**CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY**

**DEVANG PATEL INSTITUE OF ADVANCE TECHNOLOGY AND RESEARCH**

**NAME: Parth N Patel**

**ID: 19DCS098**

**SUBJECT: Database Management System**

**SUBJECT CODE: CE246**

**SEM: 4**

**PRACTICAL-1**

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

**File System**

* A file processing system(fps) is a technique of arranging the files in a storage medium like a hard disk, pen drive, DVD, etc. It helps you to organizes the data and allows easy retrieval of files when they are required. It mostly consists of different types of files like mp3, mp4, txt, doc, etc. that are grouped into directories.
* A file system enables you to handle the way of reading and writing data to the storage medium. It is directly installed into the computer with the Operating systems such as Windows and Linux.

**Example:**

NTFS (New Technology File System), EXT (Extended File System).

## Features of a File system:

* It helps you to store data in a group of files.
* Files data are dependent on each other.
* C/C++ and COBOL languages were used to design the files.
* Shared File System Support
* Fast File System Recovery.

**Advantages of File system:**

* Enforcement of development and maintenance standards.
* Helps you to reduce redundancy
* Avoid inconsistency across file maintenance to get the integrity of data independence.
* Firm theoretical foundation (for the relational model).
* It is more efficient and cost less than a DBMS in certain situations.
* The design of file processing is simpler than designing Database.

**Disadvantages of File Processing system:**

* Each application has its data file so, the same data may have to be recorded and stored many times.
* Data dependence in the file processing system are data-dependent, but, the problem is incompatible with file format.
* The problem with security.
* Time-consuming.
* It allows you to maintain the record of the big firm having a large number of items.
* Required lots of labour work to do.

**Application of File system:**

* Language-specific run-time libraries
* API programs using it to make requests of the file system
* It is used for data transfer and positioning.
* Helps you to update the metadata
* Managing directories.

**DBMS (Database Management System):**

* Database Management System is basically a software that manages the collection of related data. It is used for storing data and retrieving the data effectively when it is needed. It also provides proper security measures for protecting the data from unauthorized access. In Database Management System the data can be fetched by SQL queries and relational algebra. It also provides mechanisms for data recovery and data backup.

**Example:**

Oracle, MySQL, MS SQL server.

**Features of DBMS:**

* A user-accessible catalog of data
* Transaction support
* Concurrency control with Recovery services
* Authorization services
* The value of data is the same at all places.
* Offers support for data communication
* Independent utility services
* Allows multiple users to share a file at the same time

**Advantages of DBMS:**

* DBMS offers a variety of techniques to store & retrieve data
* Uniform administration procedures for data
* Application programmers never exposed to details of data representation and Storage.
* A DBMS uses various powerful functions to store and retrieve data efficiently.
* Offers Data Integrity and Security
* Reduced Application Development Time
* Consume lesser space
* Reduction of redundancy.
* Data independence.

**Disadvantages of the DBMS:**

* Cost of Hardware and Software of a DBMS is quite high, which increases the budget of your organization.
* Most database management systems are often complex systems, so the training for users to use the DBMS is required.
* Data-sets begins to grow large as it provides a more predictable query response time.
* It required a processor with the high speed of data processing.
* The database can fail because or power failure or the whole system stops.

**Application of the DBMS:**

* Admission System Examination System Library System
* Payroll & Personnel Management System
* Accounting System Hotel Reservation System Airline Reservation System
* DBMS system also used by universities to keep call records, monthly bills, maintaining balances, etc.
* Finance for storing information about stock, sales, and purchases of financial instruments like stocks and bonds.

**KEY DIFFERENCES BETWEEN FPS & DBMS:**

* A file system is a software that manages and organizes the files in a storage medium, whereas DBMS is a software application that is used for accessing, creating, and managing databases.
* The file system doesn't have a crash recovery mechanism on the other hand, DBMS provides a crash recovery mechanism.
* Data inconsistency is higher in the file system. On the contrary Data inconsistency is low in a database management system.
* File system does not offer concurrency, whereas DBMS provides a concurrency facility.

**RDBMS (Relational Database Management System):**

* A relational database management system (RDBMS) is a program that allows you to create, update, and administer a relational database. Most relational database management systems use the SQL language to access the database.
* RDMBS adds the R of relational to the existing Database management technology. Created in the 1970s, RDBMS was designed to be a more sophisticated version of DBMS. RDBMS also adds a degree of finesse for the organization or the individuals accessing the data stored in the database.
* One key feature of RDBMS is that it can only keep the tabular form of data. Data in RDBMS is stored and sorted in the form of rows, columns (also called tuples and attribute in the DBMS language).

**Example:**

MySQL, PostgreSQL, Db2

**Features of RDBMS:**

* All data stored in the tables are provided by an RDBMS
* Ensures that all data stored are in the form of rows and columns
* Facilitates primary key, which helps in unique identification of the rows
* Facilitates a common column to be shared amid two or more tables
* Multi-user accessibility is facilitated to be controlled by individual users.

**Advantages of RDBMS**:

* It is secured in nature.
* The data manipulation can be done.
* It limits redundancy and replication of the data.
* It offers better data integrity.
* It provides better physical data independence.

**Disadvantages of RDBMS**:

* Software is expensive.
* It requires skilled human resources to implement.
* It is difficult to recover the lost data.
* Complex software refers to expensive hardware and hence increases overall cost to avail the RDBMS service.

**DDBMS (Distributed Database Management System):**

* Distributed Database Management System (DDBMS) is a type of DBMS which manages a number of databases hoisted at diversified locations and interconnected through a computer network. It provides mechanisms so that the distribution remains oblivious to the users, who perceive the database as a single database.

**Features of DDBMS:**

* It is used to create, retrieve, update and delete distributed databases.
* It synchronizes the database periodically and provides access mechanisms by the virtue of which the distribution becomes transparent to the users.
* It is used in application areas where large volumes of data are processed and accessed by numerous users simultaneously.
* It is designed for heterogeneous database platforms.
* It maintains confidentiality and data integrity of the databases.

**Advantages of DDBMS**:

* Reflects organizational structure
* Improved share ability
* Improved availability
* Improved reliability
* Improved performance

**Disadvantages of DDBMS**:

* Increased Cost
* Integrity control more difficult,
* Lack of standards,
* Database design more complex.
* Complexity of management and control. Applications must recognize data location and they must be able to stitch together data from various sites.

**CONCLUSION:**

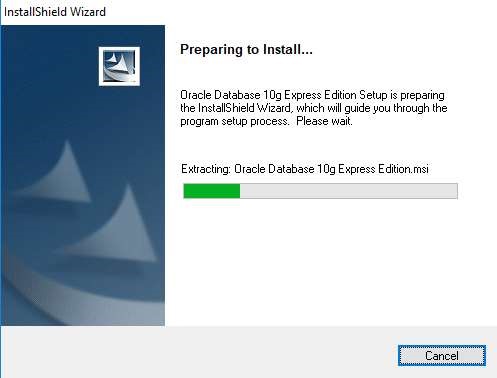
In this practical, we learned the basics of DBMS and SQL.

**PRACTICAL-2**

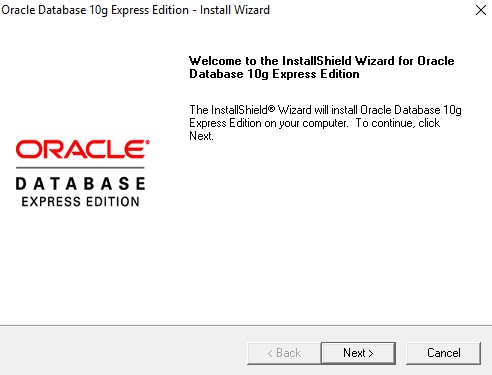
**Introduction to Oracle (step by step installation, introduction of sql, plsql).**

1. Download Oracle 10g from below link:

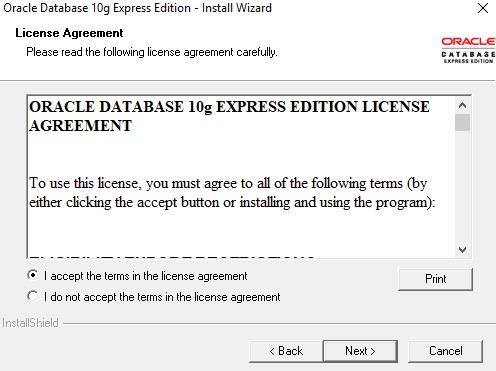
1. Install it by double clicking .exe which you have downloaded



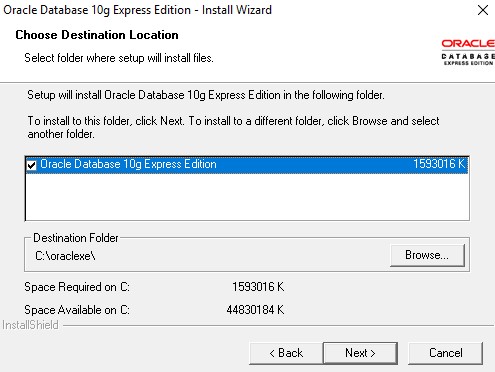
1. Click on Next button



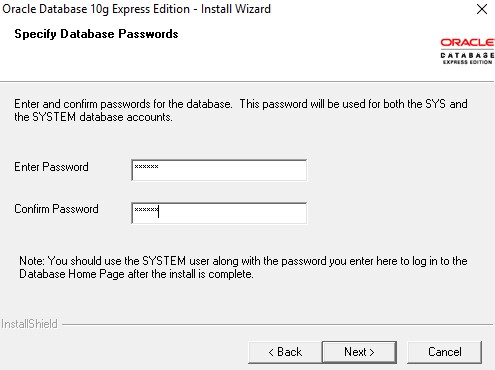
1. Accept license agreement and click on next button



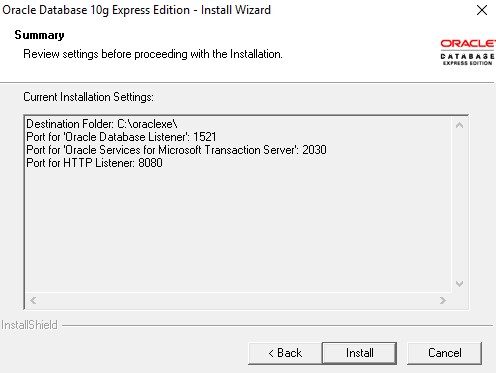
1. Click on next button



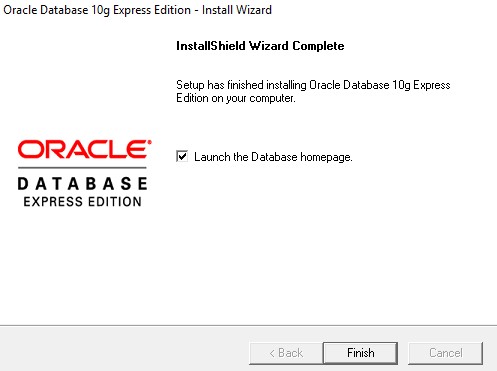
1. Enter password and confirm password for SYS and SYSTEM user. Please remember it because once installation will be over you have to enter it. To make it easy to remember give password as : “oracle”



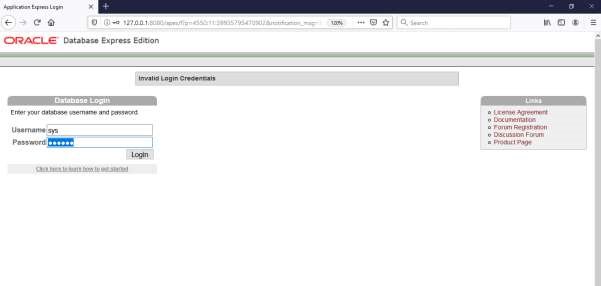
1. Click on install button



1. Click on finish button.



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



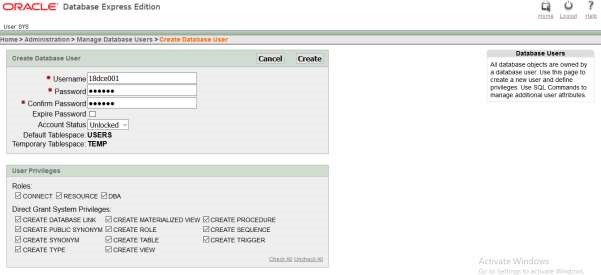
1. Click on Administration



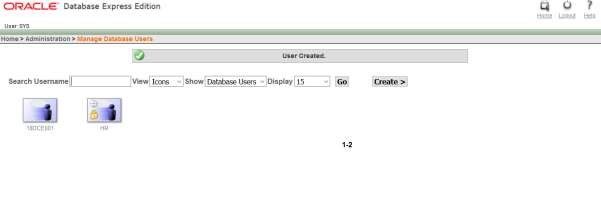
11)Now click on “database user drop down button”. From that click on “create user”.



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.



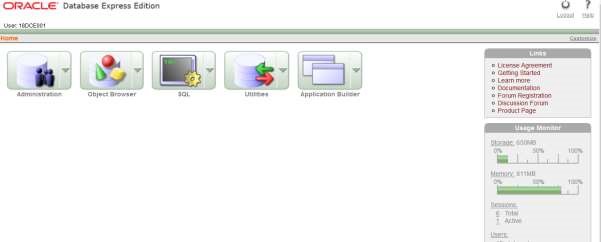
14) Click on login



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



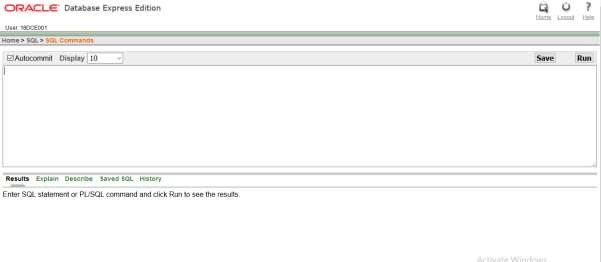
1. Click on SQL



1. Click on SQL Commands



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



**What is SQL ?**

• SQL is a standard language for accessing and manipulating databases.

• SQL is a standard language for accessing and manipulating databases. SQL stands for

Structured Query Language

• SQL lets you access and manipulate databases

• SQL became a standard of the American National Standards Institute (ANSI) in 1986,

and of the International Organization for Standardization (ISO) in 1987.

• SQL can retrieve data, update, insert, create tables, create new database, can set

permissions on tables, etc.

**What is PLSQL?**

• PL/SQL stands for “Procedural Language extensions to the Structured Query

Language”.

• SQL is a popular language for both querying and updating data in the relational

database management systems (RDBMS).

• PL/SQL adds many procedural constructs to SQL language to overcome some

limitations of SQL.

• Besides, PL/SQL provides a more comprehensive programming language solution for

building mission-critical applications on Oracle Databases.

• PL/SQL is a highly structured and readable language. Its constructs express the intent

of the code clearly. Also, PL/SQL is a straightforward language to learn.

• PL/SQL is a standard and portable language for Oracle Database development. If you

develop a program that executes on an Oracle Database, you can quickly move it to

another compatible Oracle Database without any changes.

**CONCLUSION:**

In this practical, we learned the basics of oracle database and SQL and PL/SQL.

**PRACTICAL-3**

**To study DDL-create and DML-insert commands.**

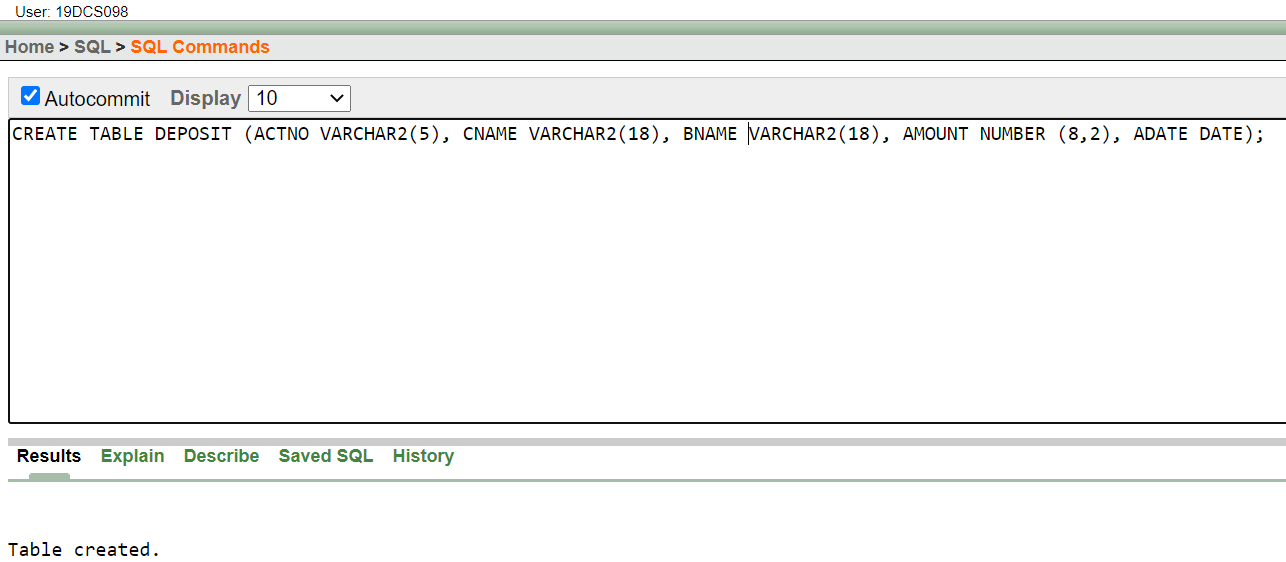
**(i) Create tables according to the following definition.**

● CREATE TABLE D

EPOSIT (ACTNO VARCHAR2(5), CNAME

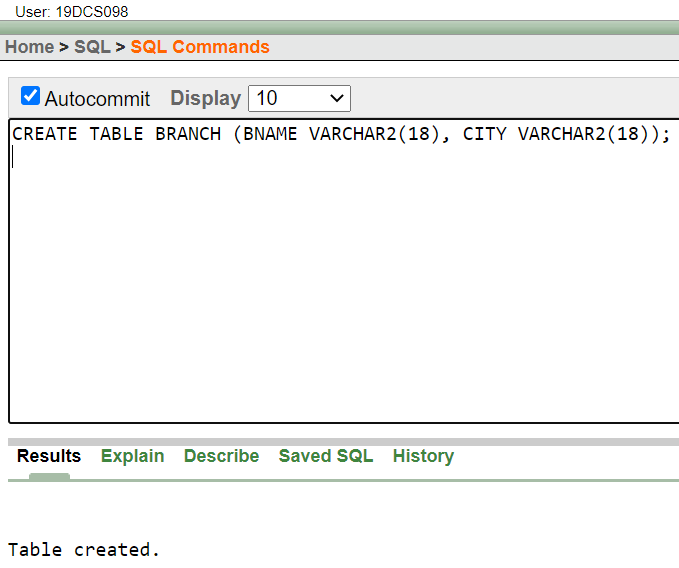
VARCHAR2(18), BNAME VARCHAR2(18), AMOUNT NUMBER (8,2),

ADATE DATE);



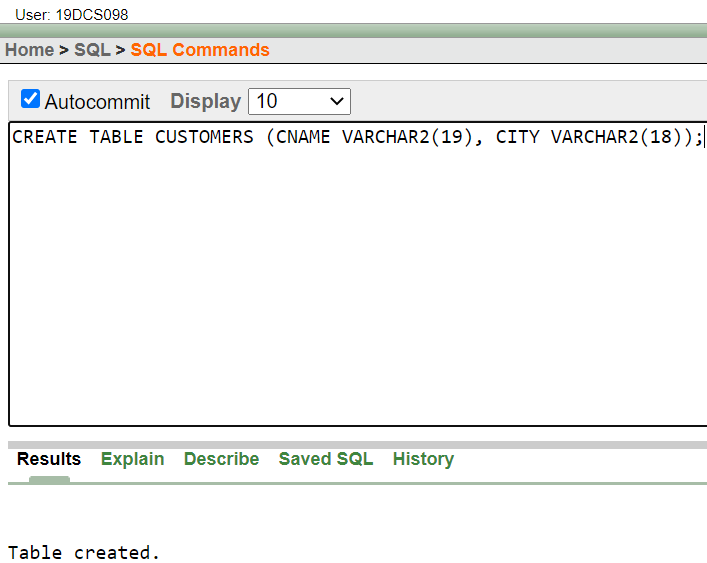
● CREATE TABLE BRANCH (BNAME VARCHAR2(18), CITY

VARCHAR2(18));



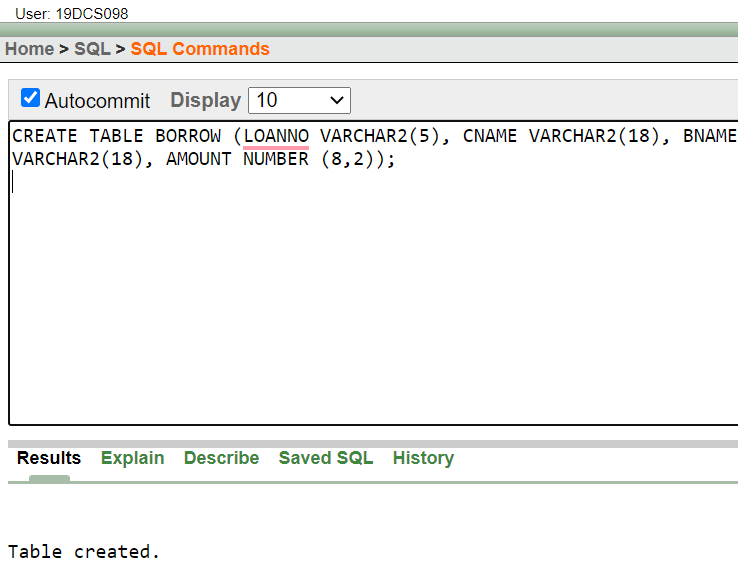
● CREATE TABLE CUSTOMERS (CNAME VARCHAR2(19), CITY

VARCHAR2(18));

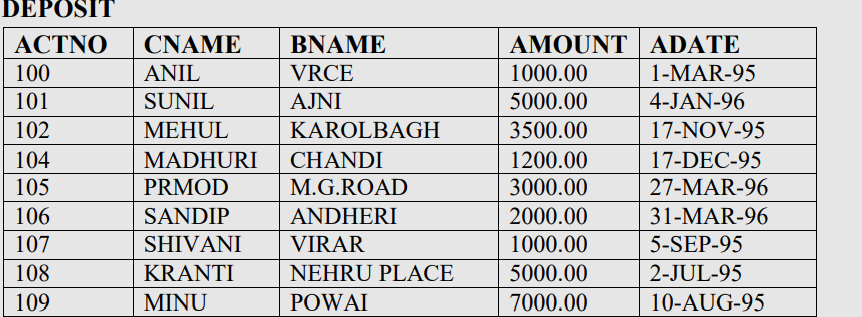


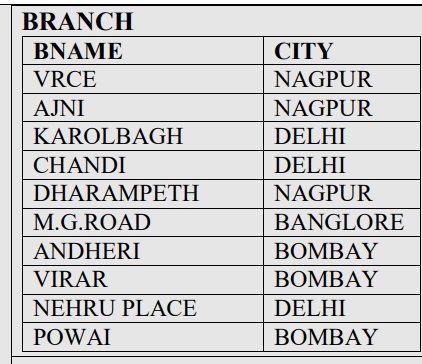
● CREATE TABLE BORROW (LOANNO VARCHAR2(5), CNAME

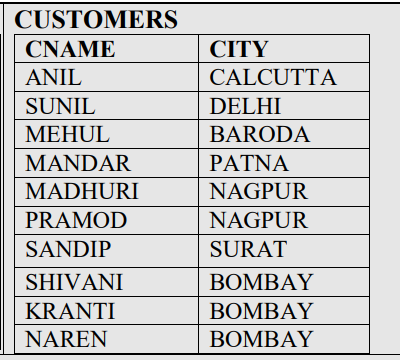
VARCHAR2(18), BNAME VARCHAR2(18), AMOUNT NUMBER (8,2));

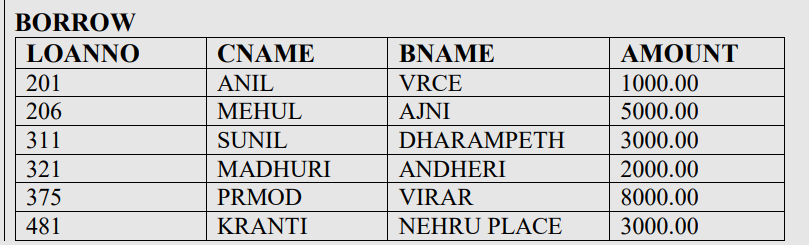


**(ii) Insert the data as shown below.**



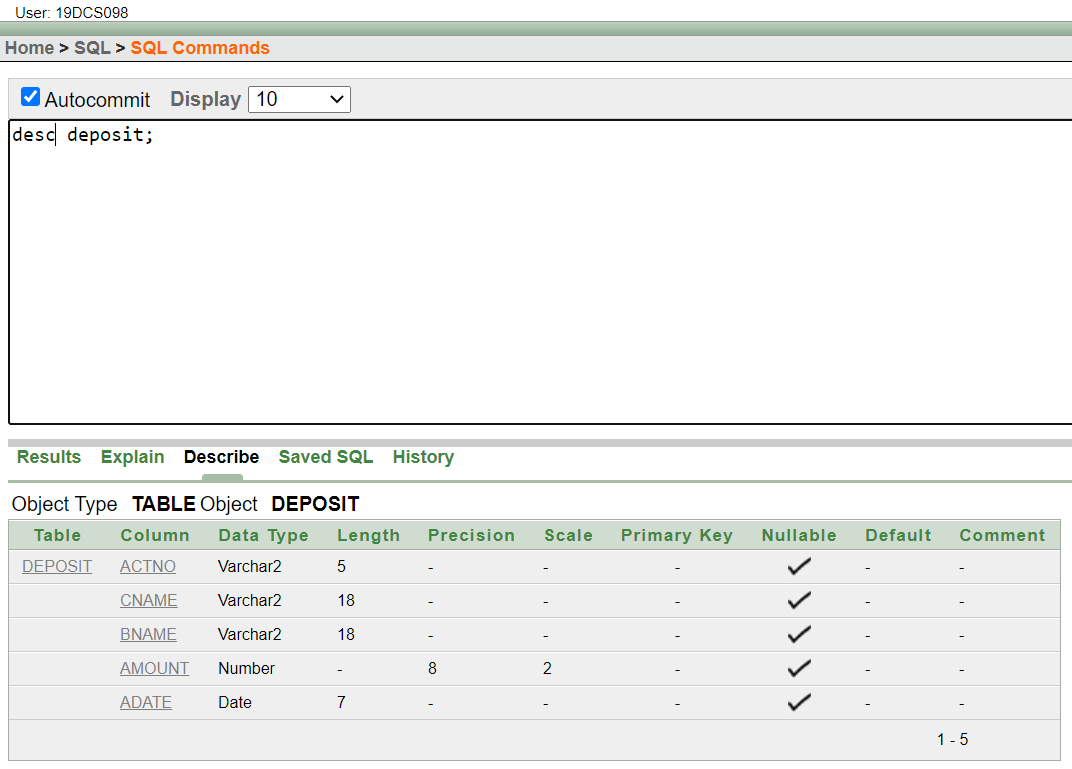


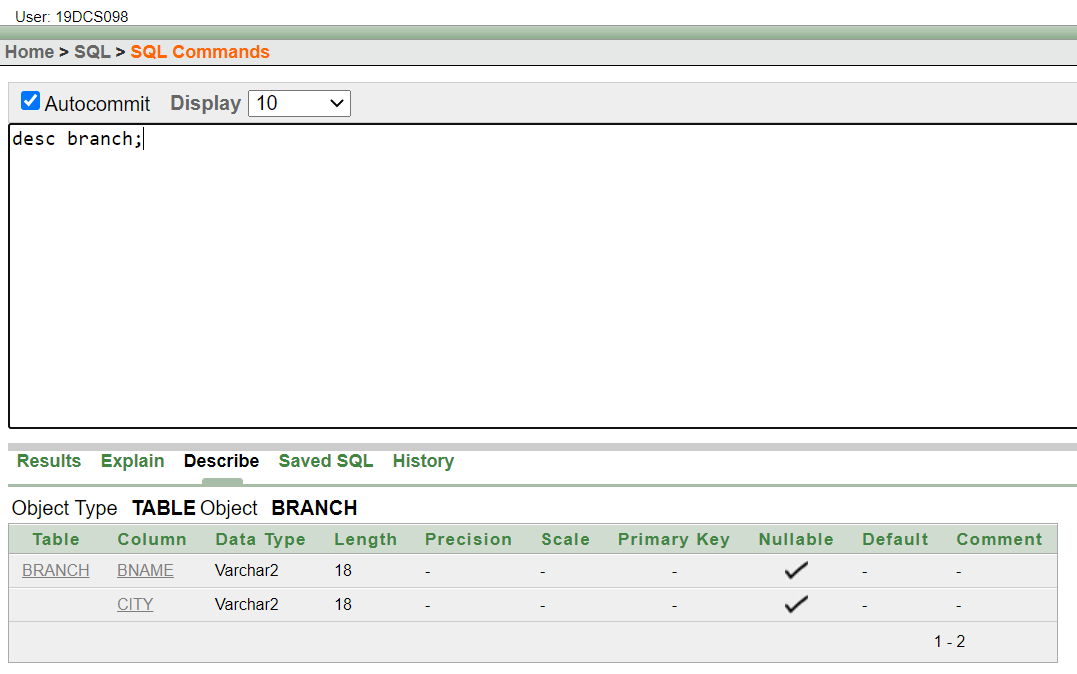




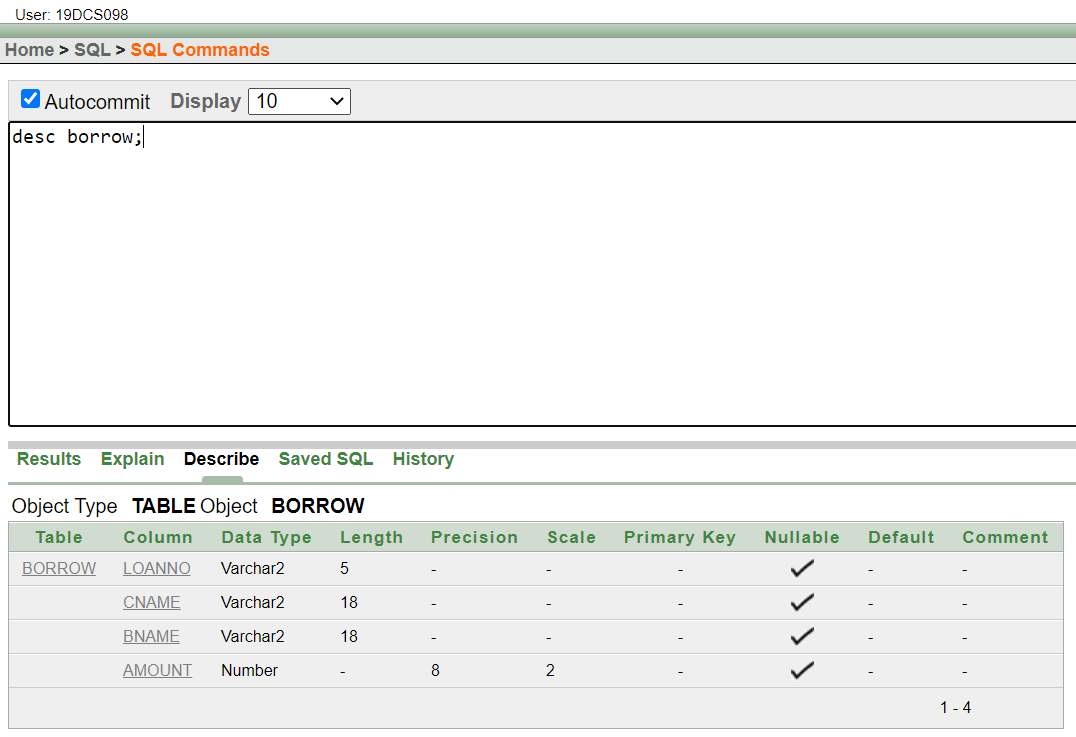
• **From the above given tables perform the following queries:**

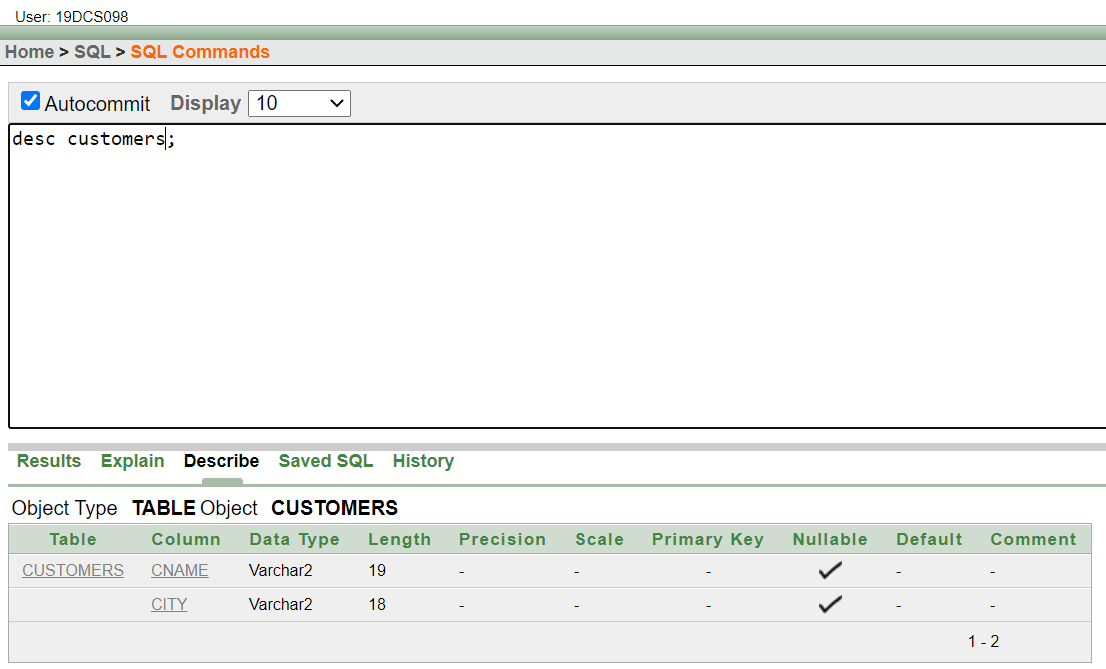
1. **Describe deposit, branch.**

e

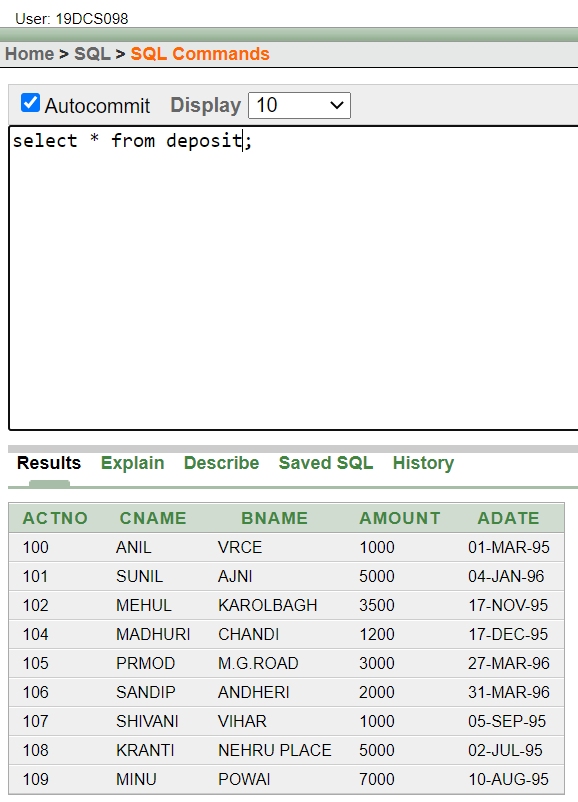


1. **Describe borrow, customers.**

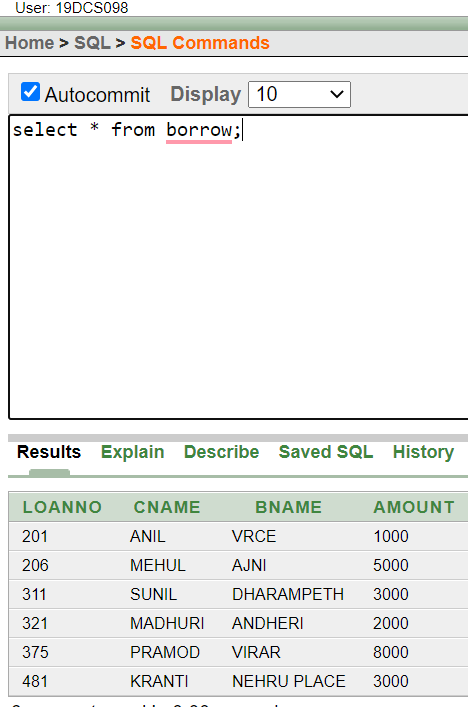




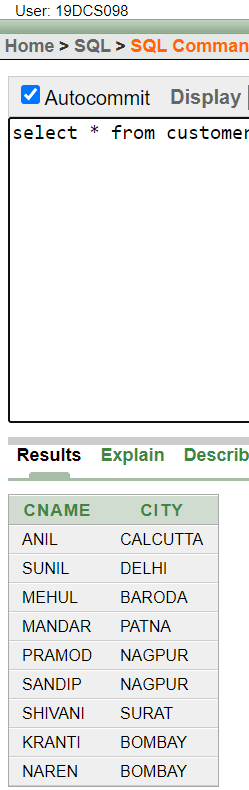
1. **List all data from table DEPOSIT.**



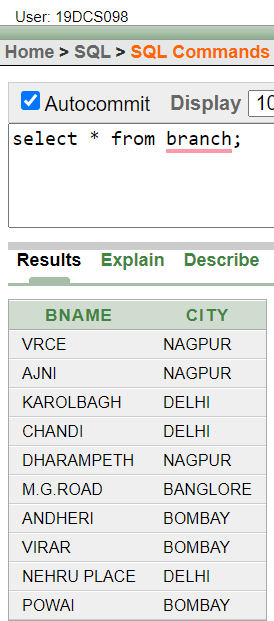
1. **List all data from table BORROW.**



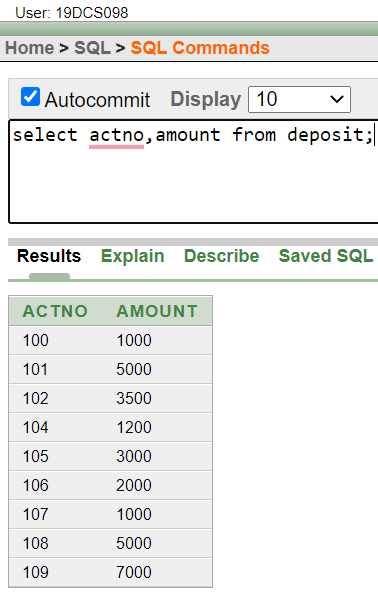
1. **List all data from table CUSTOMERS.**



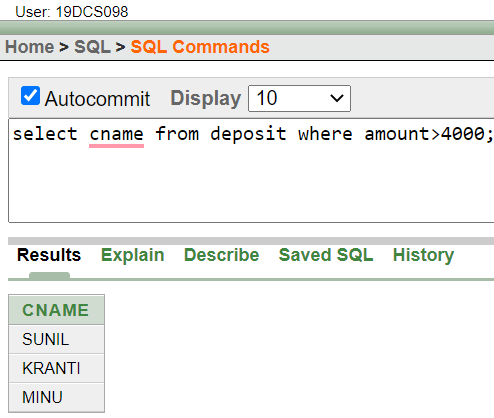
1. **List all data from table BRANCH.**



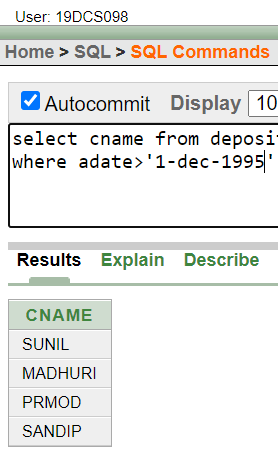
1. **Give account no and amount of depositors.**



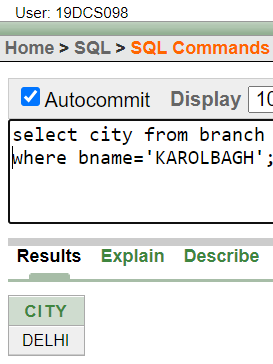
1. **Give name of depositors having amount greater than 4000.**



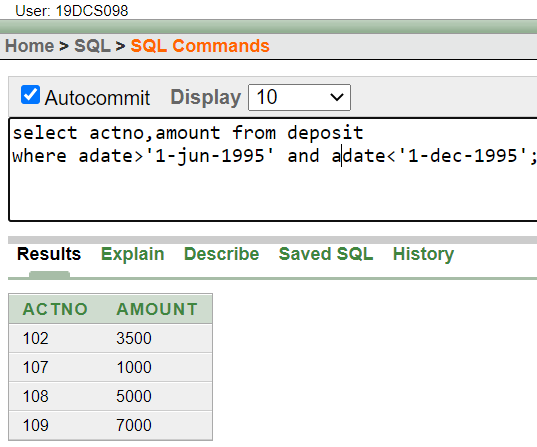
1. **Give name of customers who opened account after date '1-12-96'.**



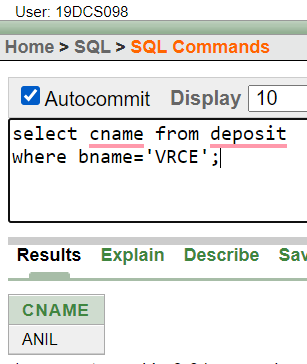
1. **Give name of city where branch karolbagh is located.**



**(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.**



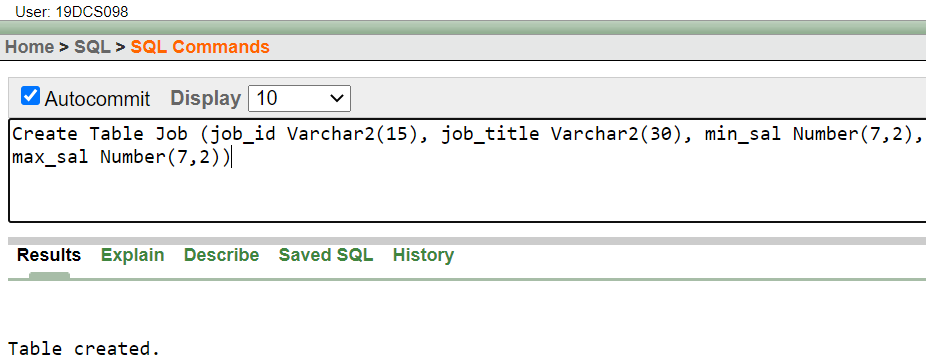
**CONCLUSION:**

In the above practicals, we learned the basics of DDL and DML.

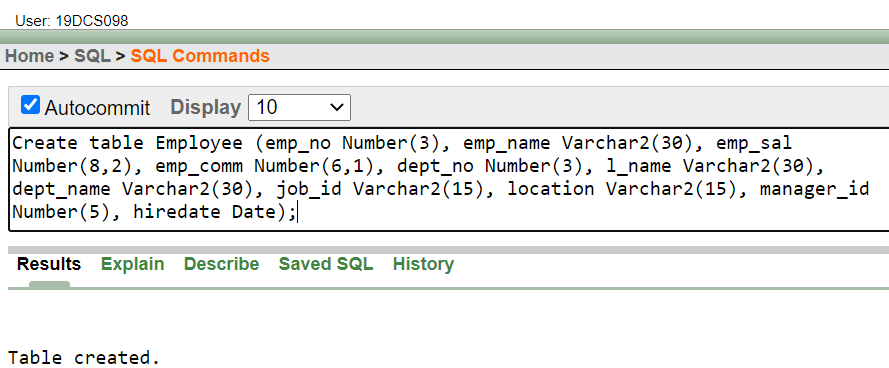
**PRACTICAL-4**

**Create the below given table and insert the data accordingly**

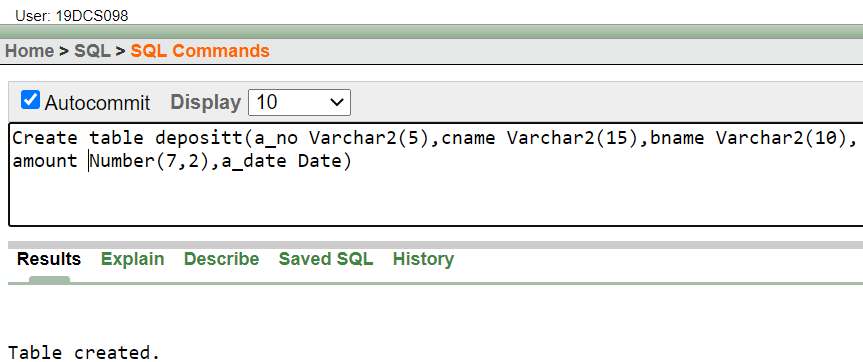
• **Create Table Job (job\_id, job\_title, min\_sal, max\_sal)**



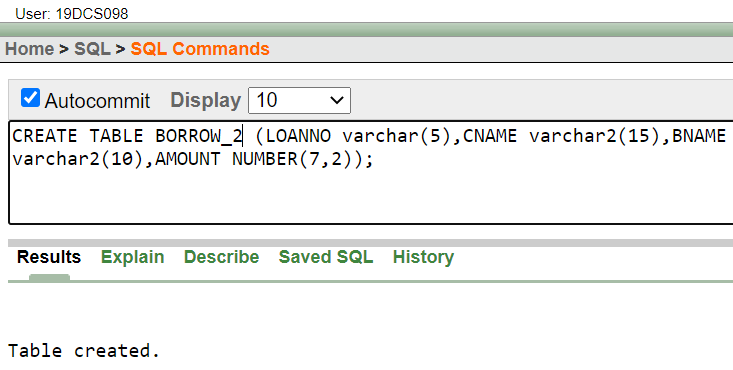
Create table Employee (emp\_no, emp\_name, emp\_sal, emp\_comm, dept\_no, l\_name, dept\_name,job\_id, location, manager\_id, hiredate)



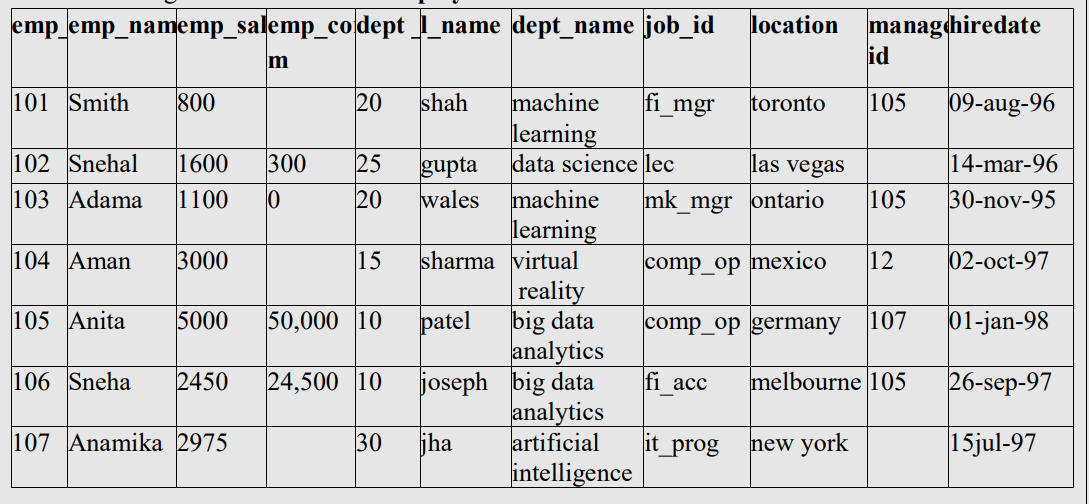
Create table deposit(a\_no,cname,bname,amount,a\_date).



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



• **Insert following values in the table Employee.**



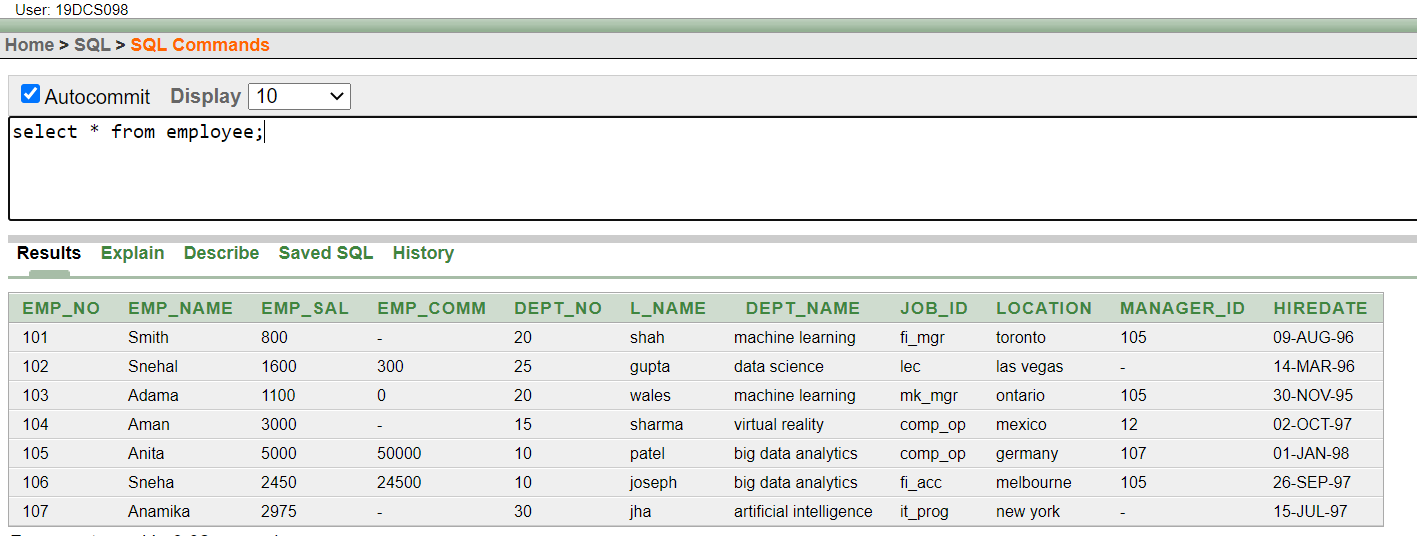
• **Insert following values in the table Job.**

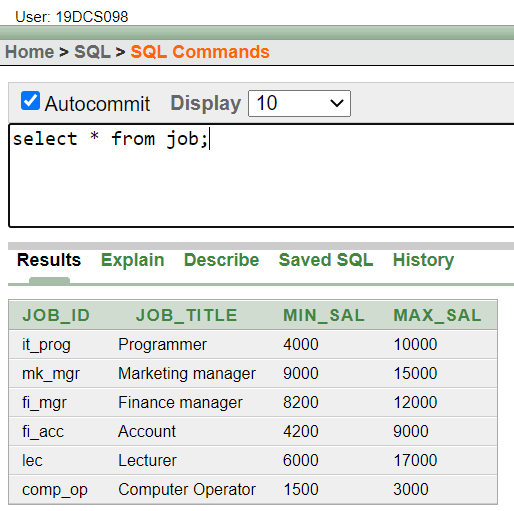


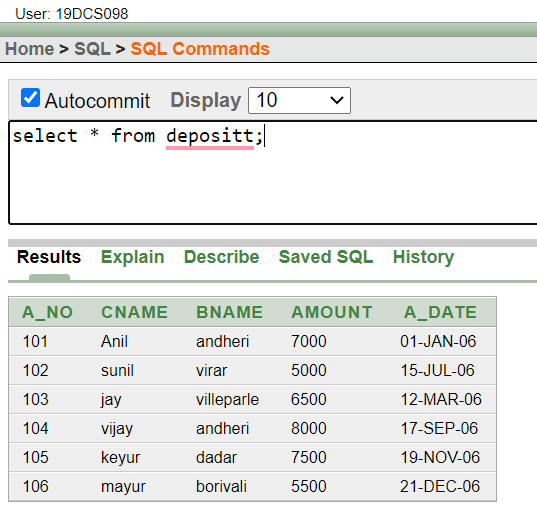


**Perform following queries**

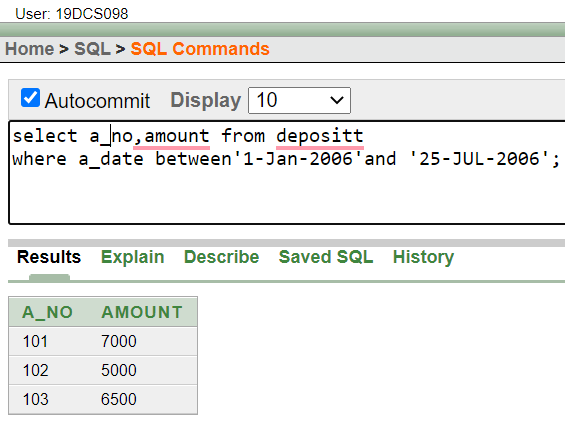
1. **Retrieve all data from employee, jobs and deposit.**



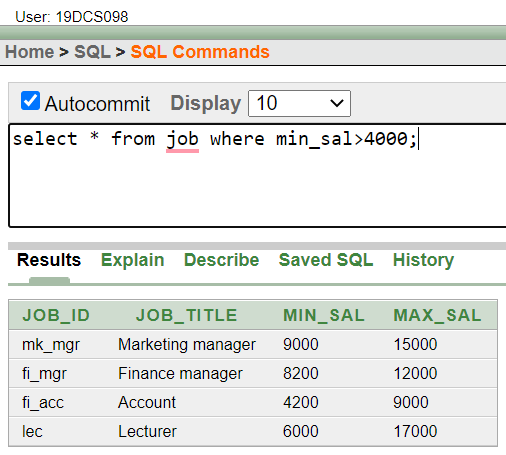




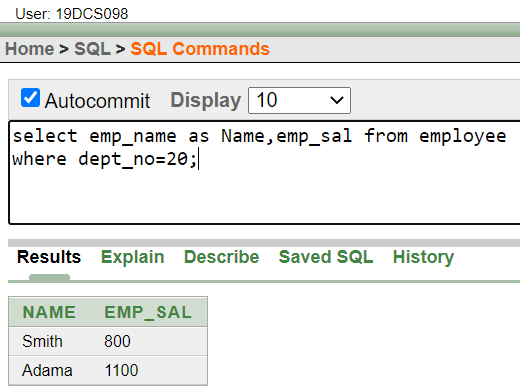
1. **Give details of account no. and deposited rupees of customers having account opened between dates 01-01-06 and 25-07-06.**



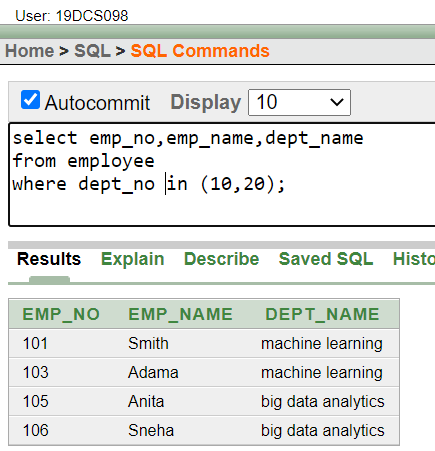
1. **Display all jobs with minimum salary is greater than 4000.**



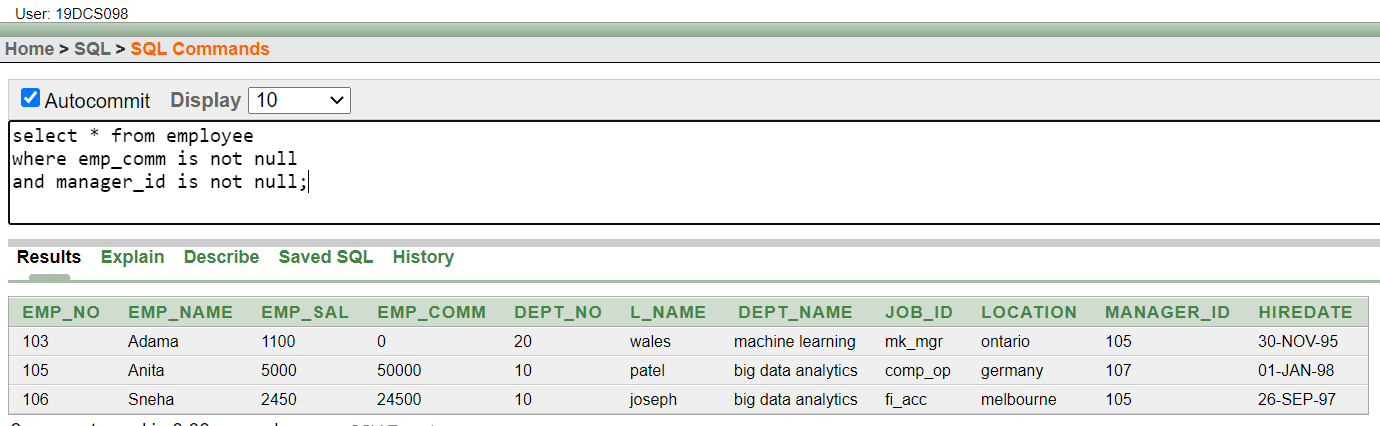
1. **Display name and salary of employee whose department no is 20. Give alias name to name of employee.**



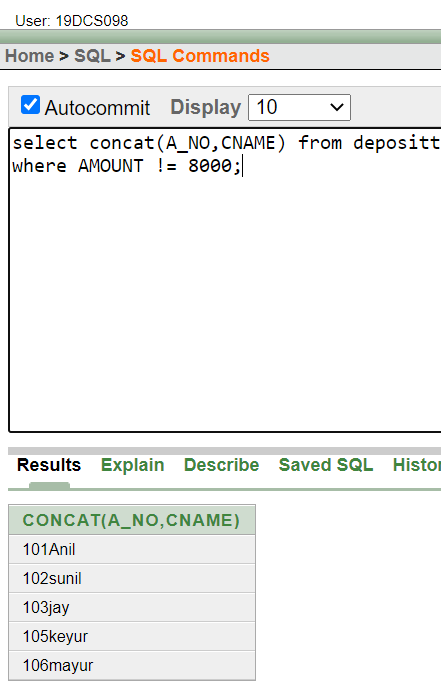
1. **Display employee no, name and department details of those employee whose department lies in (10,20).**



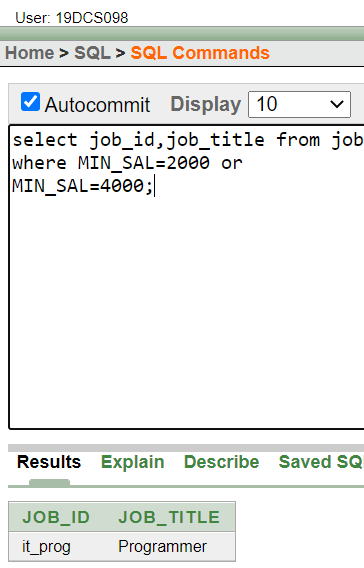
(6) **Display the non-null values of employees.**



(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.**

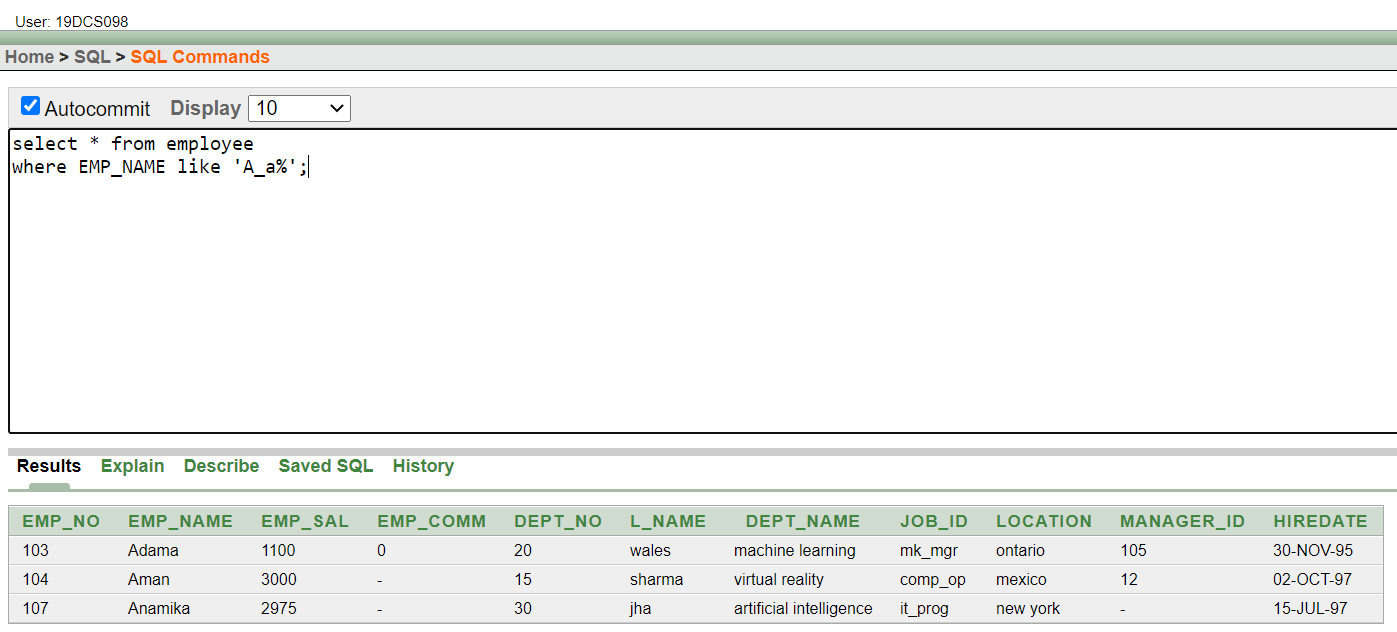


(8) **Display the content of job details with minimum salary either 2000 or 4000.**

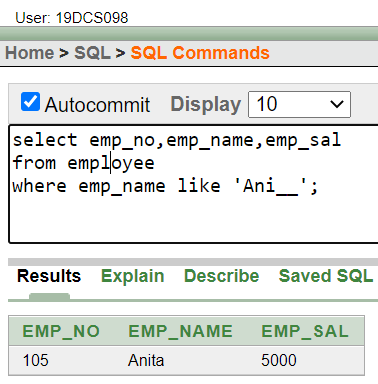


**To study various options of LIKE predicate**

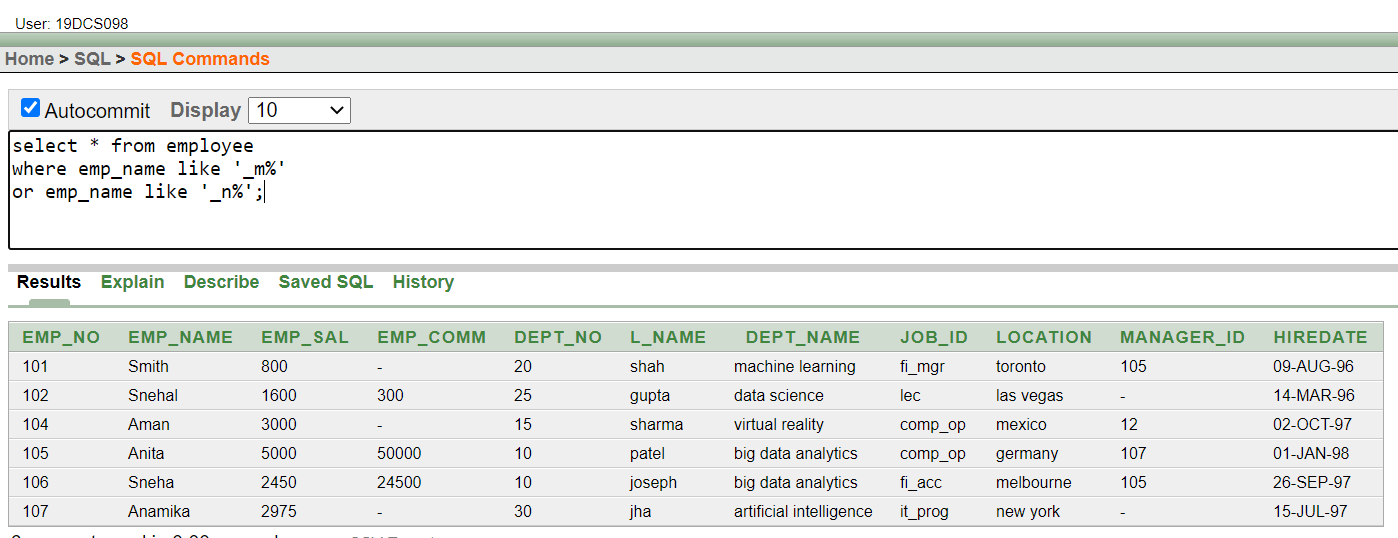
1. **Display all employee whose name start with ‘A’ and third character is ‘‘a’.**



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



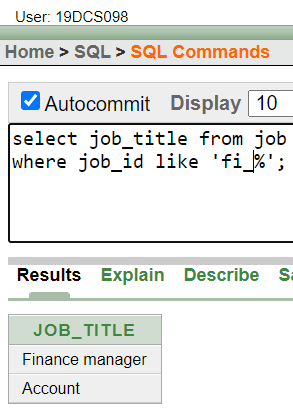
1. **Display all information of employee whose second character of name is either ‘M’ or ‘N’.**



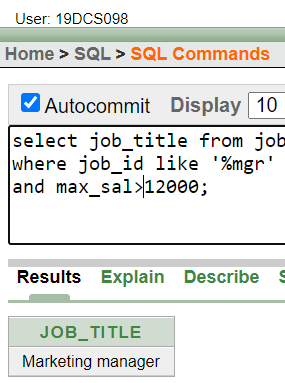
1. **Find the list of all customer name whose branch is in ‘andheri’ or ‘dadar’ or ‘virar’.**



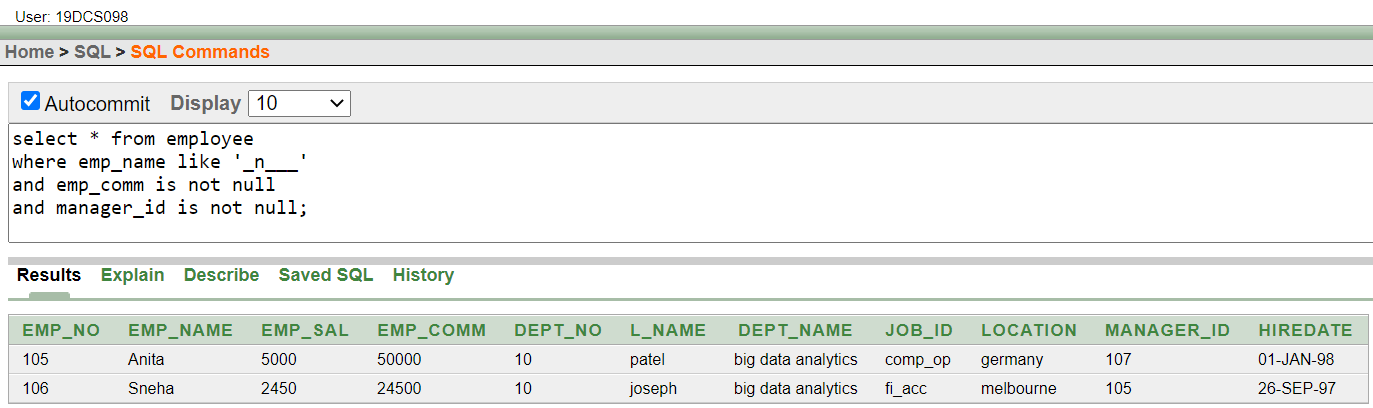
1. **Display the job name whose first three character in job id field is ‘FI\_’**



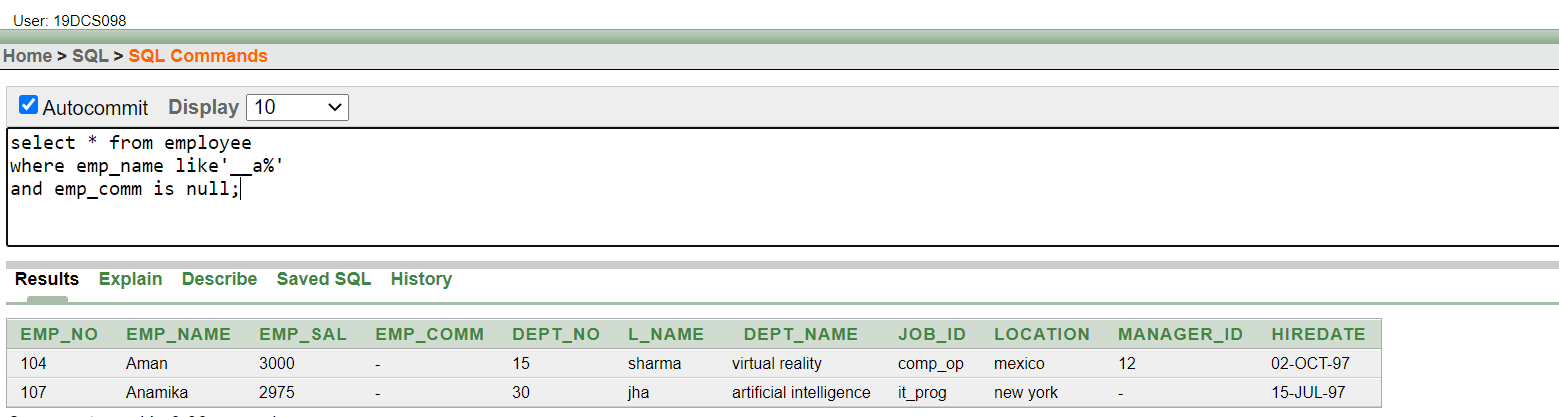
1. **Display the title/name of job who’s last three character are ‘\_MGR’ and their maximum salary is greater than Rs 12000**



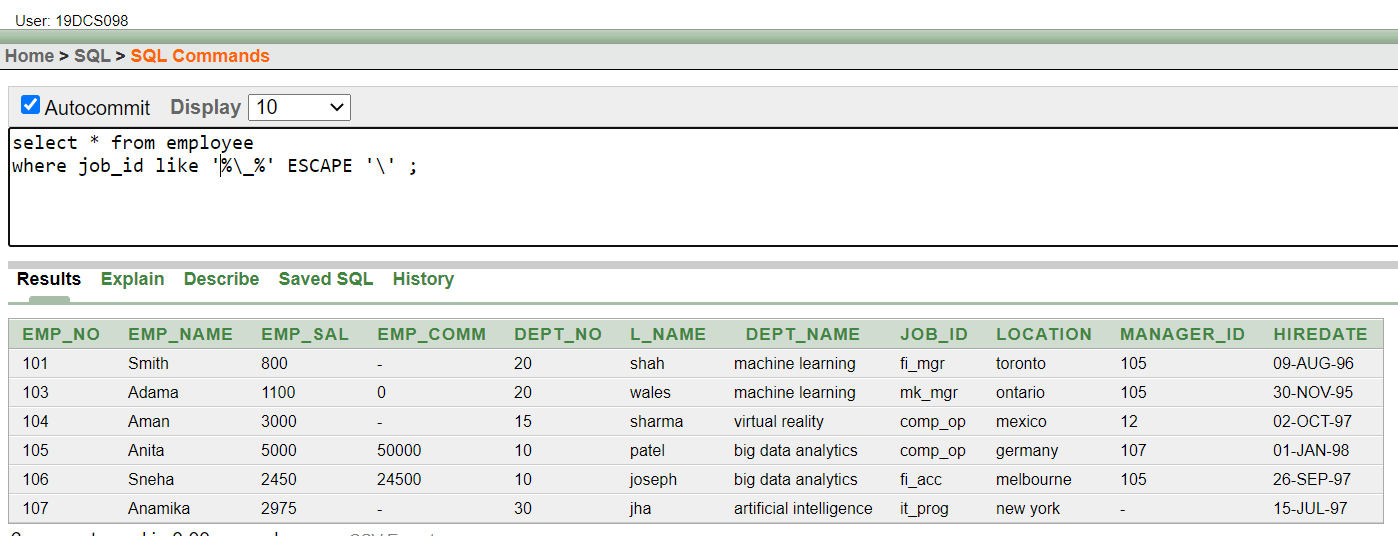
1. **Display the non-null values of employees and also employee name second character should be ‘n’ and string should be 5-character long.**



1. **Display the null values of employee and also employee name’s third character should be ‘a’.**



1. **What will be output if you are giving LIKE predicate as ‘%\\_%’ ESCAPE ‘\’**



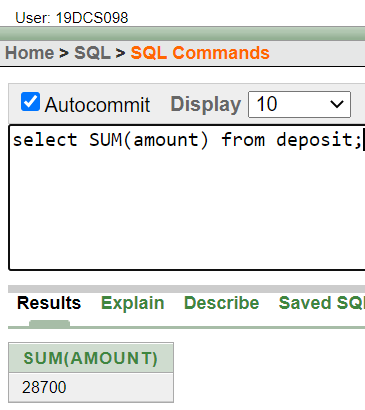
**CONCLUSION:**

In the above practical, we learned DDL,DML and the concept of ‘LIKE’

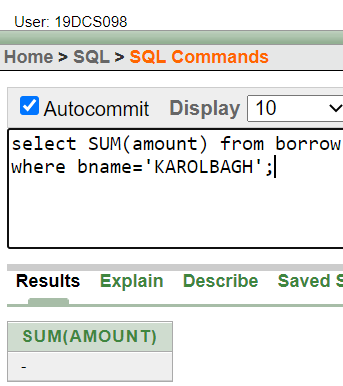
**PRACTICAL-5**

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

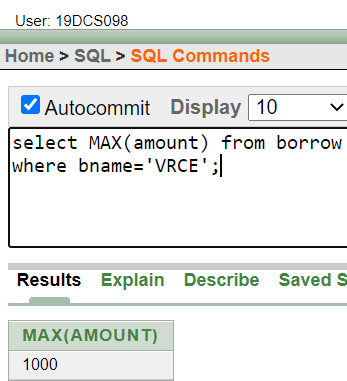
1. **List total deposit from deposit.**



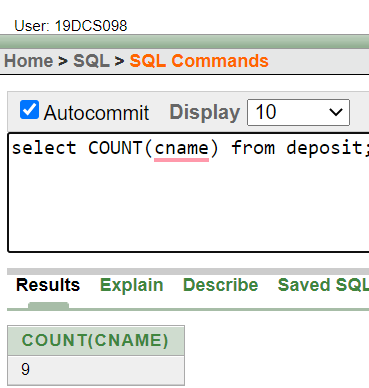
1. **List total loan from karolbagh branch**



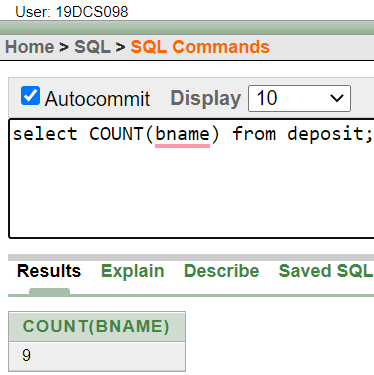
1. **Give maximum loan from branch vrce**



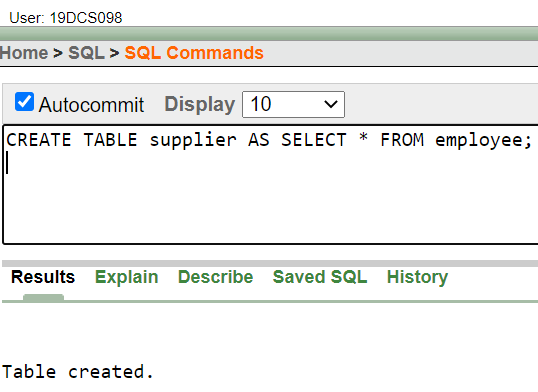
1. **Count total number of customers**

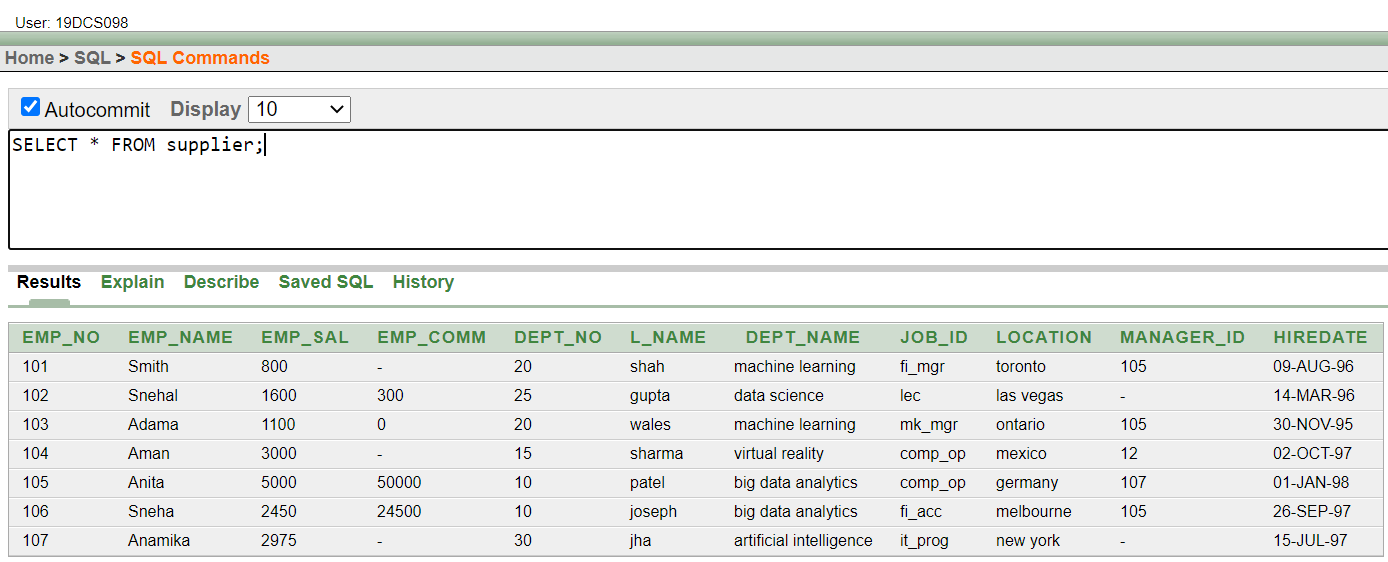


1. **Count total number of customer’s cities**

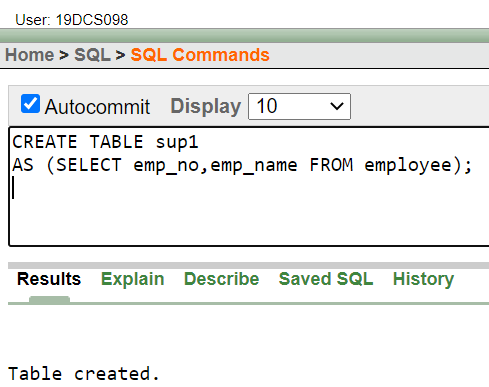


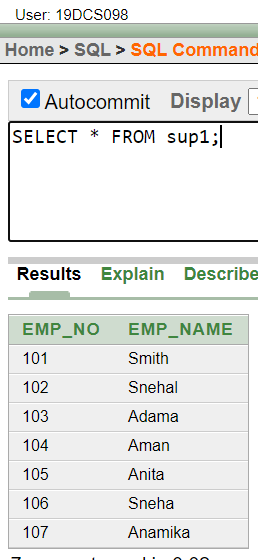
1. **Create table supplier from employee with all the columns**



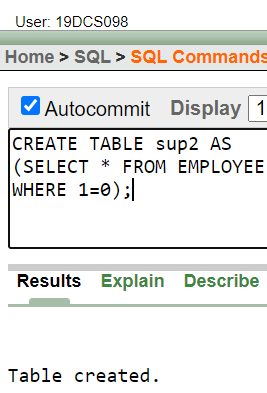


1. **Create table sup1 from employee with first two columns.**

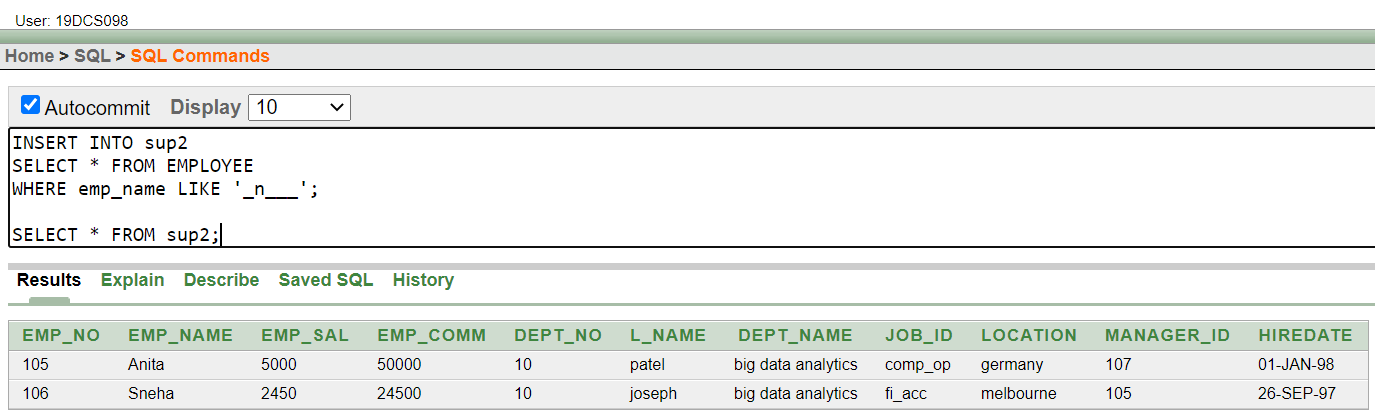




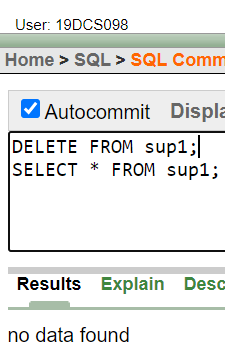
1. **Create table sup2 from employee with no data**



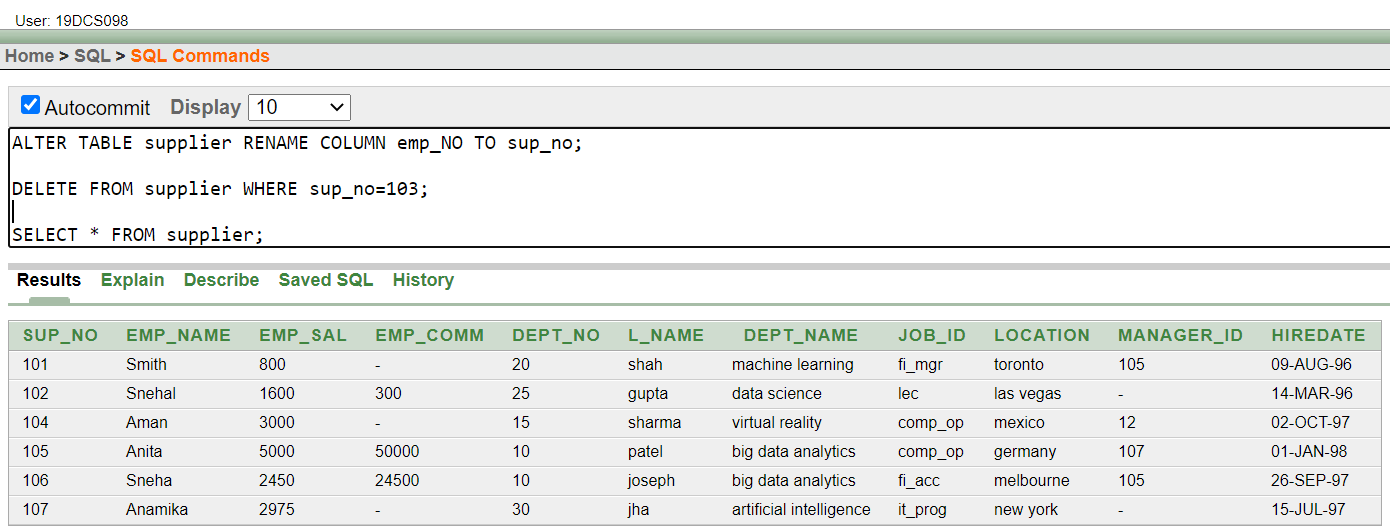
1. **Insert the data into sup2 from employee whose second character should be ‘n’ and string should be 5 characters long in employee name field.**



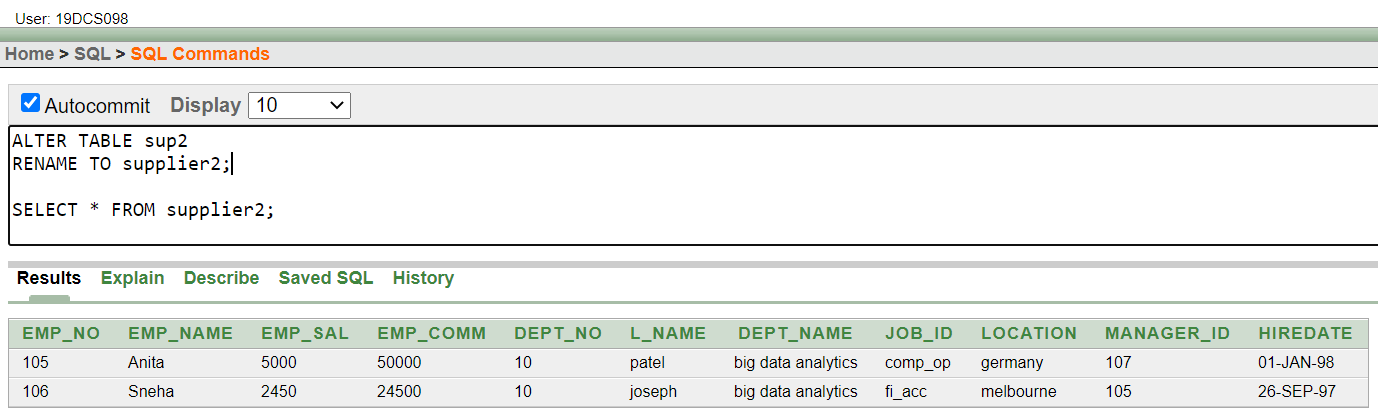
1. **Delete all the rows from sup1.**



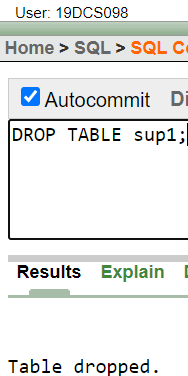
1. **Delete the detail of supplier whose sup\_no is 103**



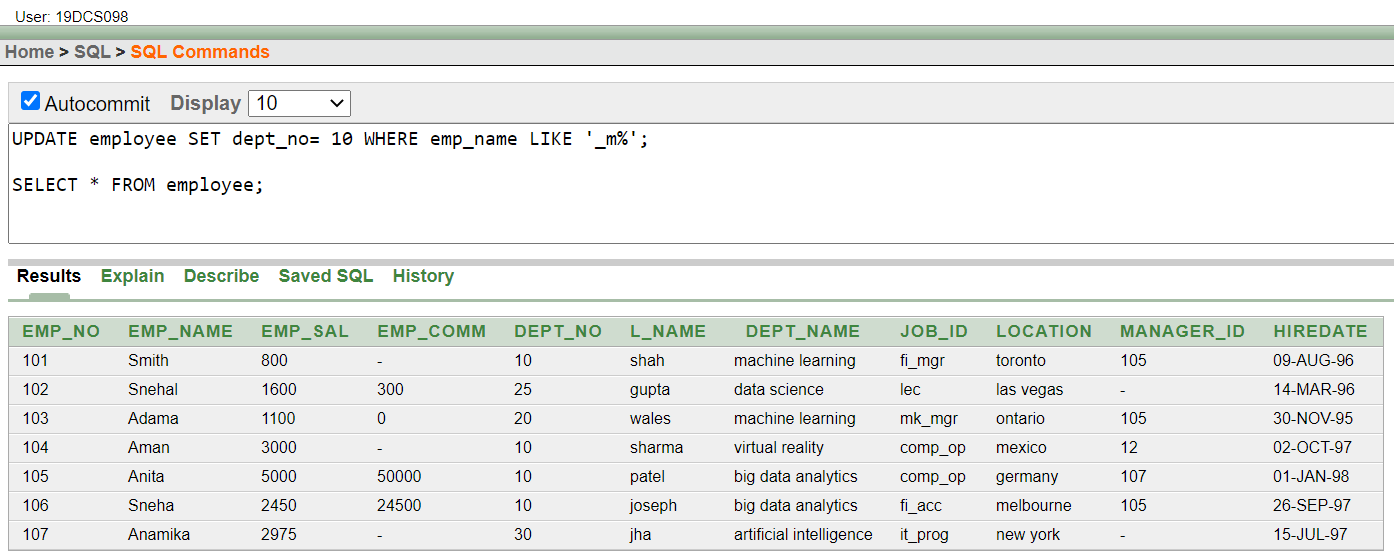
1. **Rename the table sup2.**



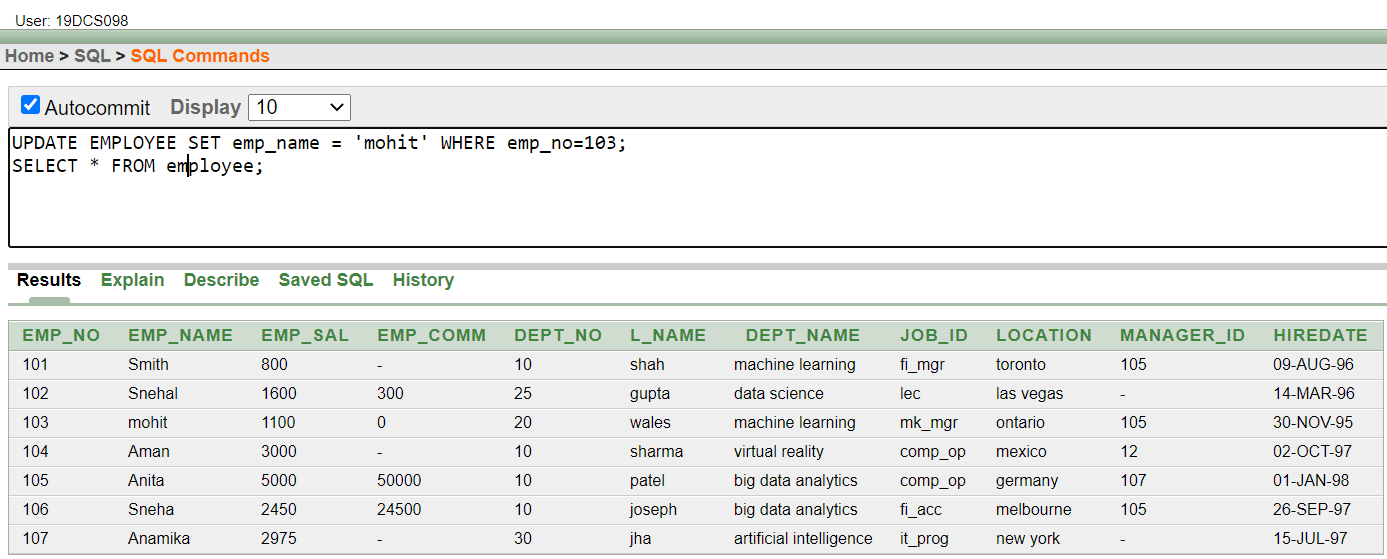
1. **Destroy table sup1 with all the data.**



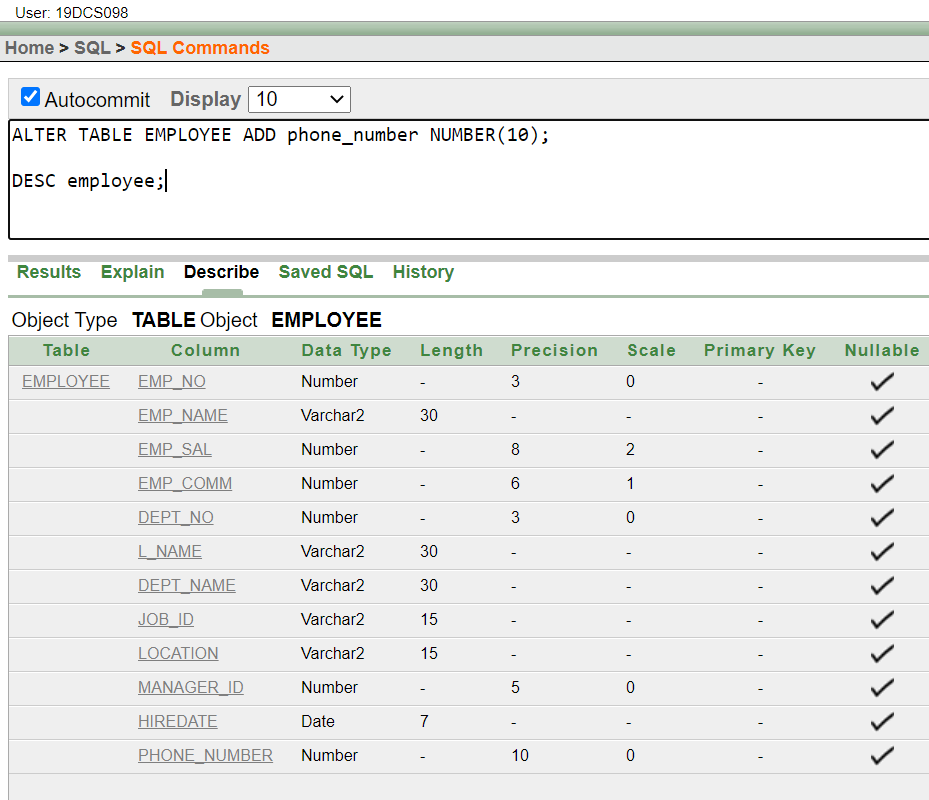
**(14)Update the value dept\_no to 10 where second character of emp.name is ‘m’**



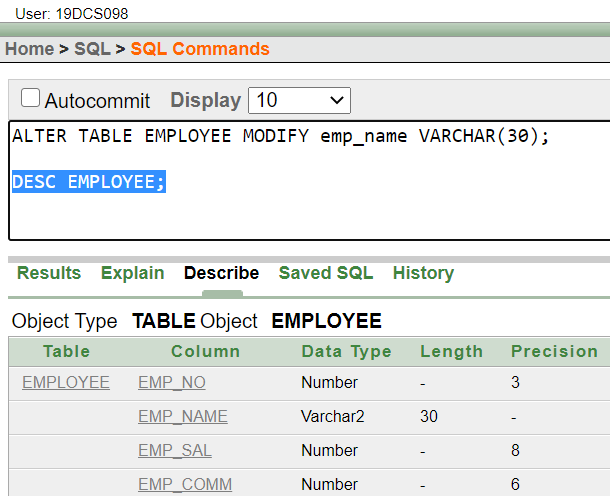
**Update the value of employee name whose employee number is 103.**



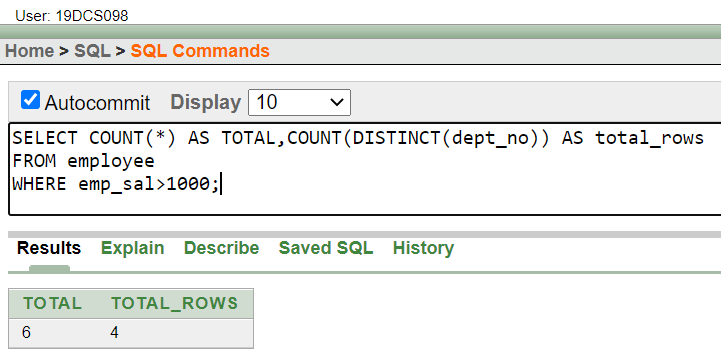
1. **Add one column phone to employee with size of column is 10.**



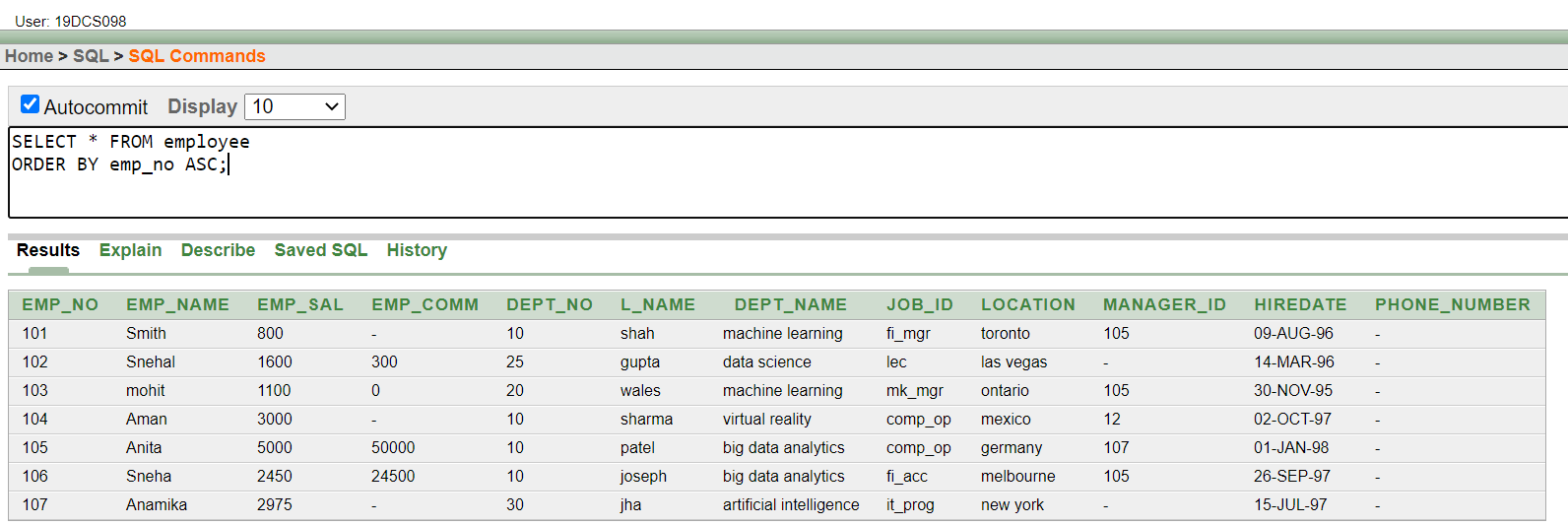
**(15) Modify the column emp\_name to hold maximum of 30 characters**

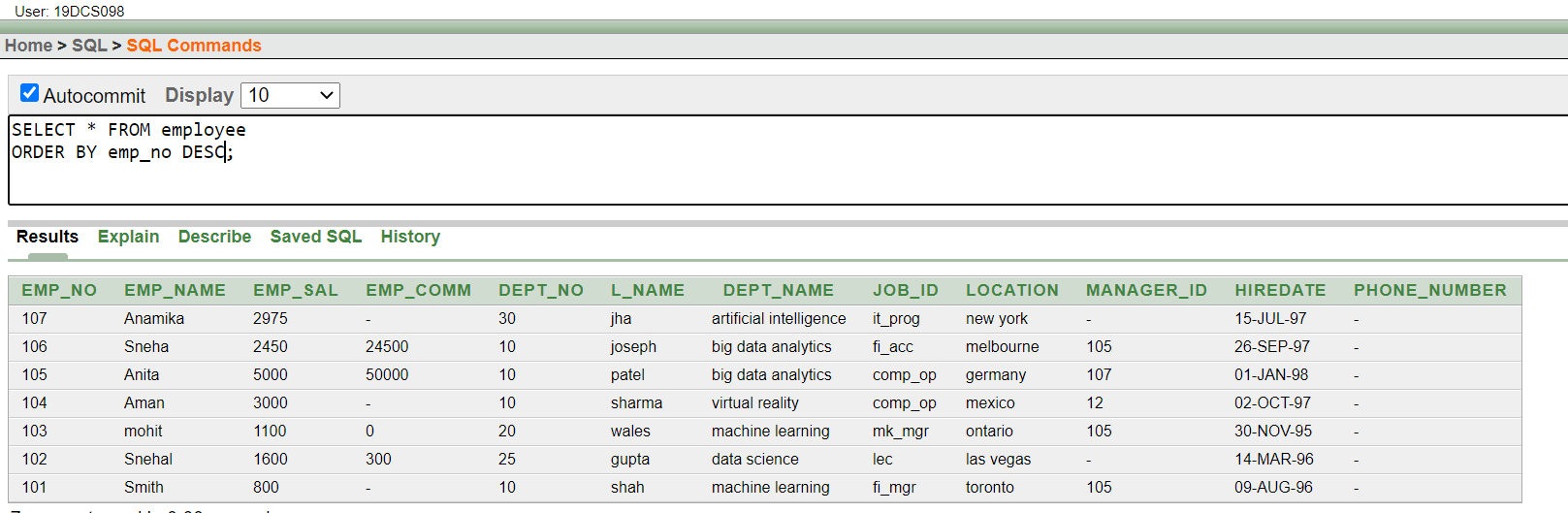


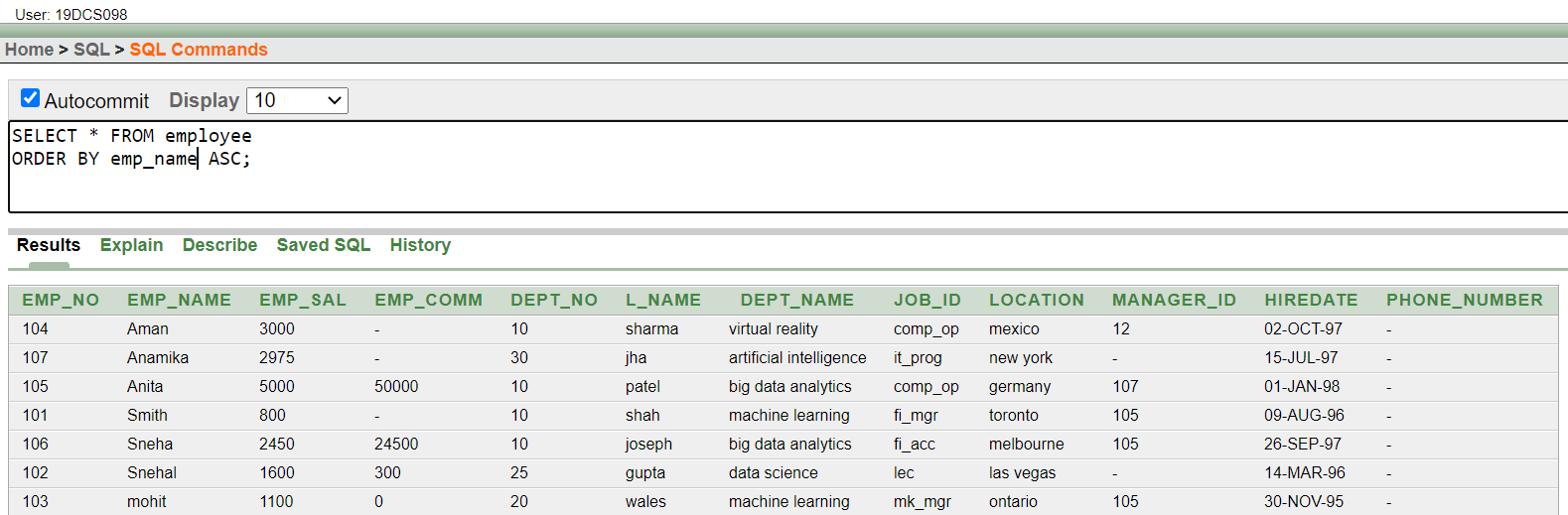
1. **Count the total no as well as distinct rows in dept\_no column with a condition of salary greater than 1000 of employee**

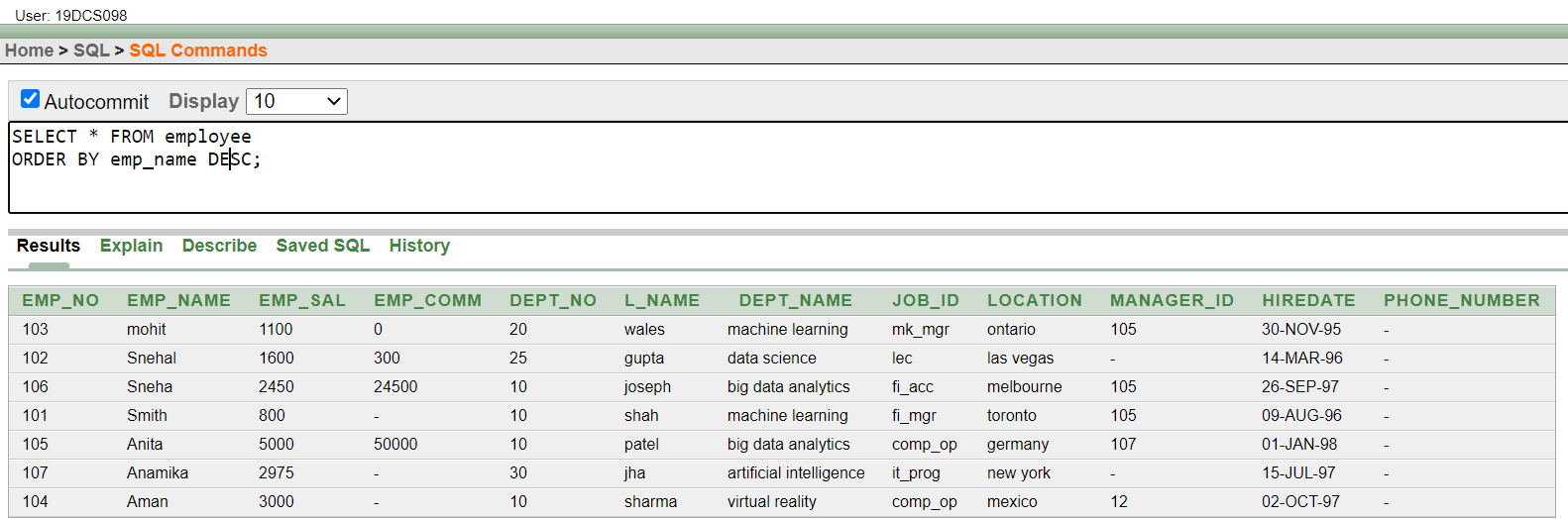


1. **Display the detail of all employees in ascending order, descending order of their name and no**

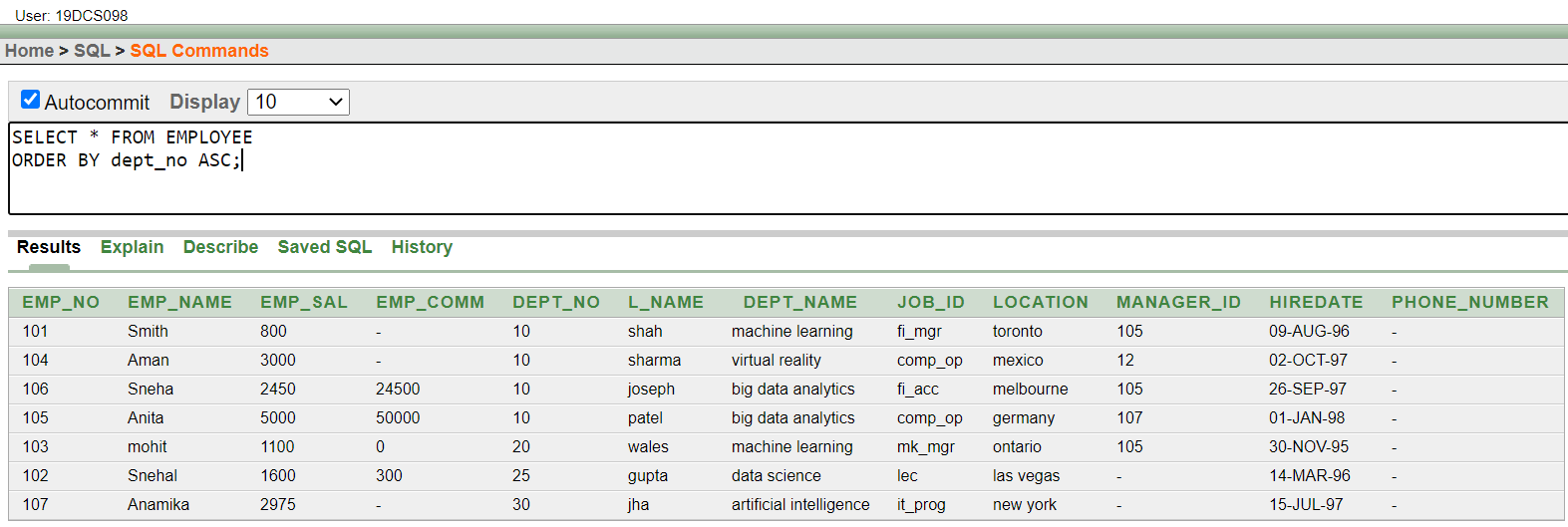


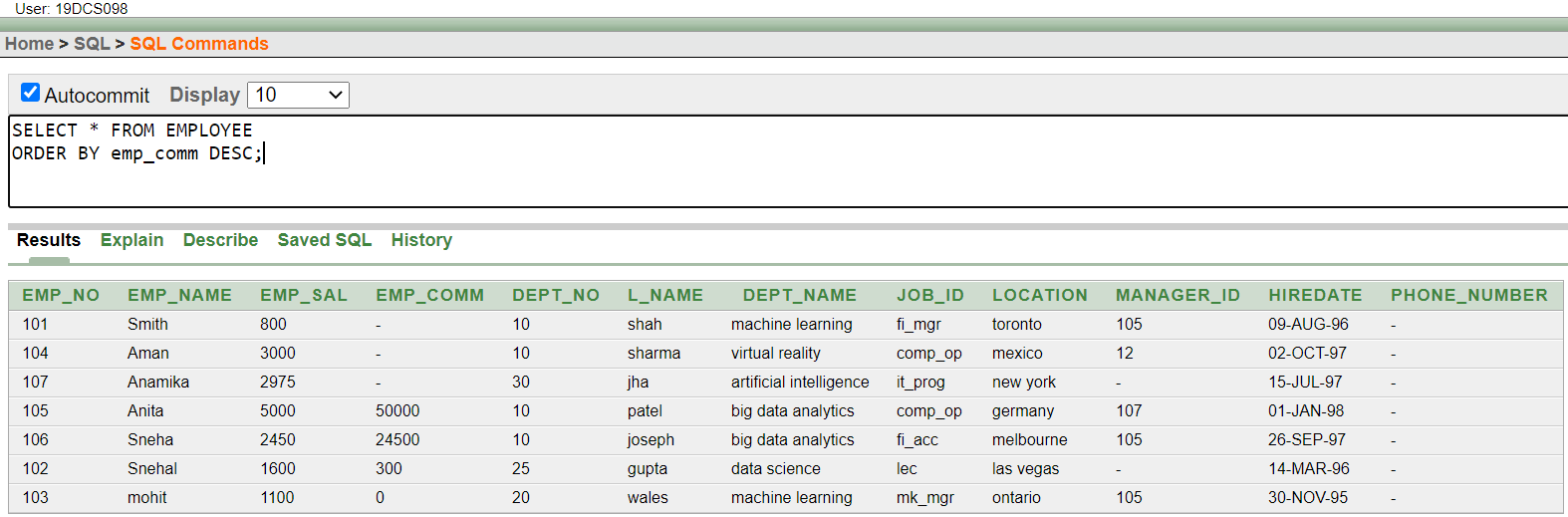




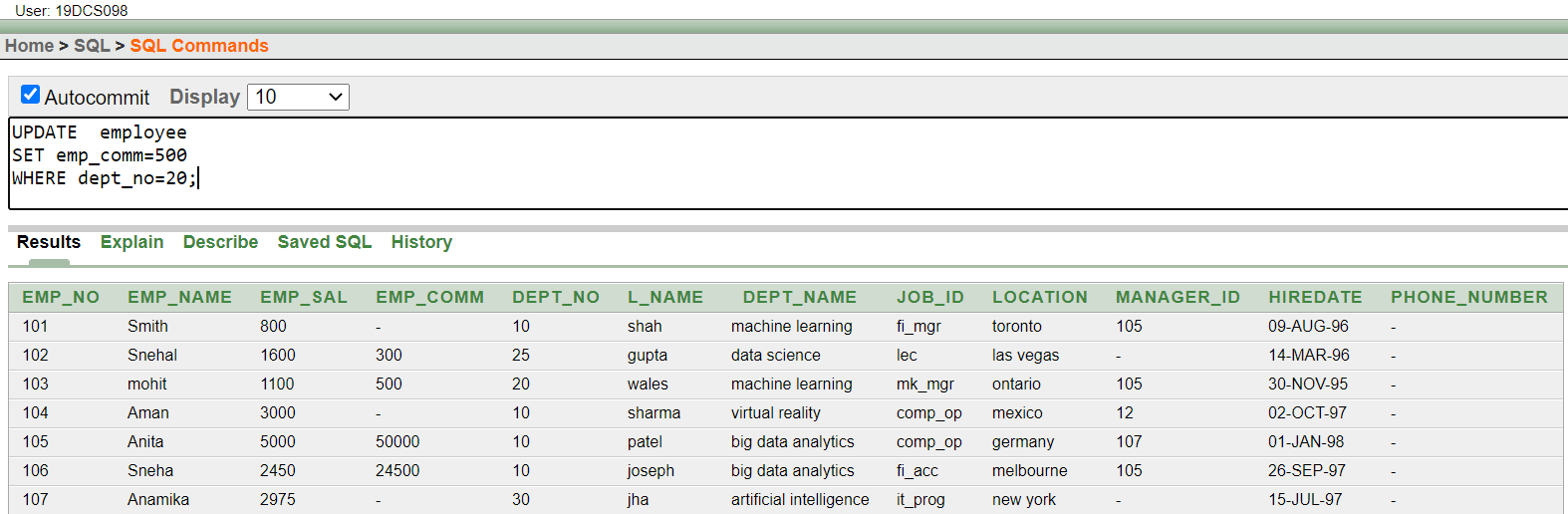


1. **Display the dept\_no in ascending order and accordingly display emp\_comm in descending orde**r

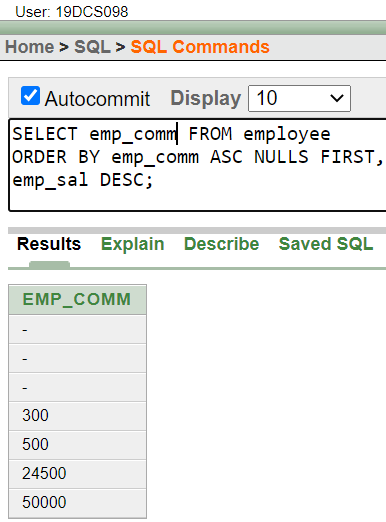




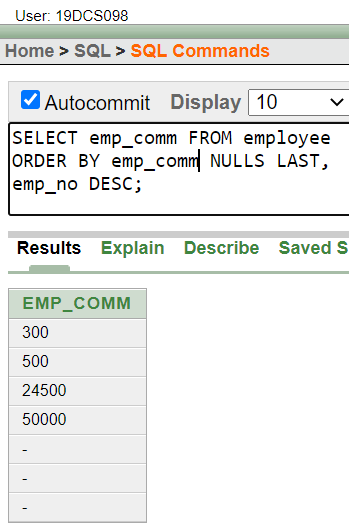
1. **Update the value of emp\_comm to 500 where dept\_no is 20.**



1. **Display the emp\_comm in ascending order with null value first and accordingly sort employee salary in descending order**



1. **Display the emp\_comm in ascending order with null value last and accordingly sort emp\_no in descending order**



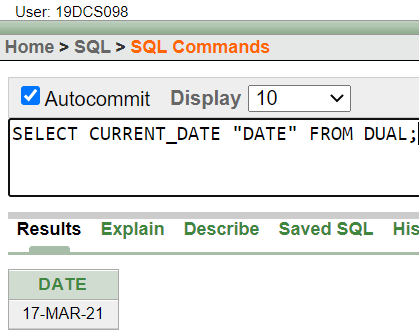
**CONCLUSION:**

In the above practical, we learned the various data manipulation commands and aggregate functions.

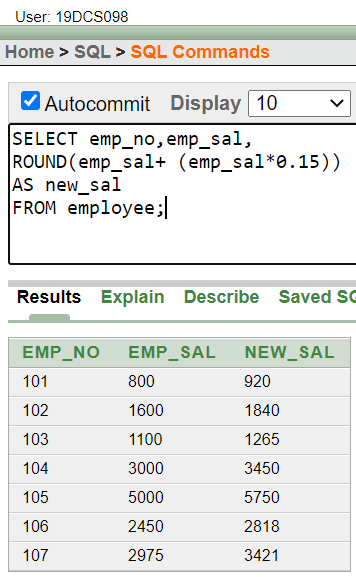
**PRACTICAL-6**

**To study Single-row functions.**

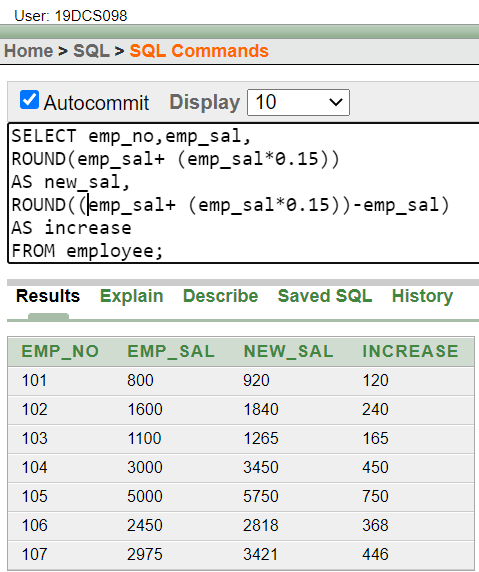
1. **Write a query to display the current date. Label the column Date**



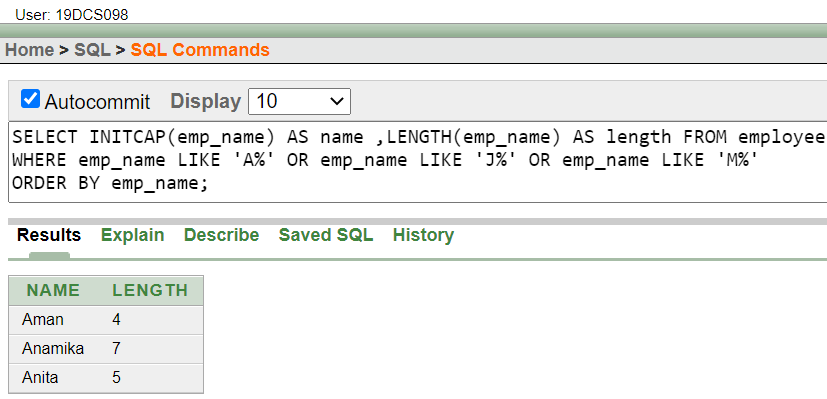
1. **For each employee, display the employee number, salary, and salary increased by 15% and expressed as a whole number. Label the column New Salary**



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

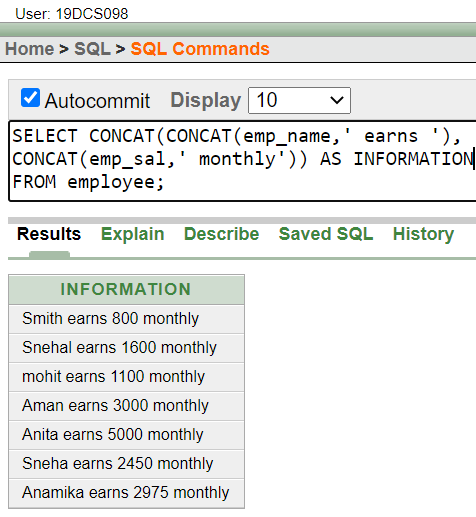


1. **Write a query that displays the employee’s names with the first letter capitalized and all other letters lowercase, and the length of the names, for all employees whose name starts with J, A, or M. Give each column an appropriate label. Sort the results by the employees’ last names.**

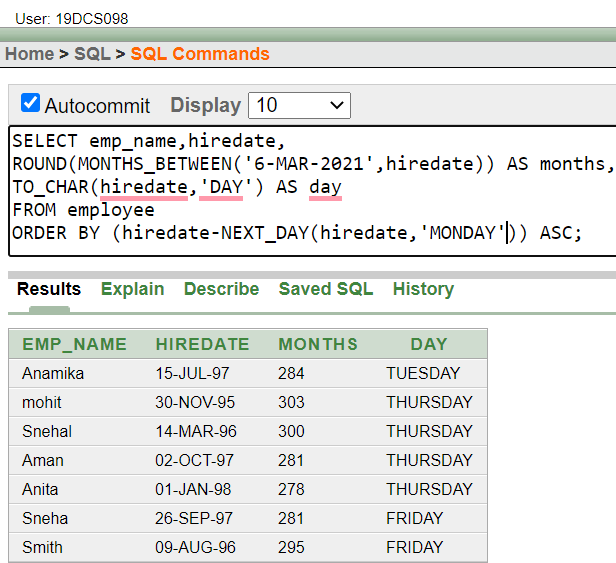


1. **Write a query that produces the following for each employee:**

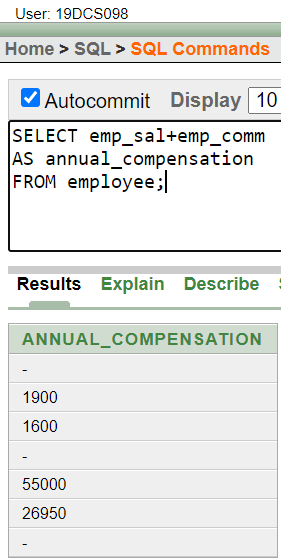
**earns monthly**



1. **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**



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



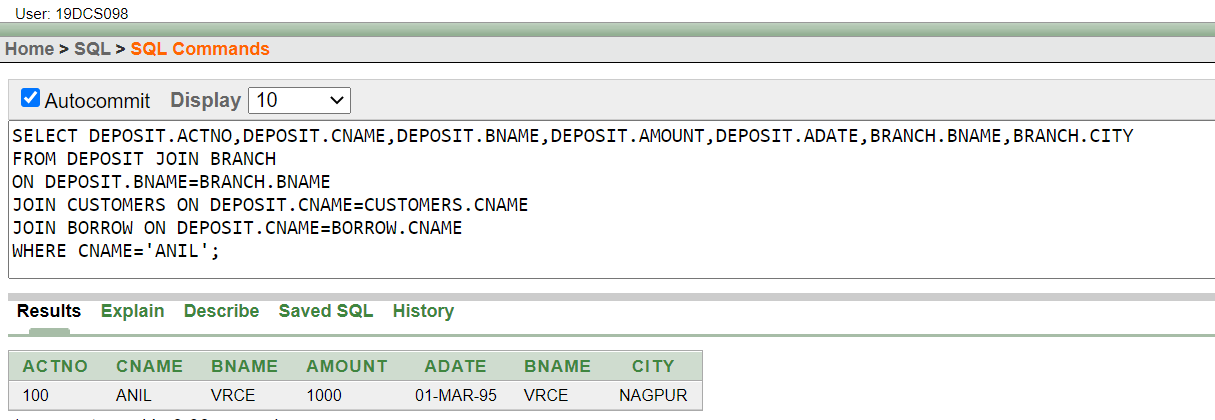
**CONCLUSION:**

In the above practical, we can learned the concept and application of single row functions.

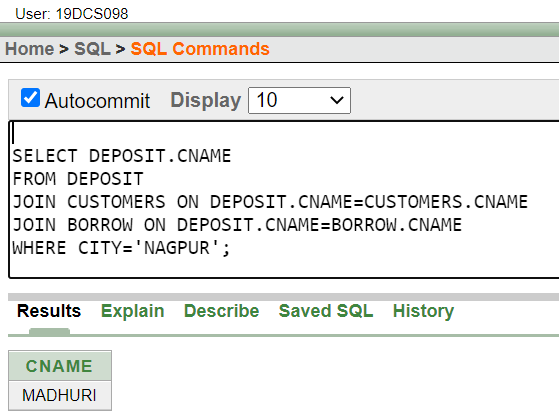
**PRACTICAL-7**

**Displaying data from Multiple Tables (join)**

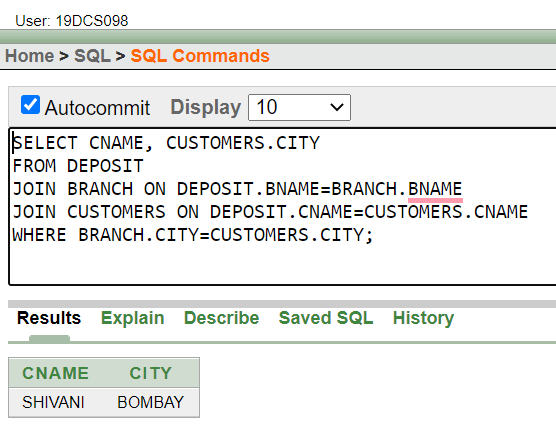
**(1)Give details of customers ANIL.**



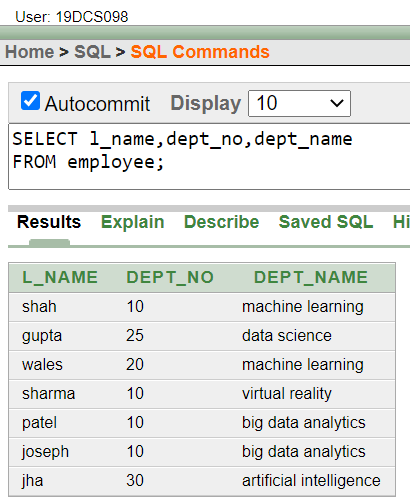
**(2)Give name of customer who are borrowers and depositors and having living city Nagpur**



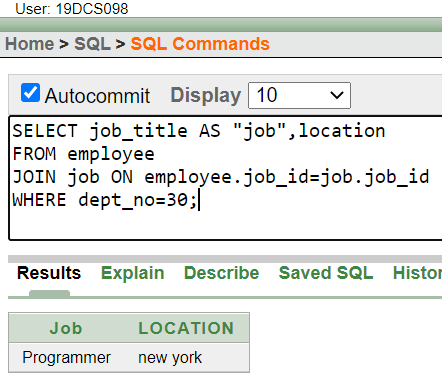
3) **Give city as their city name of customers having same living branch.**



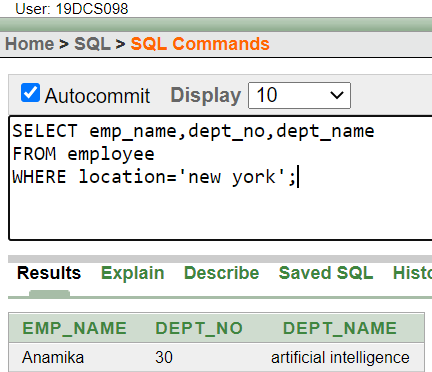
**4) Write a query to display the last name, department number, and department name for all employees**



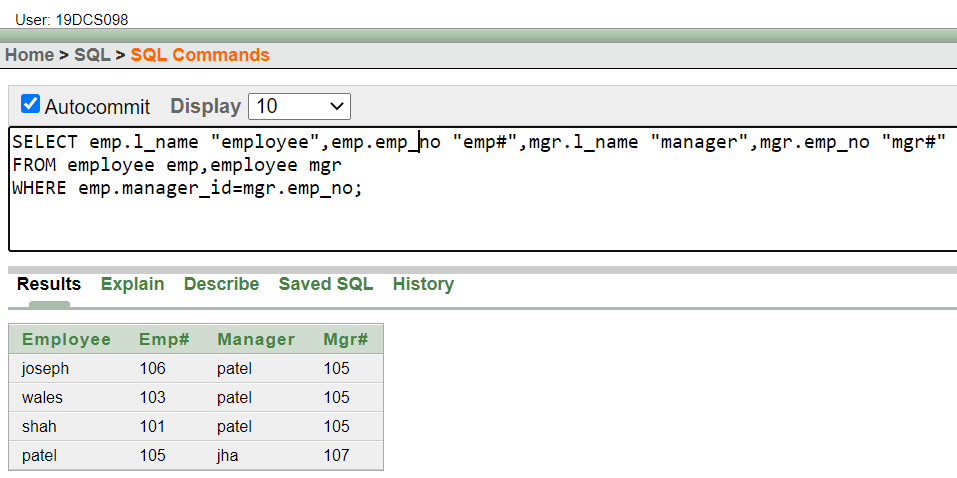
**5) Create a unique listing of all jobs that are in department 30. Include the location of the department in the output**



**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.**



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



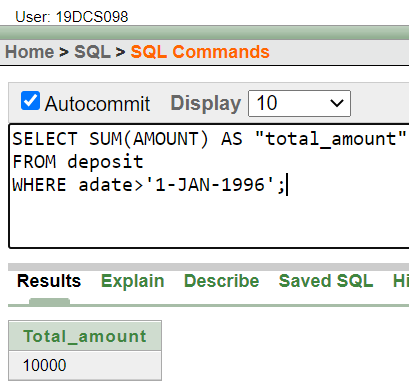
**CONCLUSION:**

In the above Practical, we learned the concept of JOINS.

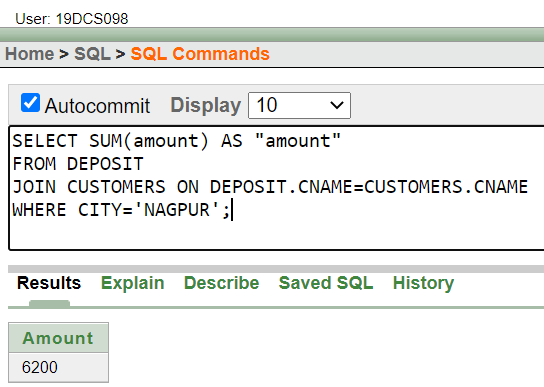
**Practical-8**

**To apply the concept of Aggregating Data using Group functions.**

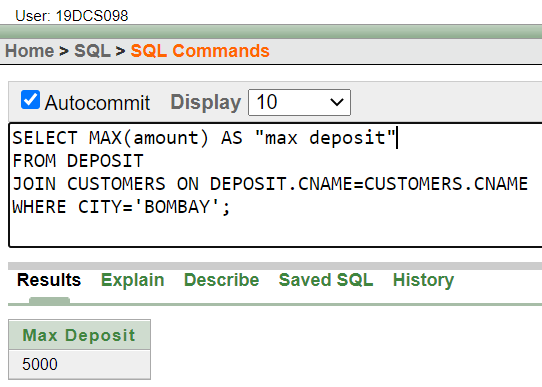
1. **List total deposit of customer having account date after 1-jan-96.**



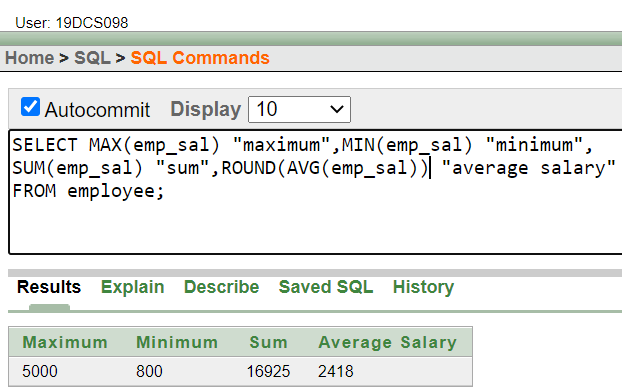
1. **List total deposit of customers living in city Nagpur.**



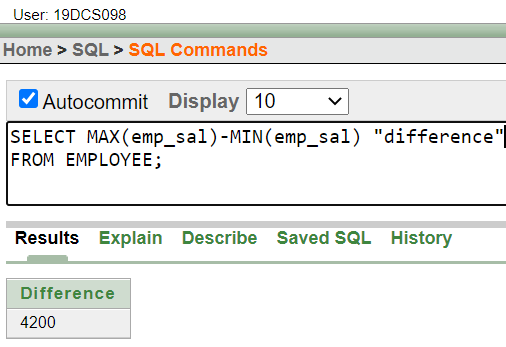
1. **List maximum deposit of customers living in bombay.**



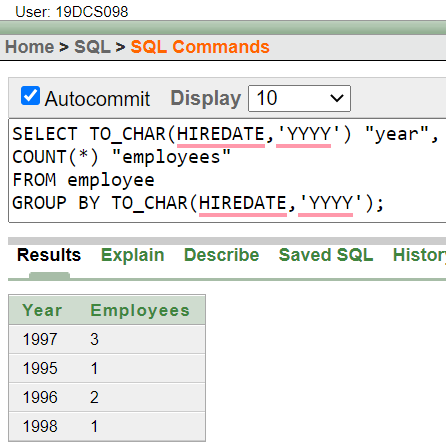
1. **Display the highest, lowest, sum, and average salary of all employees. Label the columns Maximum, Minimum, Sum, and Average, respectively. Round your results to the nearest whole number.**



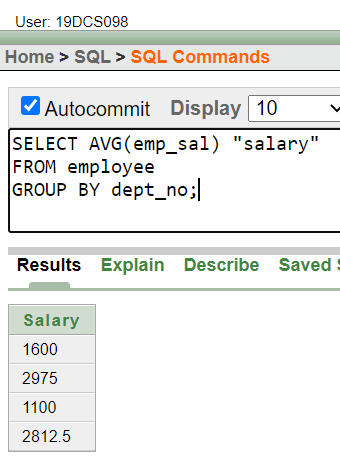
1. **Write a query that displays the difference between the highest and lowest salaries. Label the column DIFFERENCE.**



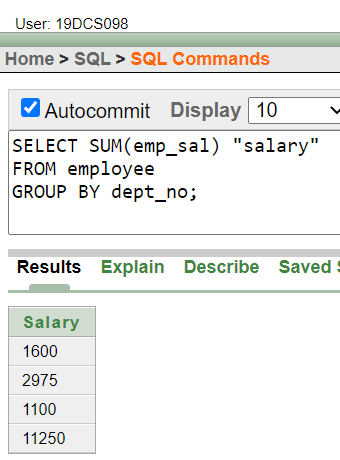
1. **Create a query that will display the total number of employees and, of that total, the number of employees hired in 1995, 1996, 1997, and 1998**



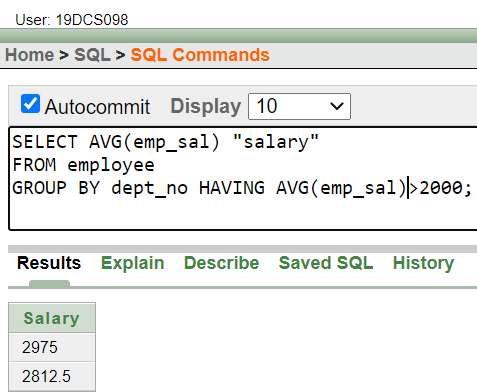
1. **Find the average salaries for each department without displaying the respective department numbers.**



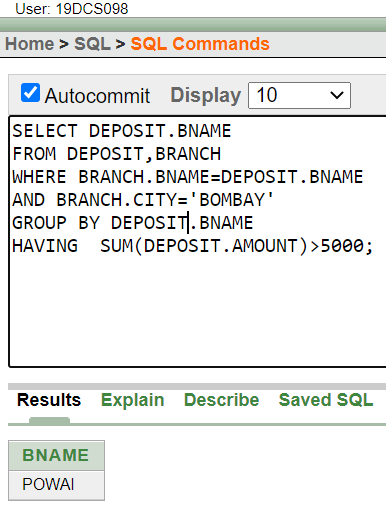
1. **Write a query to display the total salary being paid to each job title, within each department.**



1. **Find the average salaries > 2000 for each department without displaying the respective department numbers.**



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



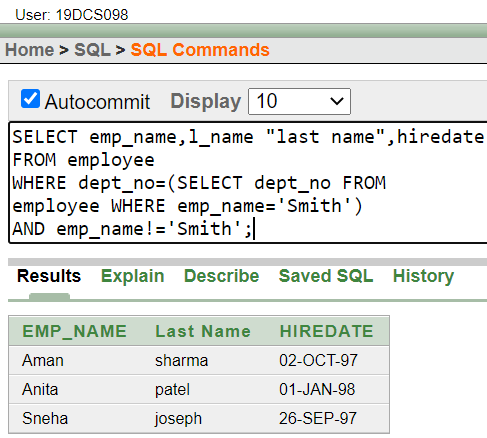
**CONCLUSION:**

In the above practical, we learned to aggregate data by using GROUP BY.

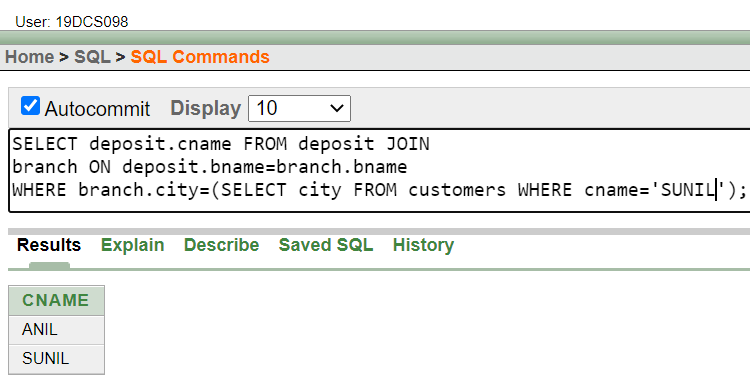
**PRACTICAL-9**

**To solve queries using the concept of sub query**

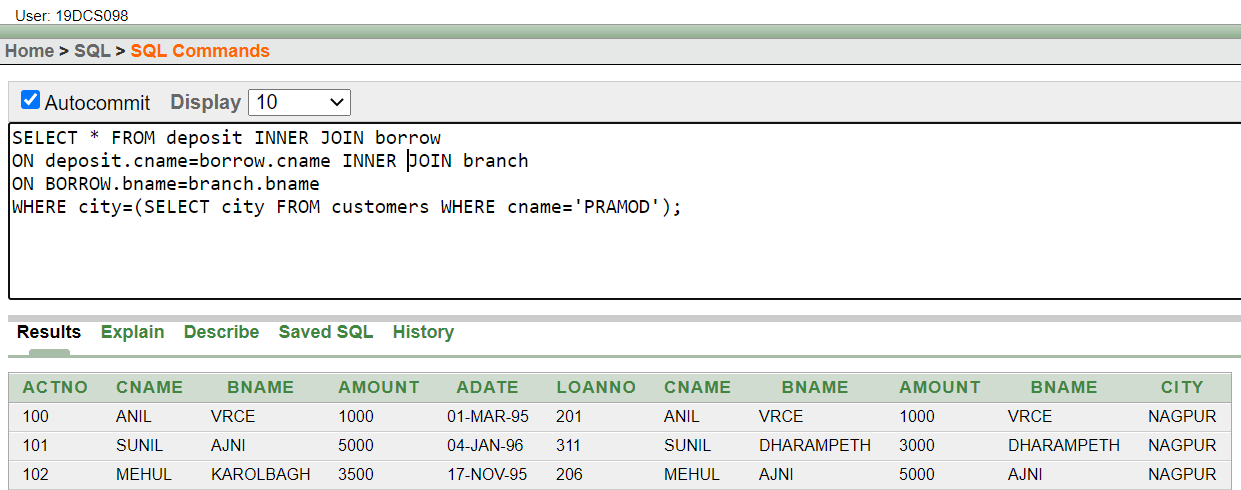
1. **Write a query to display the last name and hire date of any employee in the same department as smith. Exclude smith**



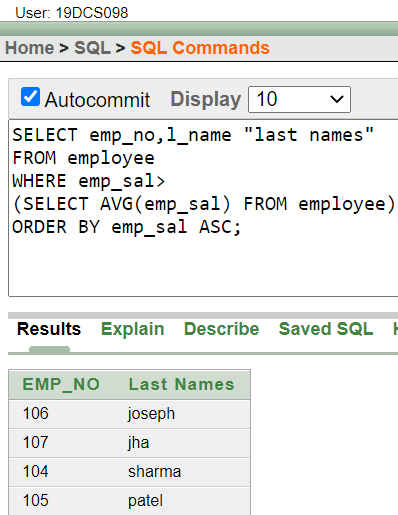
1. **Give name of customers who are depositors having same branch city of mr. sunil.**



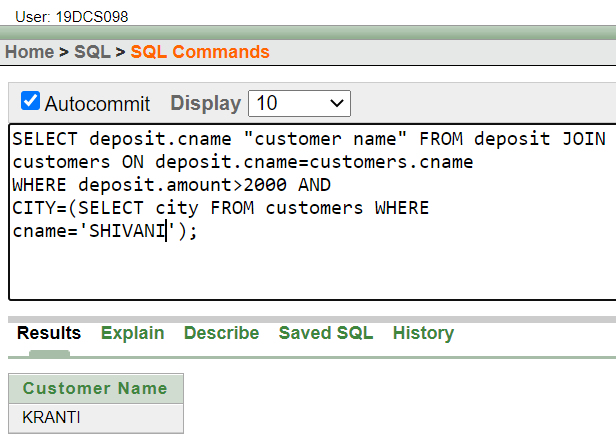
1. **Give deposit details and loan details of customer in same city where pramod is living**



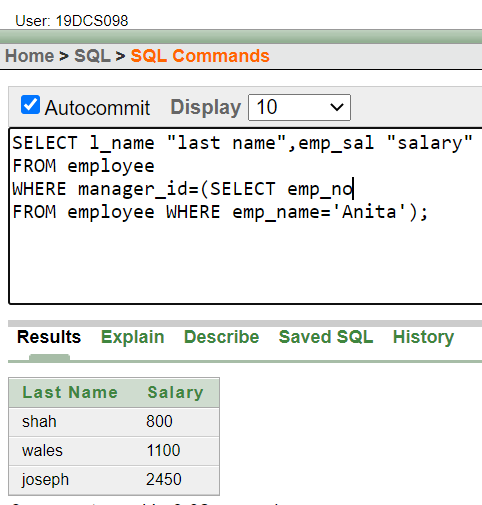
1. **Create a query to display the employee numbers and last names of all employees who earn more than the average salary. Sort the results in ascending order of salary.**



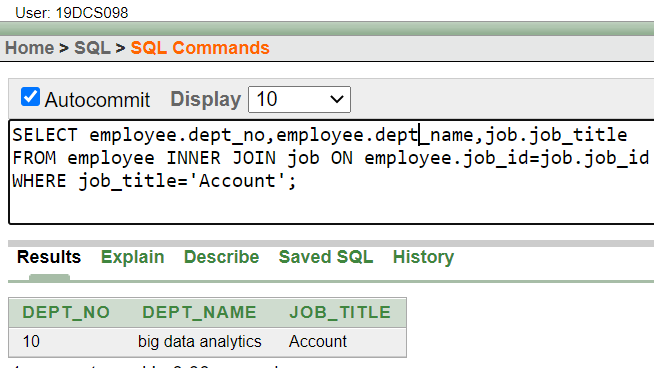
1. **Give names of depositors having same living city as mr. anil and having deposit amount greater than 2000**



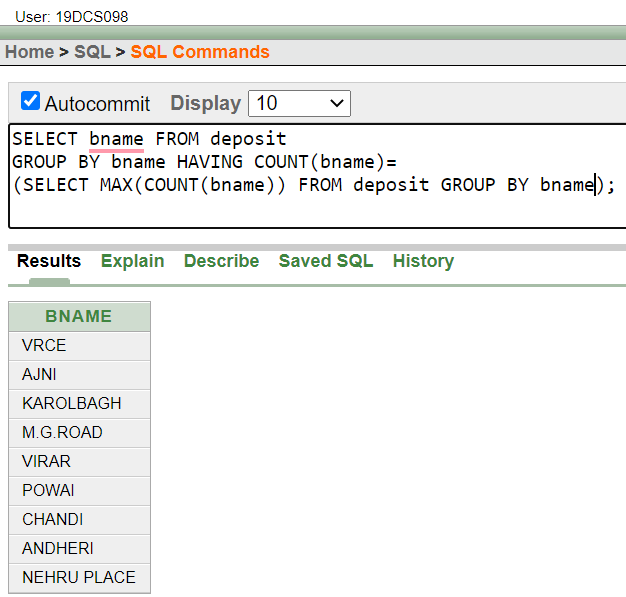
1. **Display the last name and salary of every employee who reports to ford.**



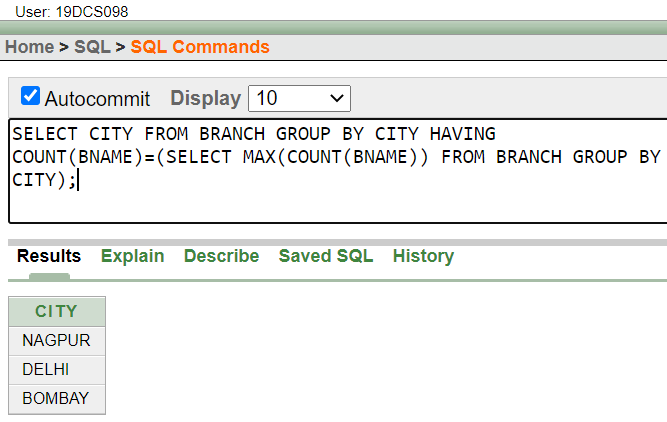
1. **Display the department number, name, and job for every employee in the Accounting department**.



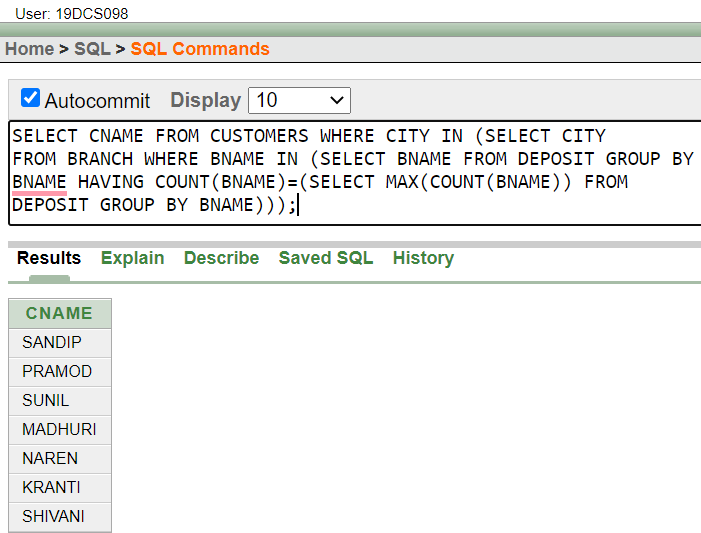
1. **List the name of branch having highest number of depositors**.



1. **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.**



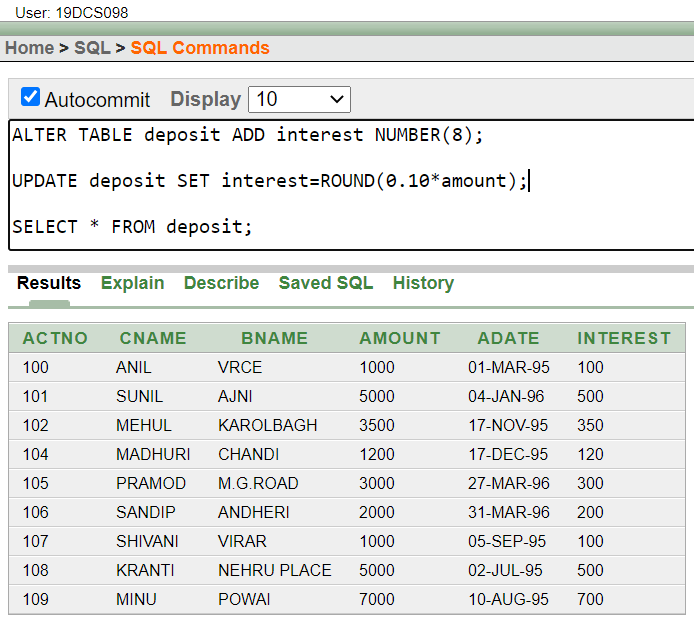
**CONCLUSION:**

We learned the concept of sub query.

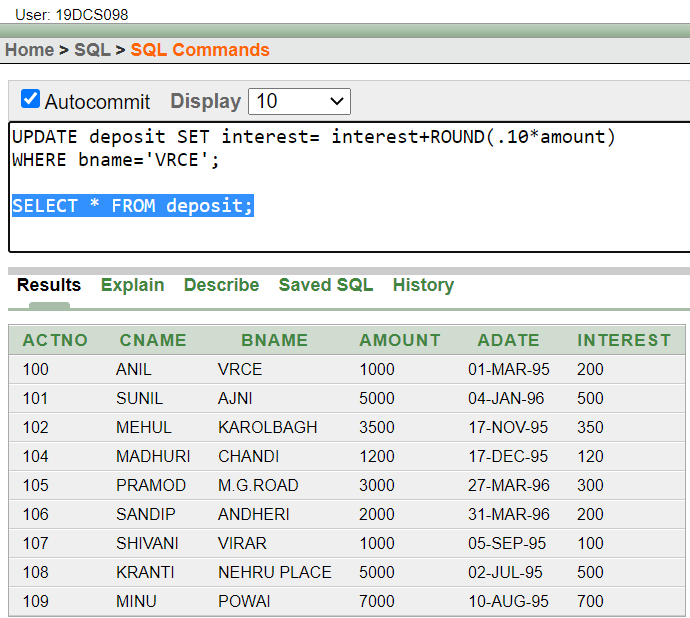
**PRACTICAL-10**

**Manipulating Data**

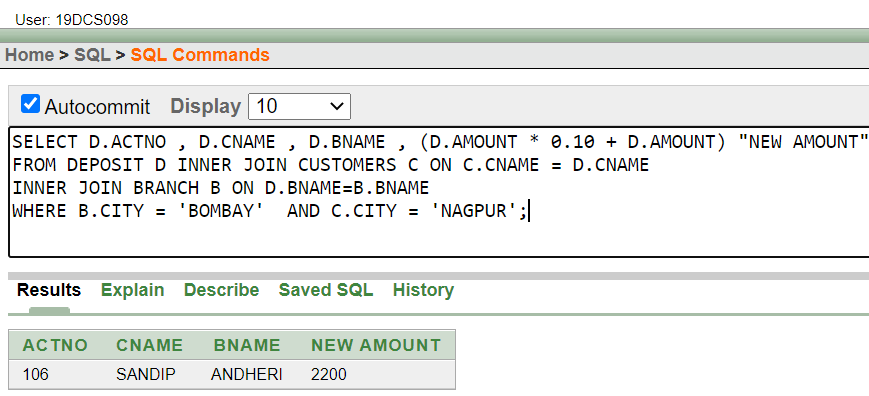
1. **Give 10% interest to all depositors**



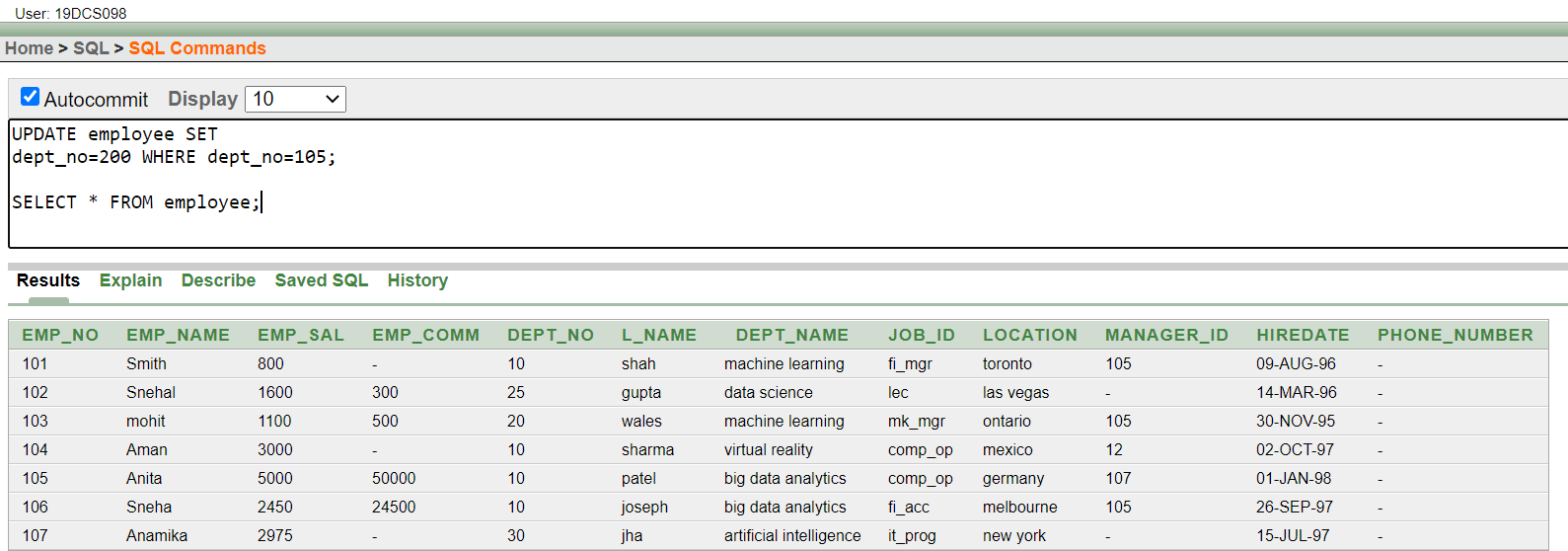
1. **Give 10% interest to all depositors having branch vrce**



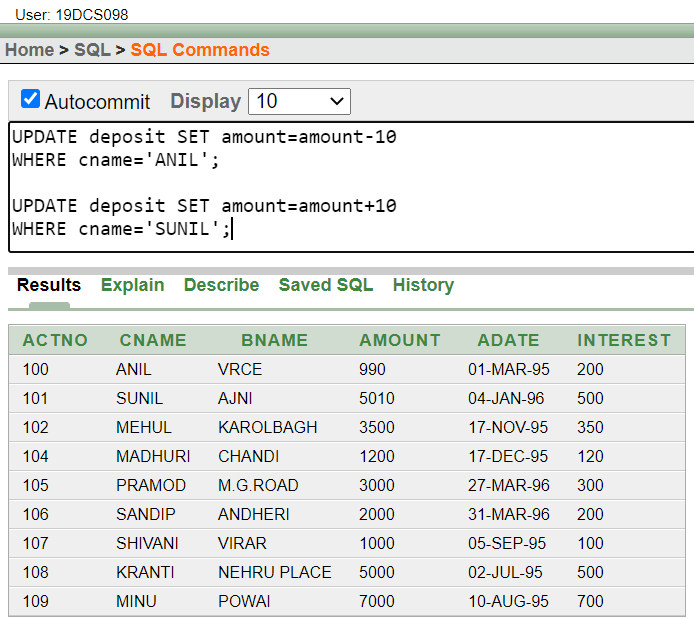
1. **Give 10% interest to all depositors living in nagpur and having branch city Bombay**



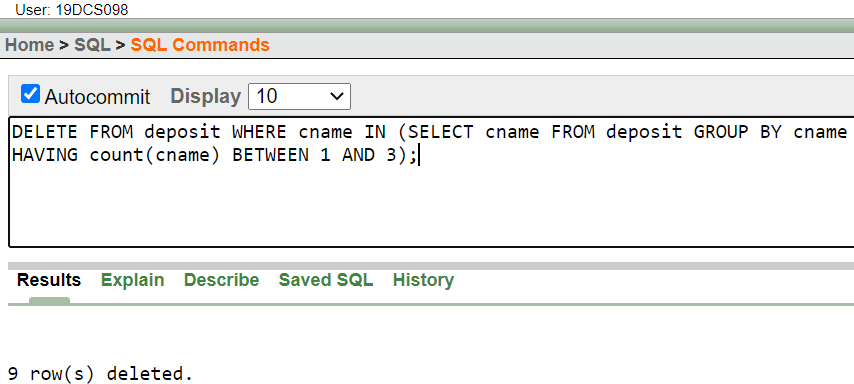
1. **Write a query which changes the department number of all employees with empno 7788’s job to employee 7844’current department number.**



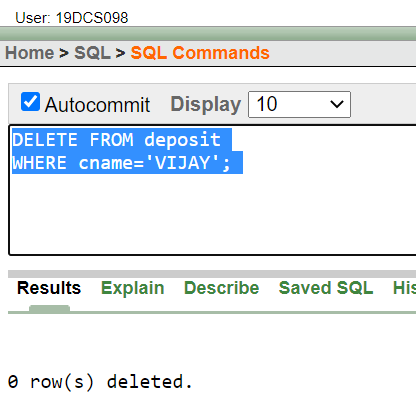
1. **Transfer 10 Rs from account of anil to sunil if both are having same branch.**



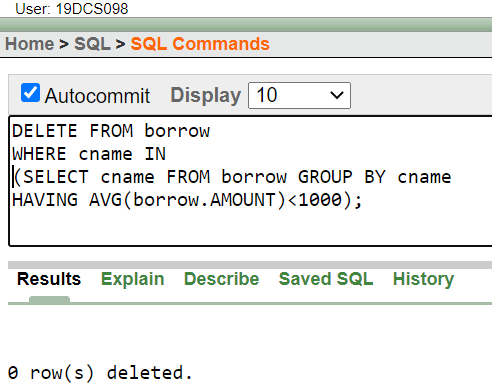
1. **Delete depositors of branches having number of customers between 1 to 3.**



1. **Delete deposit of vijay.**



1. **Delete borrower of branches having average loan less than 1000**



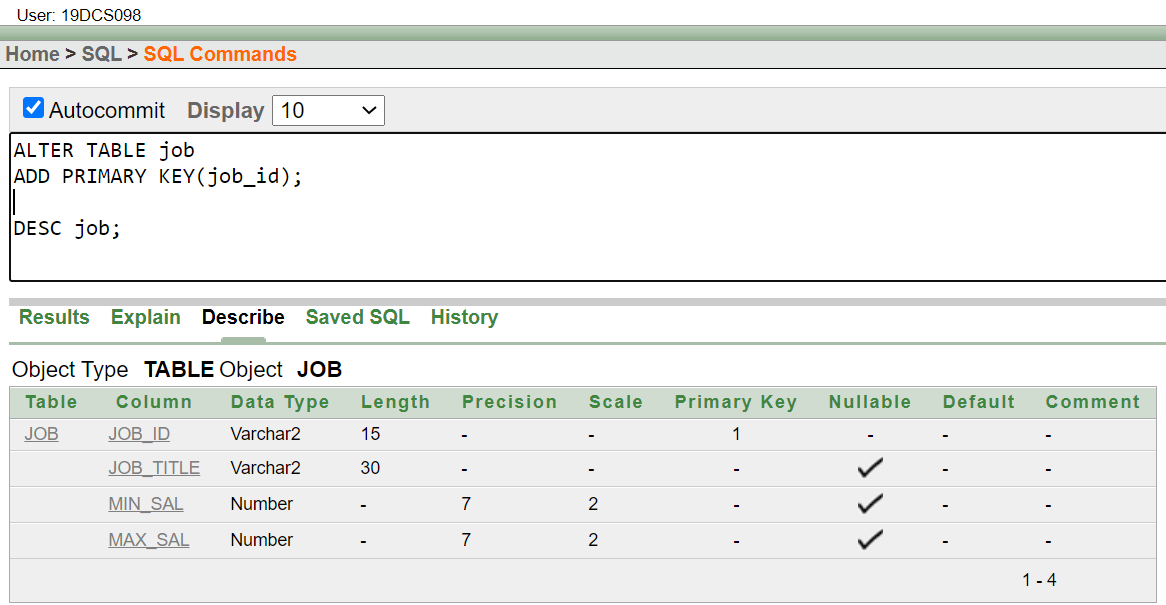
**CONCLUSION:**

In the above practical, we learned the concept of data manipulation.

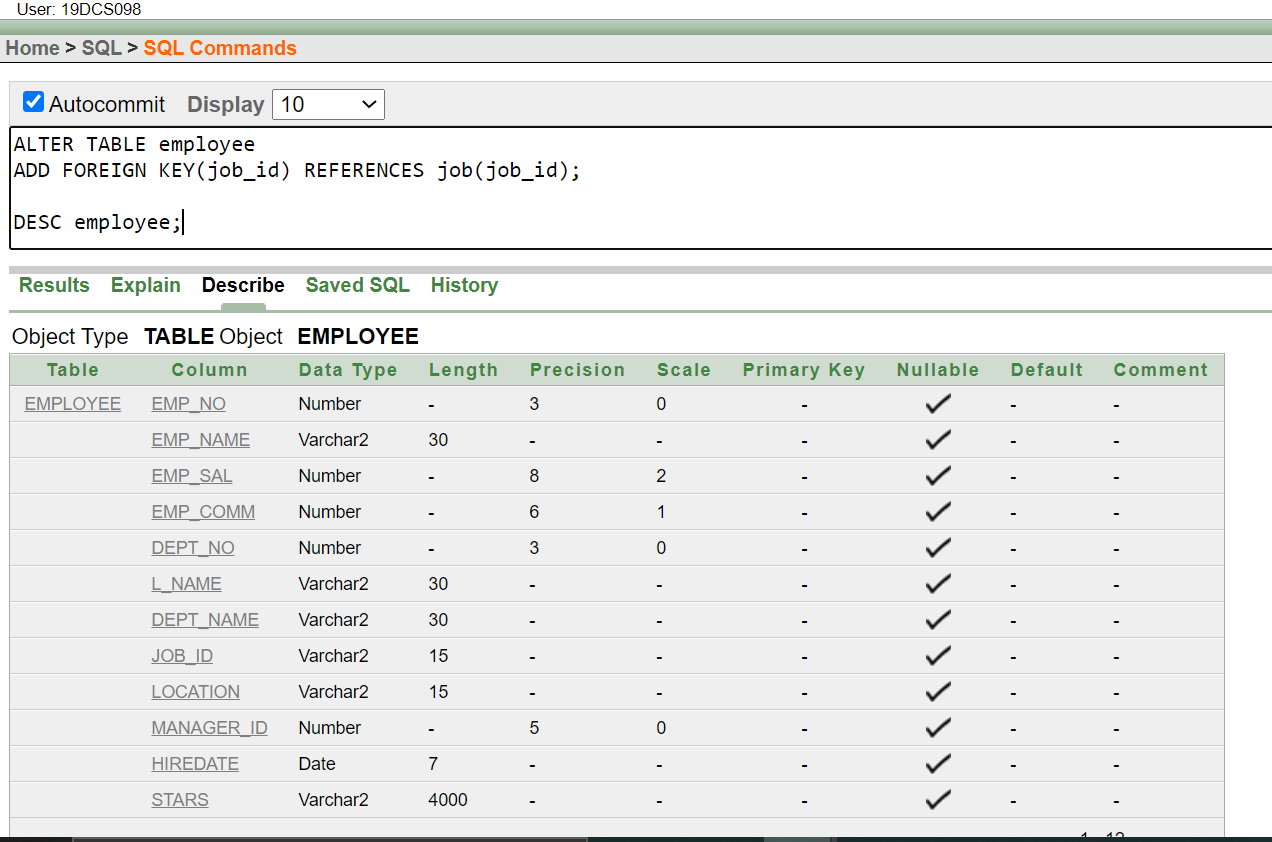
**PRACTICAL-11**

**Add and Remove constraint**

**(1)Add primary key constraint on job\_id in job table**.



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



(**4)Remove primary key constraint on job\_id**

ALTER TABLE job DROP CONSTRAINT job\_id ;

(**5)Remove foreign key constraint on employee table**

ALTER TABLE EMPLOYEE DROP CONSTRAINT job\_id ;

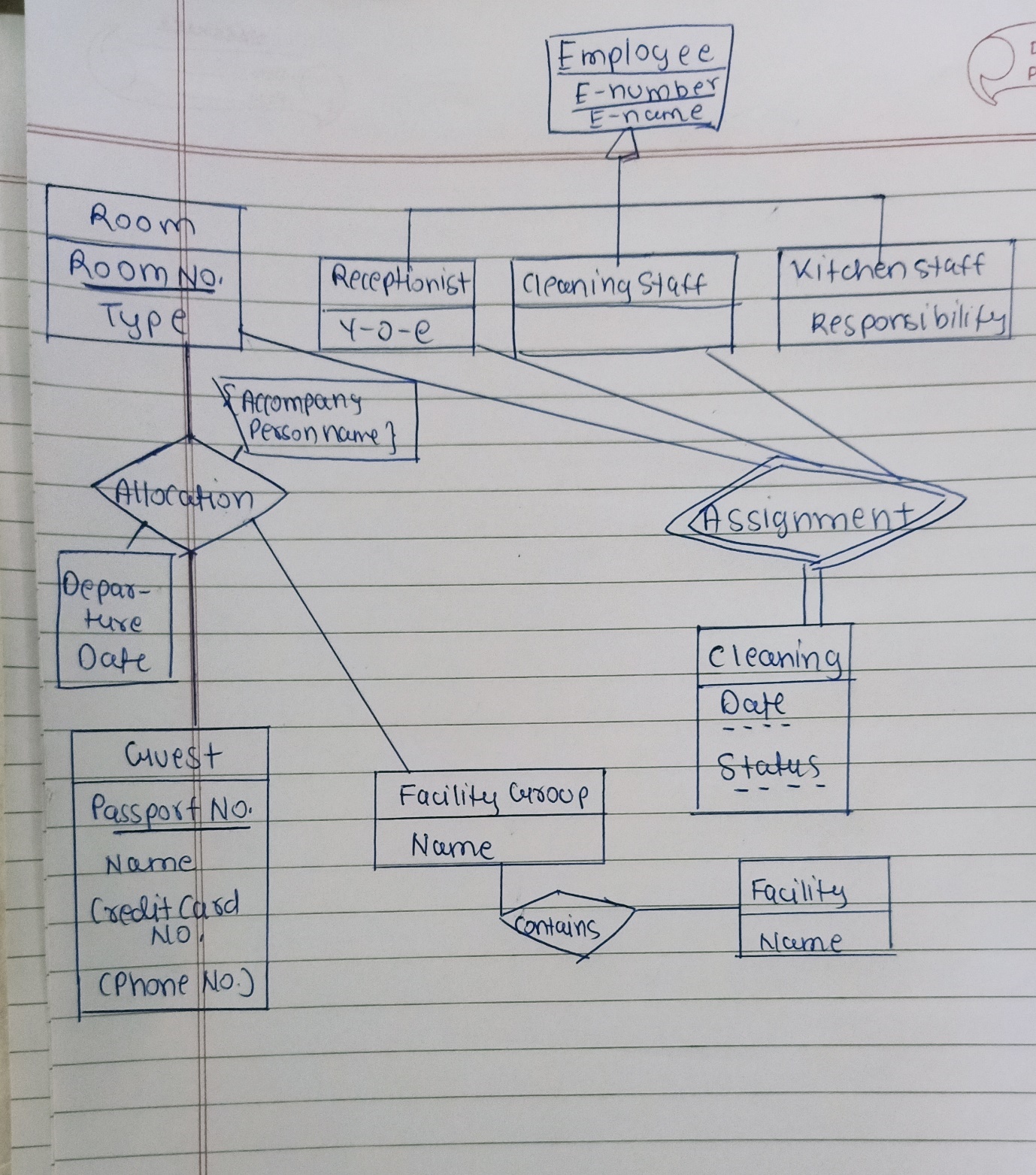
**CONCLUSION:**

In the above practical, we learned how to add and remove constraints.

**PRACTICAL-12**

**Data Dictionary and E-R Diagram**

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

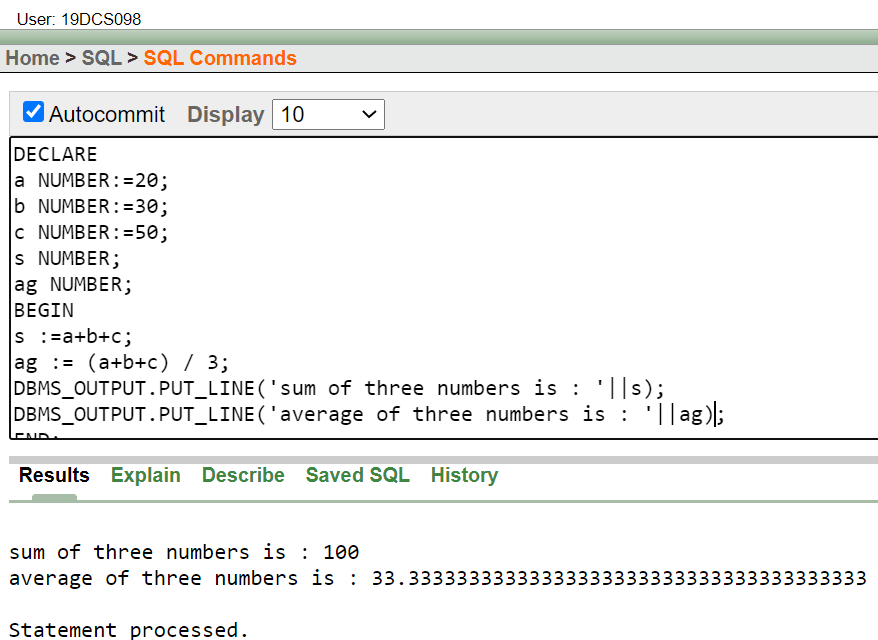


**CONCLUSION:**

In the above practical, we learned the concept of E-R diagram.

**PRACTICAL-13**

**Write a PL-SQL block to find Sum and average of three numbers.**

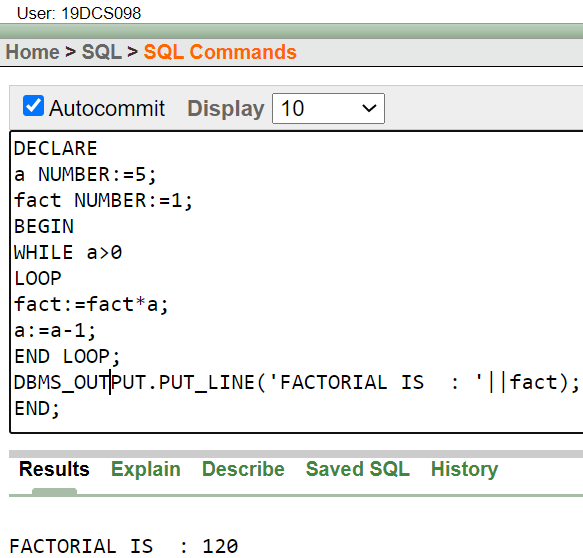


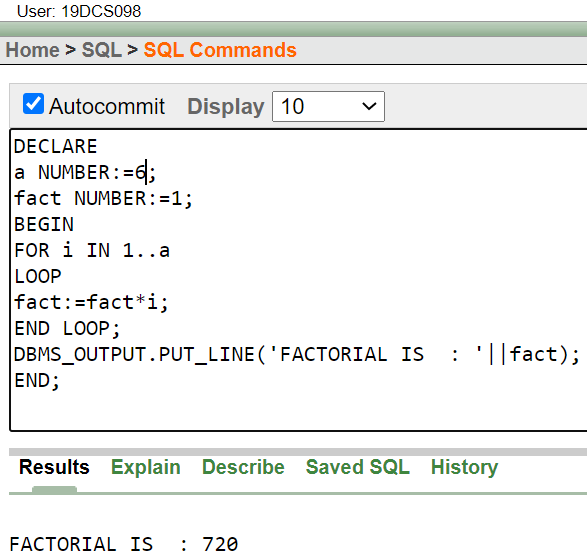
**CONCLUSION:**

In the above practical, we learned the concept of PL/SQL

**PRACTICAL-14**

**Find the factorial of a number in pl/sql using for, While and Simple Loop**





**CONCLUSION:**

In the above practical, we learned the concept of loops using PL/SQL

**PRACTICAL-15**

**To understand the concept of “select into” and “% type” attribute.**

Create an EMPLOYEES table that is a replica of the EMP table. Add a new column, STARS, of VARCHAR2 data type and length of 50 to the EMPLOYEES table for storing asterisk (\*). Create a PL/SQL block that rewards an employee by appending an asterisk in the STARS column for every Rs1000/- of the employee’s salary. For example, if the employee has a salary amount of Rs8000/-, the string of asterisks should contain eight asterisks. If the employee has a salary amount of Rs12500/-, the string of asterisks should contain 13 asterisks. Update the STARS column for the employee with the string of asterisks.

**PROGRAM CODE:**

DECLARE

EMPS\_NO EMPLOYEE.EMP\_NO%TYPE;

EMPS\_NAME EMPLOYEE.EMP\_NAME%TYPE;

EMPS\_SAL EMPLOYEE.EMP\_SAL%TYPE;

EMPS\_COMM EMPLOYEE.EMP\_COMM%TYPE;

DEPT\_NO1 EMPLOYEE.DEPT\_NO%TYPE;

EMPS\_STAR VARCHAR2(50);

i number;

BEGIN

for i in 101..107

loop

select NVL( ROUND (EMP\_SAL/1000),0) INTO EMPS\_SAL FROM EMPLOYEE WHERE EMP\_NO=i;

EMPS\_STAR:=NULL;

FOR J IN 1..EMPS\_SAL

LOOP

EMPS\_STAR:=EMPS\_STAR || '\*' ;

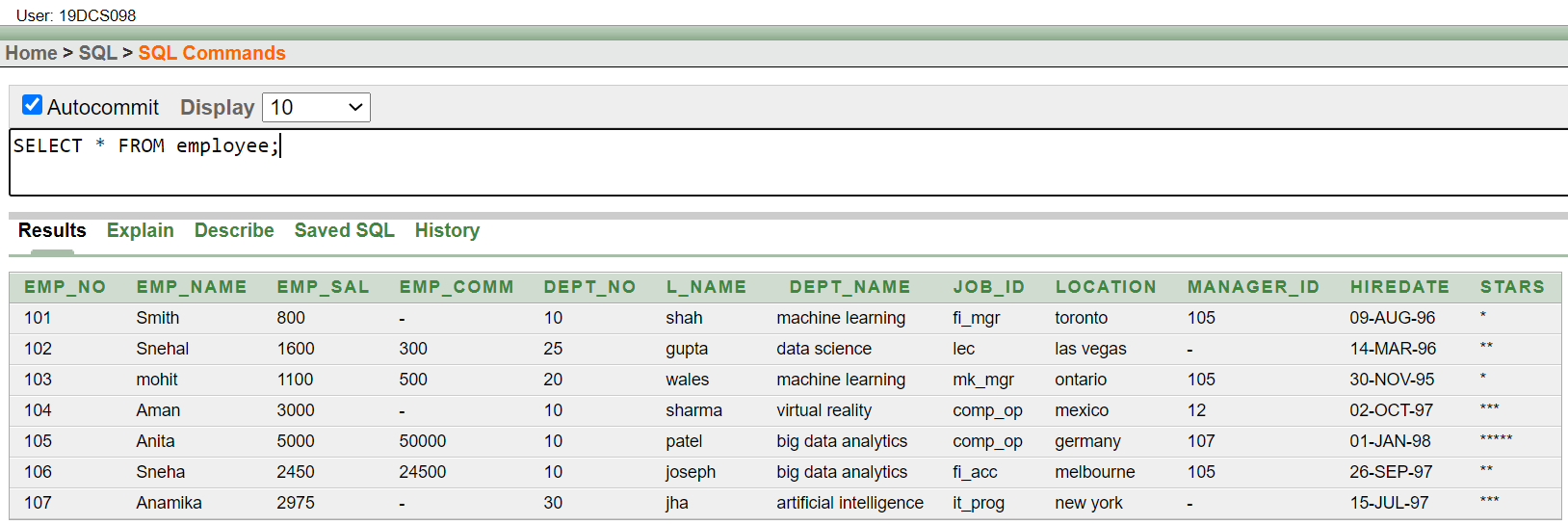
END LOOP;

UPDATE employee SET stars=emps\_star WHERE emp\_no=i;

END LOOP;

END;

**OUTPUT:**



**CONCLUSION:**

In the above practical, we learned the concept of “select into” and “% type” attribute.

**PRACTICAL-16**

**To perform the concept of cursor (a) Display all the information of EMP table using %ROWTYPE.**

DECLARE

employee\_rec employee%rowtype;

CURSOR c\_employee is

SELECT \* FROM employee;

BEGIN

OPEN c\_employee;

LOOP

FETCH c\_employee into employee\_rec;

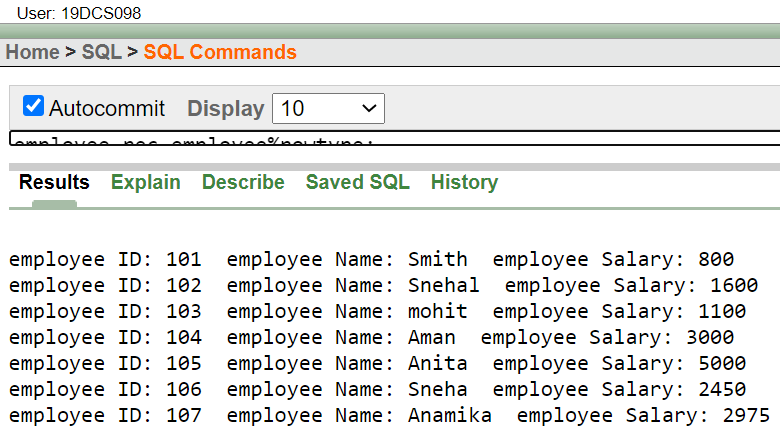
EXIT WHEN c\_employee%notfound;

dbms\_output.put\_line('employee ID: ' || employee\_rec.emp\_no || ' employee Name: ' || employee\_rec.emp\_name || ' employee Salary: ' || employee\_rec.emp\_sal);

END LOOP;

CLOSE c\_employee;

END;

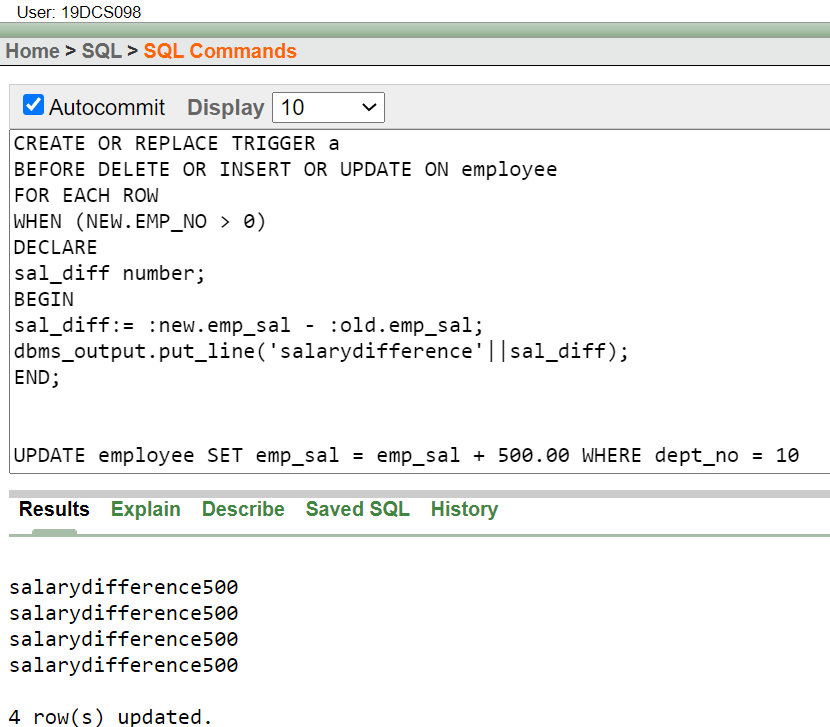


**CONCLUSION:**

In the above practical, we learnt the concept of cursor

**PRACTICAL-17**

**Write a PL/SQL block to update the salary where deptno is 10. Generate trigger that will store the original record in other table before updation take place**



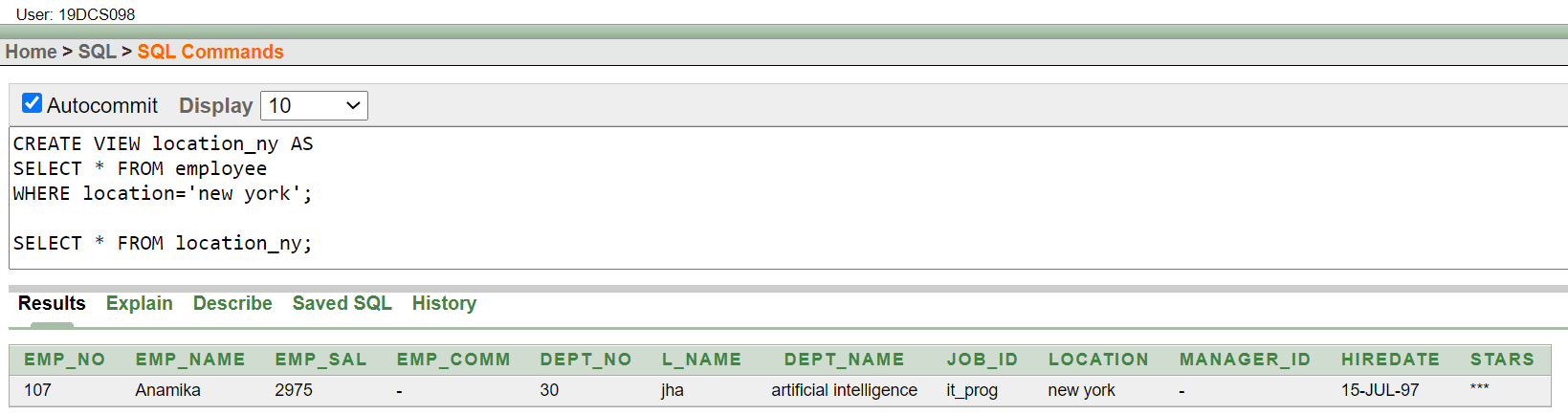
**CONCLUSION:**

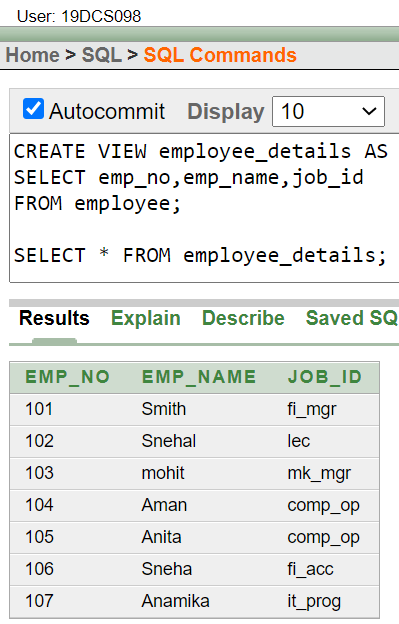
In the above practical, we learned the concept of cursor.

**PRACTICAL-18**

**To solve queries using the concept of View.**

**(1) Write a query to create a view for those employee belongs to the location New York.**





**CONCLUSION:**

In the above practical, we learnt the concept of views.

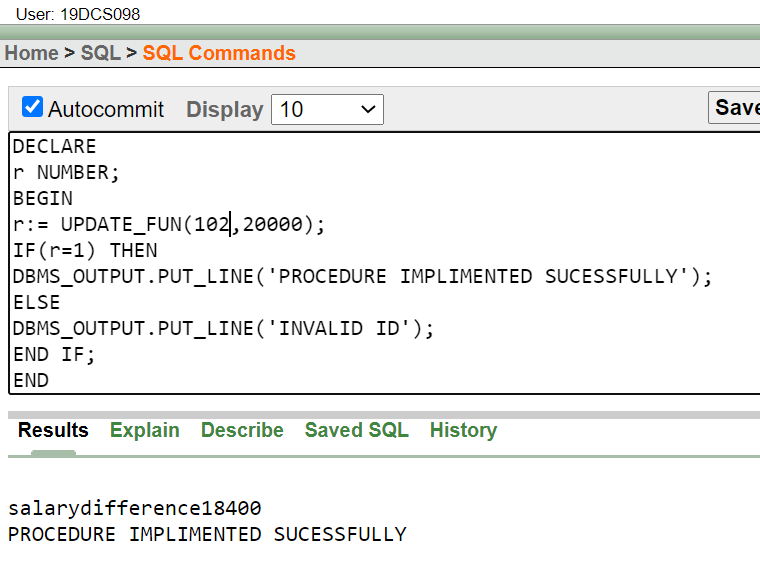
**PRACTICAL-19**

**To perform the concept of function and procedure**

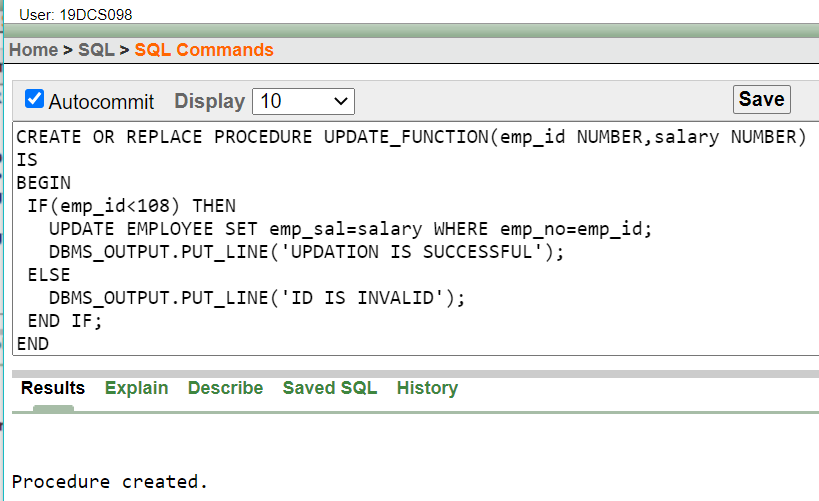
Write a PL/SQL block to update the salary of employee specified by empid. If record exist, then update the salary otherwise display appropriate message. Write a function as well as procedure for updating salary.

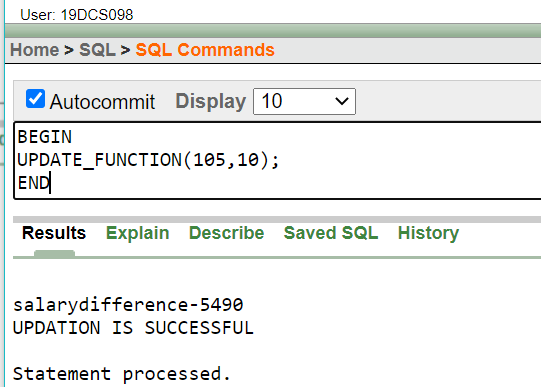
FUNCTION:





PROCEDURE:





**CONCLUSION:**

In the above practical, we learned the concept of functions and procedure.

**PRACTICAL-20**

**To perform the concept of exception handler**

Write a PL/SQL block that will accept the employee code, amount and operation. Based on specified operation amount is added or deducted from salary of said employee. Use user defined exception handler for handling the exception.

SELECT \* FROM employee;

DECLARE

CURSOR c IS SELECT \* FROM employee;

greater exception Exception;

emp\_code employee.emp\_no%type :=101;

V c%rowtype;

amount NUMBER(5):=900;

operation NUMBER(2):=1;

newsl NUMBER(5);

BEGIN

OPEN c;

LOOP

FETCH c INTO v;

EXIT WHEN c%notfound;

IF(v.emp\_no=emp\_code) THEN

CASE OPERATION

WHEN 0 THEN

IF amount>v.emp\_sal THEN

RAISE greater exception;

ELSE

new\_sal=v.emp\_sal-amount;

DBMS\_OUTPUT.PUT\_LINE('AMOUNT : '||amount);

DBMS\_OUTPUT.PUT\_LINE('NEW SALARY : '||new\_sal);

END IF;

WHEN 1 THEN

new\_sal=v.emp\_sal+amount;

DBMS\_OUTPUT.PUT\_LINE('AMOUNT : '||amount);

DBMS\_OUTPUT.PUT\_LINE('NEW SALARY : '||new\_sal);

ELSE

DBMS\_OUTPUT.PUT\_LINE('INVALID EXPRESSION');

END CASE;

END IF;

END LOOP;

CLOSE c;

exception

WHEN greater exception THEN

DBMS\_OUTPUT.PUT\_LINE('AMOUNT : '||amount||' BALANCE : '||v.emp\_sal);

DBMS\_OUTPUT.PUT\_LINE('WITHDRAW!!');

WHEN OTHERS THEN DBMS\_OUTPUT.PUT\_LINE('ERROR');

END;

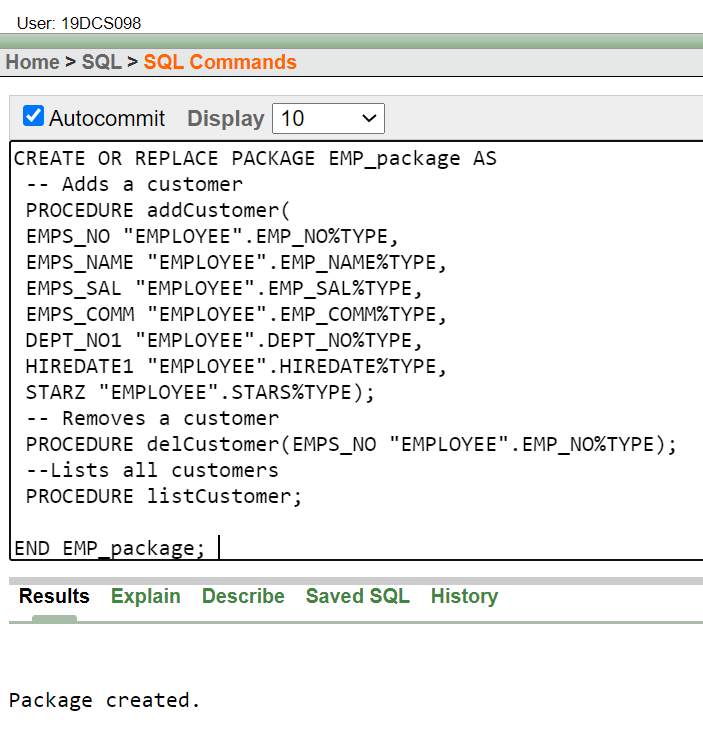
**CONCLUSION:**

In the above practical, we learned the concept of exceptional handling.

**PRACTICAL-21**

**To perform the concept of package**

**Create and invoke a package that contains private and public constructs.**



**CONCLUSION:**

In the above practical, we learnt the concept of packages.