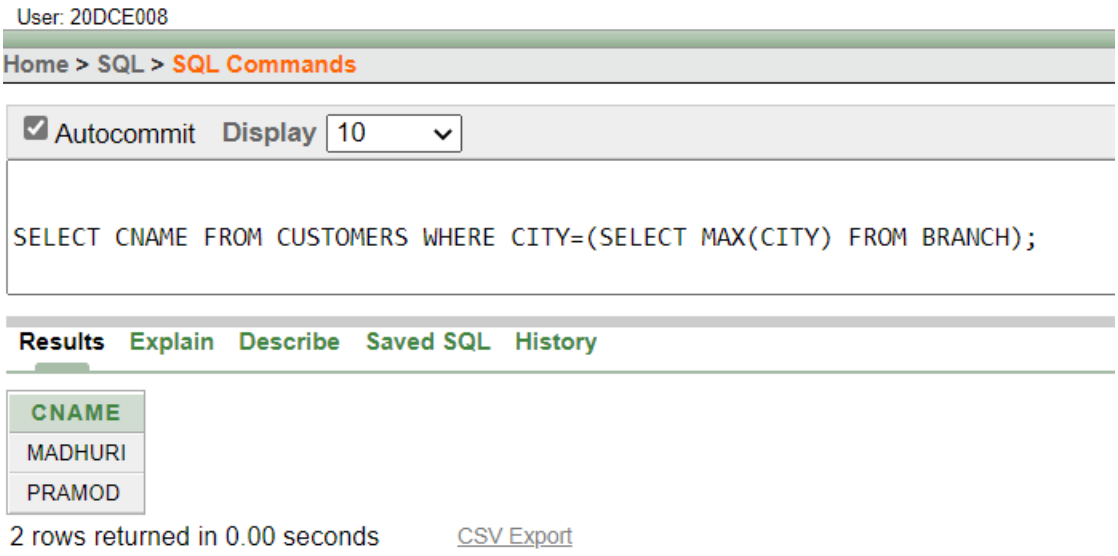
(8) List the name of branch having highest number of depositors.



(9) Give the name of cities where in which the maximum numbers of branches are located.



(10) Give name of customers living in same city where maximum depositors are located.



Conclusion:

We learnt to solve queries using concept of sub queries.

Practical-10

Aim: Manipulating Data

Program:

(1) Give 10% interest to all depositors.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
alter table deposit add interest number(10);
update deposit set interest=amount+(amount*(0.1))
select * from deposit
```

Results Explain Describe Saved SQL History

ACTNO	CNAME	BNAME	AMOUNT	ADATE	INTEREST
100	ANIL	VRCE	1000	01-MAR-95	1100
101	SUNIL	AJNI	5000	04-JAN-96	5500
102	MEHUL	KAROLBAGH	3500	17-NOV-95	3850
104	MADHURI	CHANDI	1200	17-DEC-95	1320
105	PRMOD	M.G.ROAD	3000	27-MAR-96	3300
106	SANDIP	ANDHERI	2000	31-MAR-96	2200
107	SHIVANI	VIRAR	1000	05-SEP-95	1100
108	KRANTI	NEHRUPLACE	5000	02-JUL-95	5500
109	MINU	POWAI	7000	10-AUG-95	7700

9 rows returned in 0.00 seconds CSV Export

(2) Give 10% interest to all depositors having branch vrce

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
alter table deposit add interest number(10);
update deposit set interest=amount+(amount*(0.1)) where bname='VRCE'
select * from deposit
```

Results Explain Describe Saved SQL History

ACTNO	CNAME	BNAME	AMOUNT	ADATE	INTEREST
100	ANIL	VRCE	1000	01-MAR-95	1100
101	SUNIL	AJNI	5000	04-JAN-96	5500
102	MEHUL	KAROLBAGH	3500	17-NOV-95	3850
104	MADHURI	CHANDI	1200	17-DEC-95	1320
105	PRMOD	M.G.ROAD	3000	27-MAR-96	3300
106	SANDIP	ANDHERI	2000	31-MAR-96	2200
107	SHIVANI	VIRAR	1000	05-SEP-95	1100
108	KRANTI	NEHRUPLACE	5000	02-JUL-95	5500
109	MINU	POWAI	7000	10-AUG-95	7700

9 rows returned in 0.00 seconds CSV Export

(3) Give 10% interest to all depositors living in nagpur and having branch city bombay.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10 Save Ru

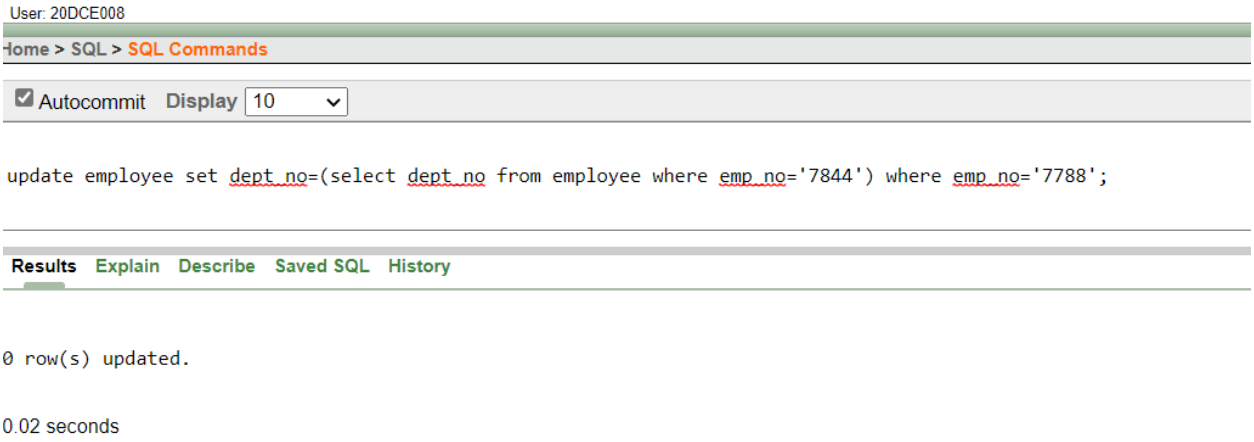
```
update deposit set interest=amount+(amount*10)/100  where cname in (select cname from customers where city='NAGPUR') and bname in (select bname from branch
where city='BOMBAY')
```

Results Explain Describe Saved SQL History

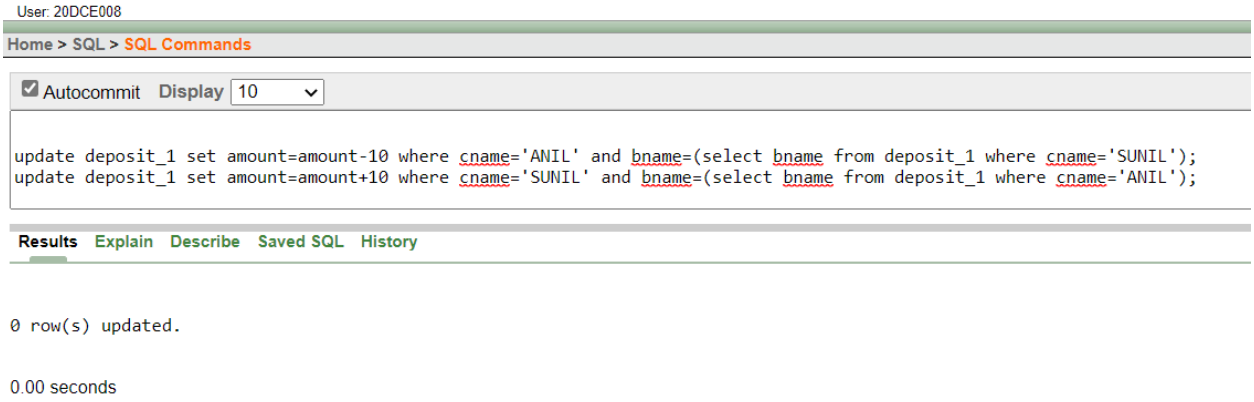
0 row(s) updated.

0.00 seconds

(4) Write a query which changes the department number of all employees with empno 7788's job to employee 7844'current department number.



(5) Transfer 10 Rs from account of anil to sunil if both are having same branch.



(6) Give 100 Rs more to all depositors if they are maximum depositors in their respective branch.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
update deposit set amount=amount+100 where amount in (select max(amount) from deposit group by bname);
select * from deposit
```

Results Explain Describe Saved SQL History

ACTNO	CNAME	BNAME	AMOUNT	ADATE	INTEREST
100	ANIL	VRCE	1100	01-MAR-95	1100
101	SUNIL	AJNI	5100	04-JAN-96	5500
102	MEHUL	KAROLBAGH	3600	17-NOV-95	3850
104	MADHURI	CHANDI	1300	17-DEC-95	1320
105	PRMOD	M.G.ROAD	3100	27-MAR-96	3300
106	SANDIP	ANDHERI	2100	31-MAR-96	2200
107	SHIVANI	VIRAR	1100	05-SEP-95	1100
108	KRANTI	NEHRUPLACE	5100	02-JUL-95	5500
109	MINU	POWAI	7100	10-AUG-95	7700

9 rows returned in 0.02 seconds CSV Export

(7) Delete depositors of branches having number of customers between 1 to 3.

User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

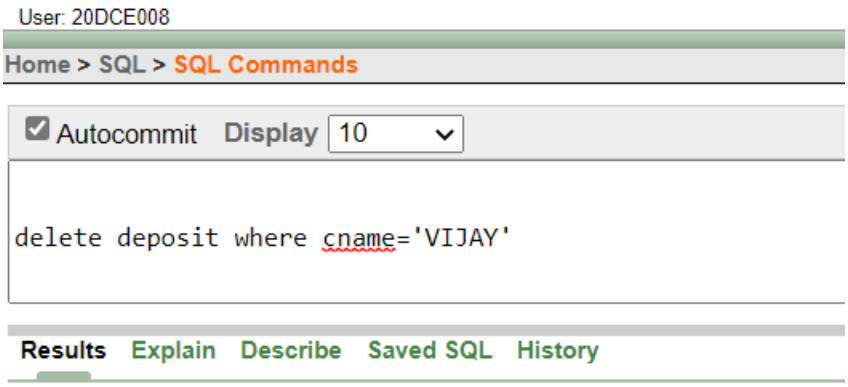
```
delete deposit where bname in (select bname from deposit group by bname having count(cname) between '1' and '3')
```

Results Explain Describe Saved SQL History

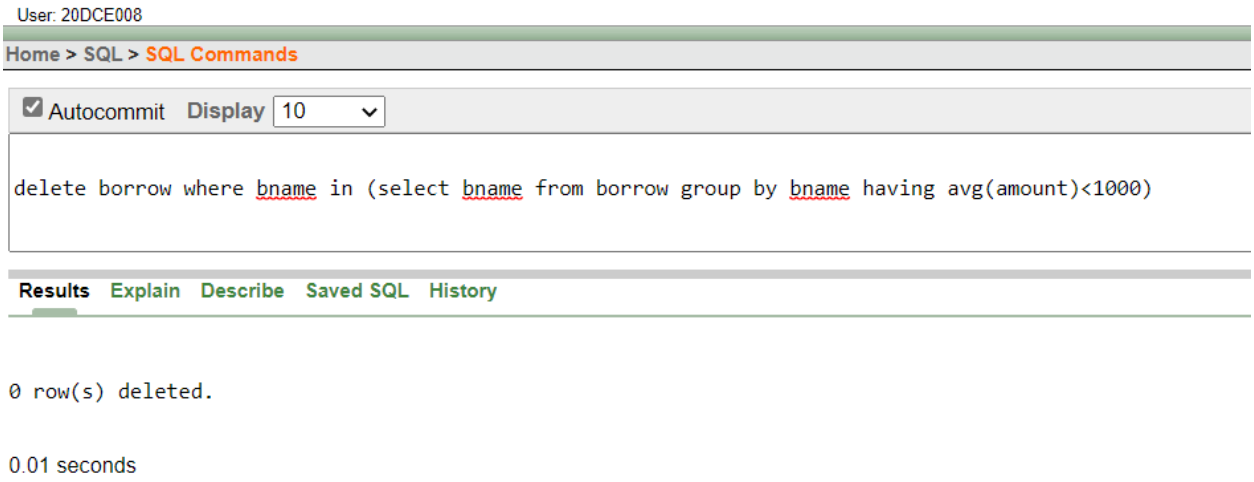
```
9 row(s) deleted.
```

0.02 seconds

(8) Delete deposit of vijay.



(9) Delete borrower of branches having average loan less than 1000.



Conclusion:

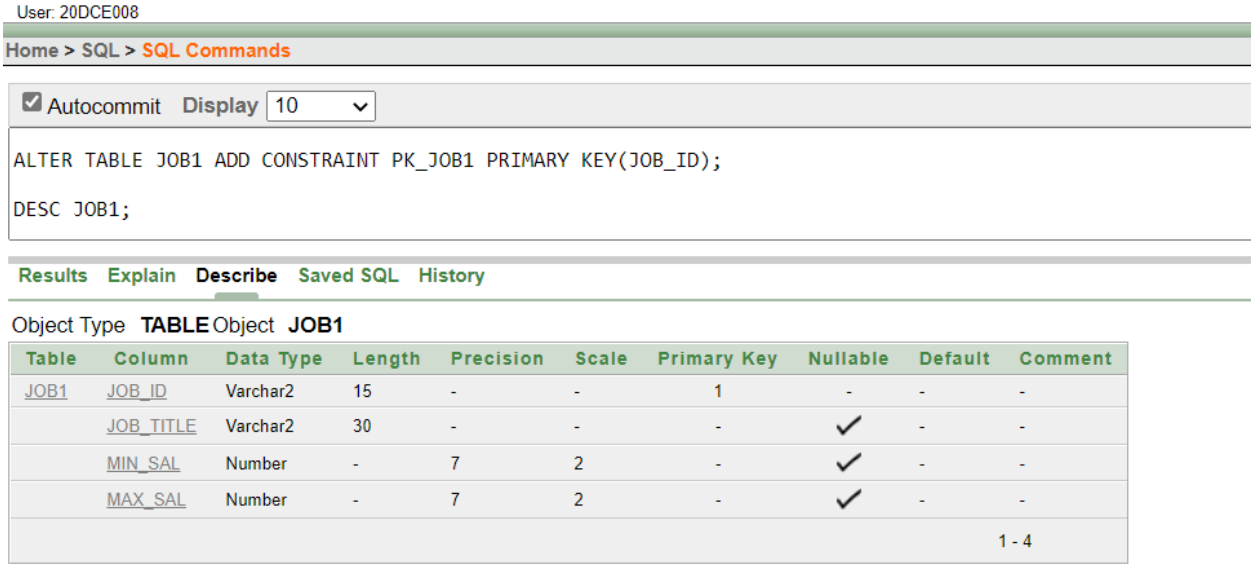
We learnt manipulation of data using different techniques.

Practical-11

Aim: Add and remove constraint

Program:

(1)Add primary key constraint on job_id in job table.



User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
ALTER TABLE JOB1 ADD CONSTRAINT PK_JOB1 PRIMARY KEY(JOB_ID);

DESC JOB1;
```

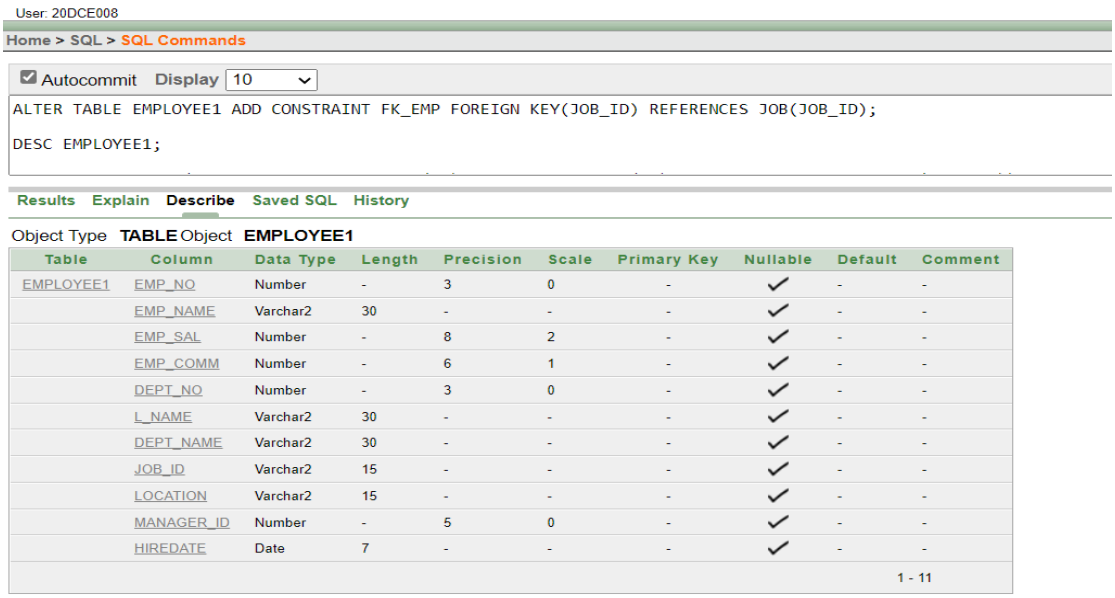
Results Explain Describe Saved SQL History

Object Type TABLE Object JOB1

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
JOB1	JOB_ID	Varchar2	15	-	-	1	-	-	-
	JOB_TITLE	Varchar2	30	-	-	-	✓	-	-
	MIN_SAL	Number	-	7	2	-	✓	-	-
	MAX_SAL	Number	-	7	2	-	✓	-	-

1 - 4

(2)Add foreign key constraint on employee table referencing job table.



User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
ALTER TABLE EMPLOYEE1 ADD CONSTRAINT FK_EMP FOREIGN KEY(JOB_ID) REFERENCES JOB(JOB_ID);

DESC EMPLOYEE1;
```

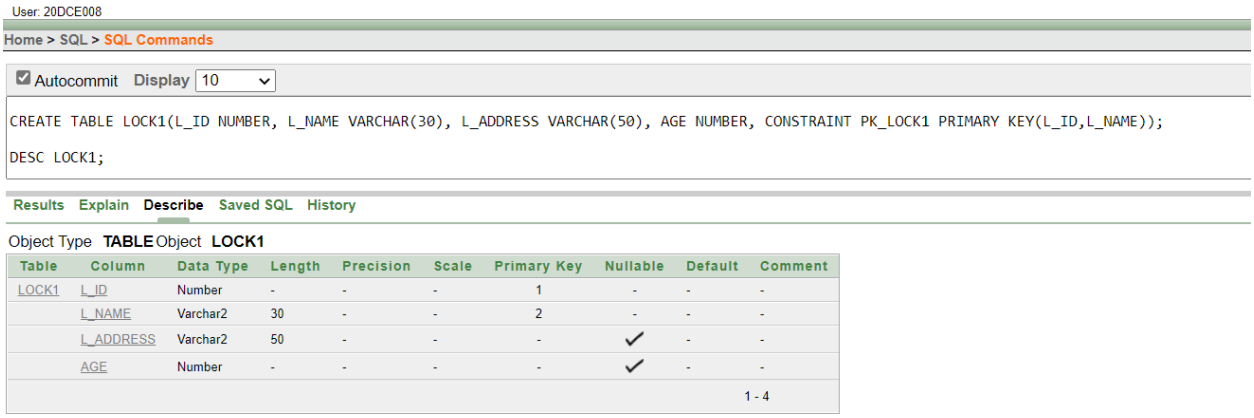
Results Explain Describe Saved SQL History

Object Type TABLE Object EMPLOYEE1

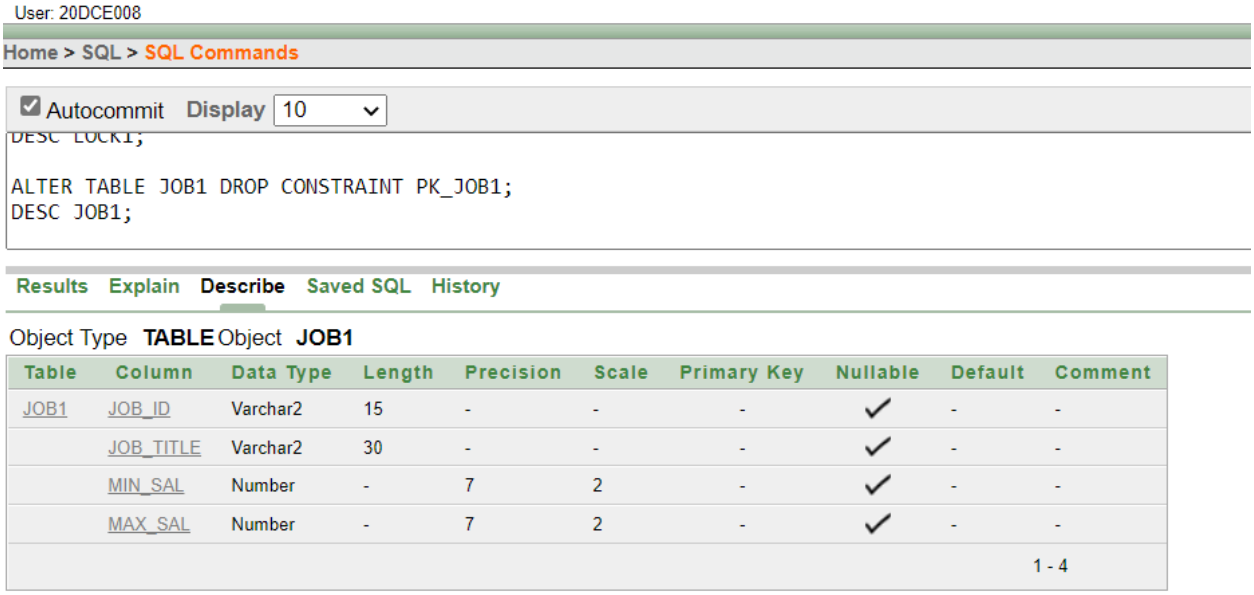
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEE1	EMP_NO	Number	-	3	0	-	✓	-	-
	EMP_NAME	Varchar2	30	-	-	-	✓	-	-
	EMP_SAL	Number	-	8	2	-	✓	-	-
	EMP_COMM	Number	-	6	1	-	✓	-	-
	DEPT_NO	Number	-	3	0	-	✓	-	-
	L_NAME	Varchar2	30	-	-	-	✓	-	-
	DEPT_NAME	Varchar2	30	-	-	-	✓	-	-
	JOB_ID	Varchar2	15	-	-	-	✓	-	-
	LOCATION	Varchar2	15	-	-	-	✓	-	-
	MANAGER_ID	Number	-	5	0	-	✓	-	-
	HIREDATE	Date	7	-	-	-	✓	-	-

1 - 11

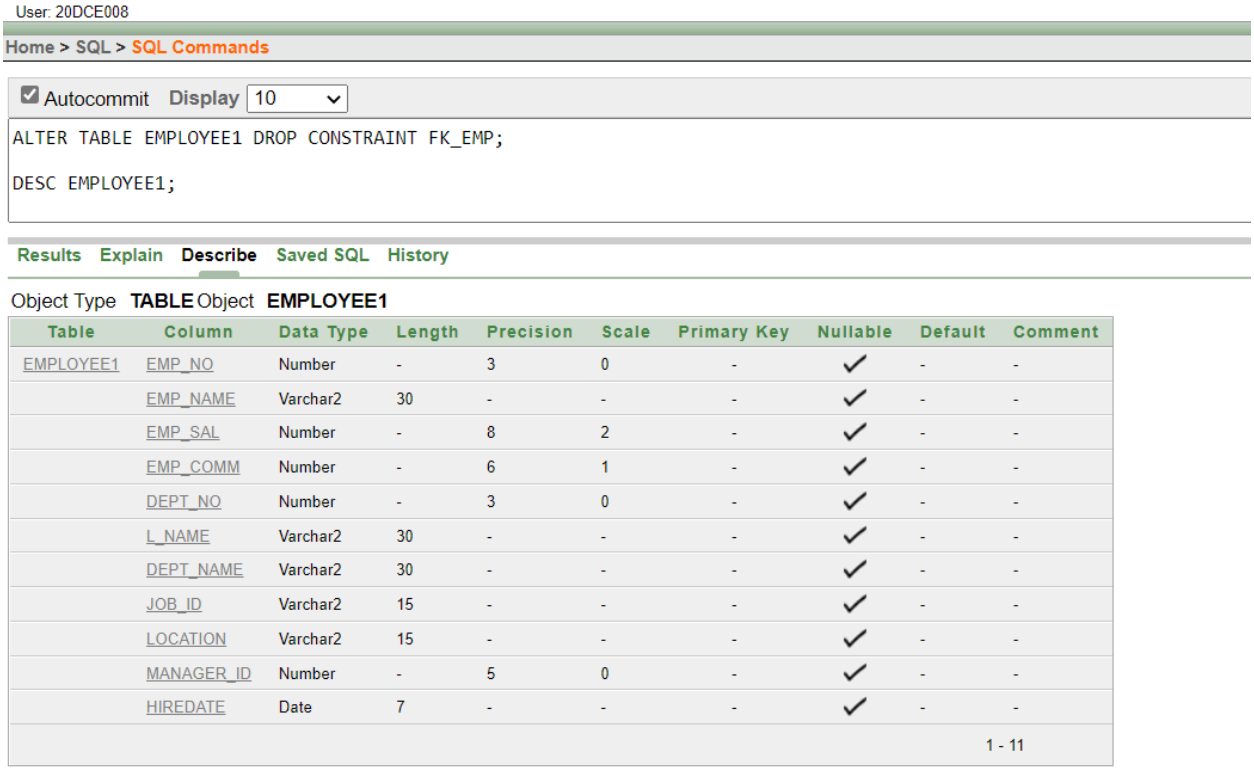
(3)Add composite primary key on lock table (lock table does not exist, while creating table add composite key)



(4)Remove primary key constraint on job_id



(5)Remove foreign key constraint on employee table



User: 20DCE008

Home > SQL > SQL Commands

Autocommit Display 10

```
ALTER TABLE EMPLOYEE1 DROP CONSTRAINT FK_EMP;

DESC EMPLOYEE1;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **EMPLOYEE1**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEE1	EMP_NO	Number	-	3	0	-	✓	-	-
	EMP_NAME	Varchar2	30	-	-	-	✓	-	-
	EMP_SAL	Number	-	8	2	-	✓	-	-
	EMP_COMM	Number	-	6	1	-	✓	-	-
	DEPT_NO	Number	-	3	0	-	✓	-	-
	L_NAME	Varchar2	30	-	-	-	✓	-	-
	DEPT_NAME	Varchar2	30	-	-	-	✓	-	-
	JOB_ID	Varchar2	15	-	-	-	✓	-	-
	LOCATION	Varchar2	15	-	-	-	✓	-	-
	MANAGER_ID	Number	-	5	0	-	✓	-	-
	HIREDATE	Date	7	-	-	-	✓	-	-

1 - 11

Conclusion:

We learnt how to add and remove constraint .

Practical-12

Aim:

Data Dictionary and E-R Diagram

Suppose that as the database administrator (DBA) in a hotel, you have to set up a database to capture all the following information that the hotel needs to maintain.

- The hotel offers three types of ROOMS, including single room, double room, and triple room.
- Every room is Identified by its unique number.
- Every employee at the hotel is either a receptionist, a cleaning staff, or a kitchen staff. Each RECEP-TIONIST is identified with her/his name, employee number and years of experience.Receptionists are responsible for ensuring the room is clean before the room is assigned to the guest. Thus, they assign a single CLEANING STAFF to clean each room every morning and/or whenever it is required. Note that the same room may need to be cleaned several times on the same day, before it gets reassigned. For each cleaning assignment, the date and the status need to be provided. The KITCHEN STAFF is characterized by their specific responsibilities, e.g. being a cook or a waiter. The cleaning staff and the kitchen staff are also uniquely identified by their employee number.
- Receptionists welcome GUESTS and upon presentation of their valid traveling documents, they allocate a unique room to each guest and specify one group of facilities which is accessible to the guest during his stay. Guests are uniquely identified with their passport number but other necessary information are also recorded about the guests, including: name, phone numbers, arrival date, departure date, and credit card number. Each FACILITY GROUP contains specific set of facilities, e.g. the bar or gym, in order to be used by the guests. The arrival and departure dates of a guest will in turn determine the occupation of a specific room.
- A guest can be accompanied with one person to have a double room or at most two people for a triple room. Each ACCOMPANYING person is identified by his/her name.

Ans:

1. **Design Data Dictionary for above problem.**

No.	Field Name	Data Type	Size	Detail	Example	Constraints
1						
2						

2. **Considering the descriptions given above, draw an ER diagram for the database, representing entities, attributes, and relationships. Hint: Pay attention to clear identification of different kinds of attributes (e.g. multi-valued, derived, and Primary key), the total participation for the relationship sets and generalization (or specialization) of entities.**