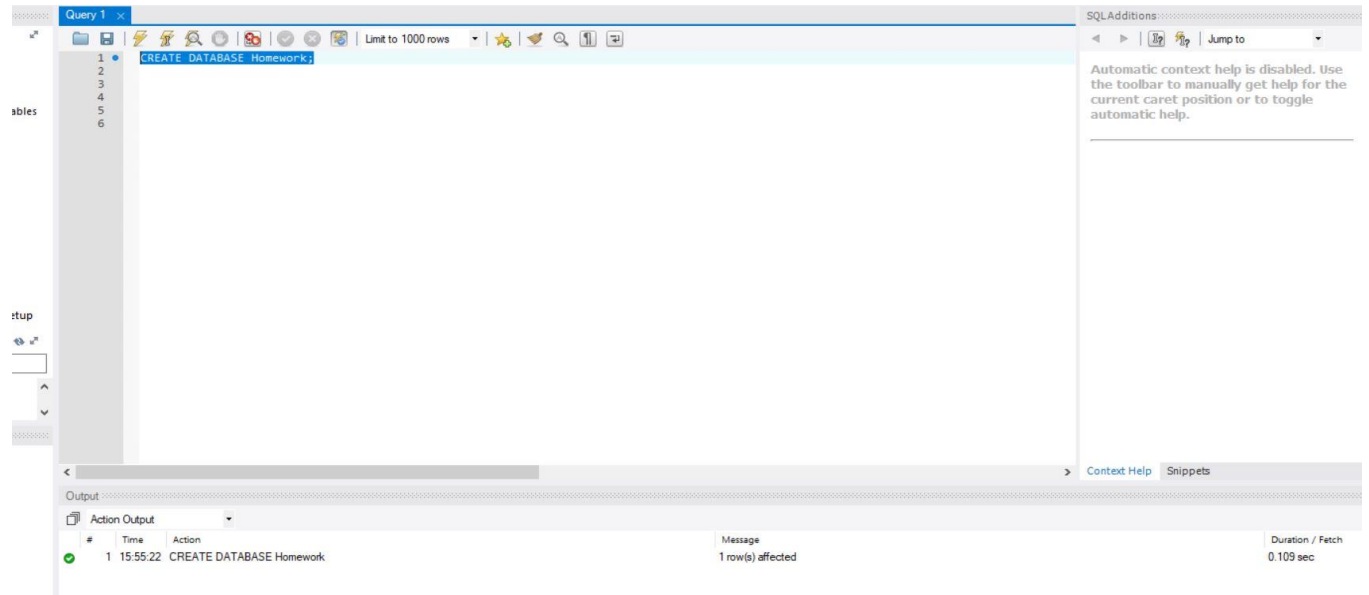


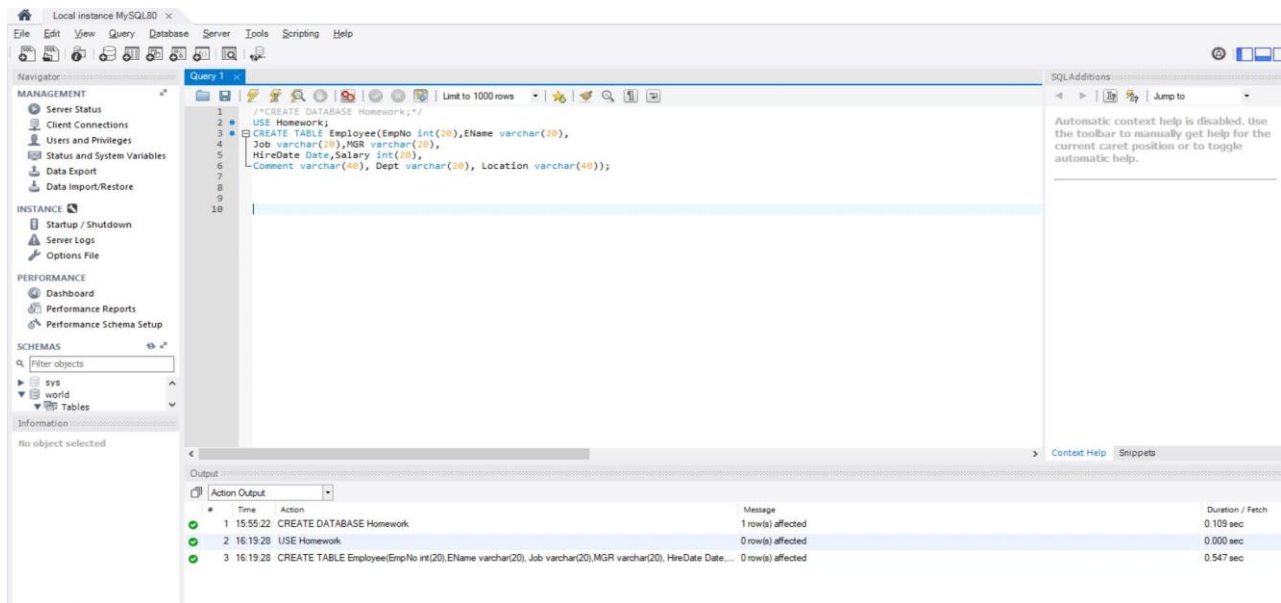
Assignment 1

1) a. CREATE DATABASE Homework;



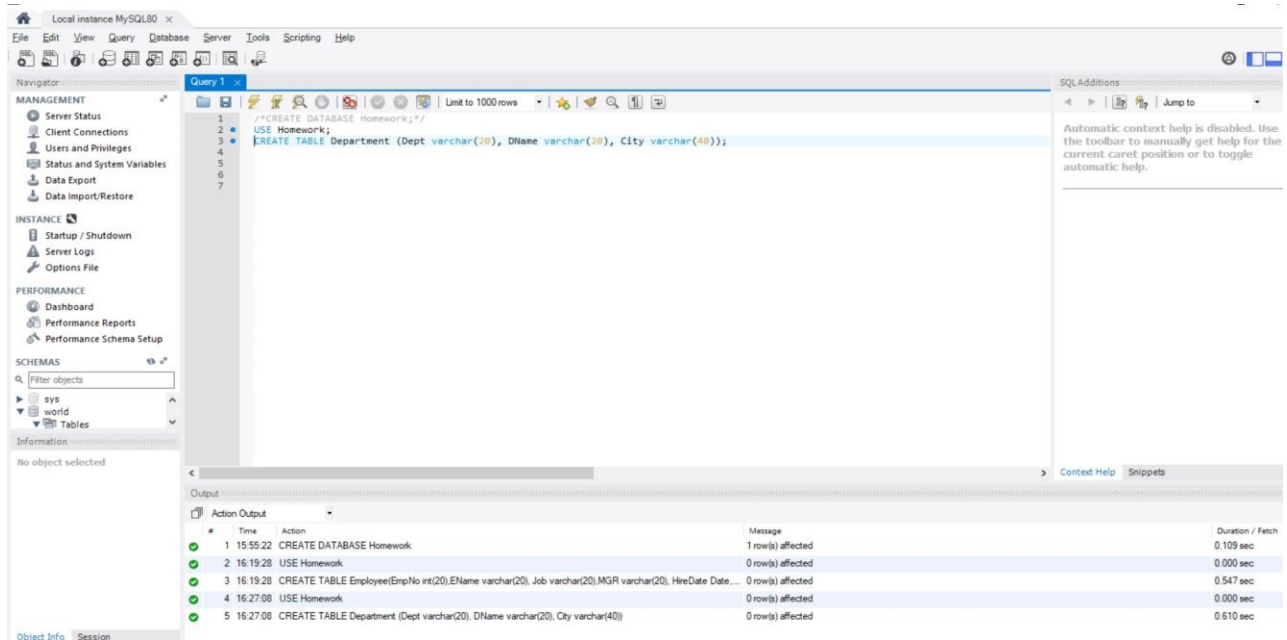
b. USE Homework;

```
CREATE TABLE Employee(EmpNo int(20), EName varchar(20),  
Job varchar(20),MGR varchar(20),  
HireDate Date, Salary int(20),  
Comment varchar(40), Dept varchar(20), Location varchar(40));
```



USE Homework;

CREATE TABLE Department (Dept varchar(20), DName varchar(20), City varchar(40));

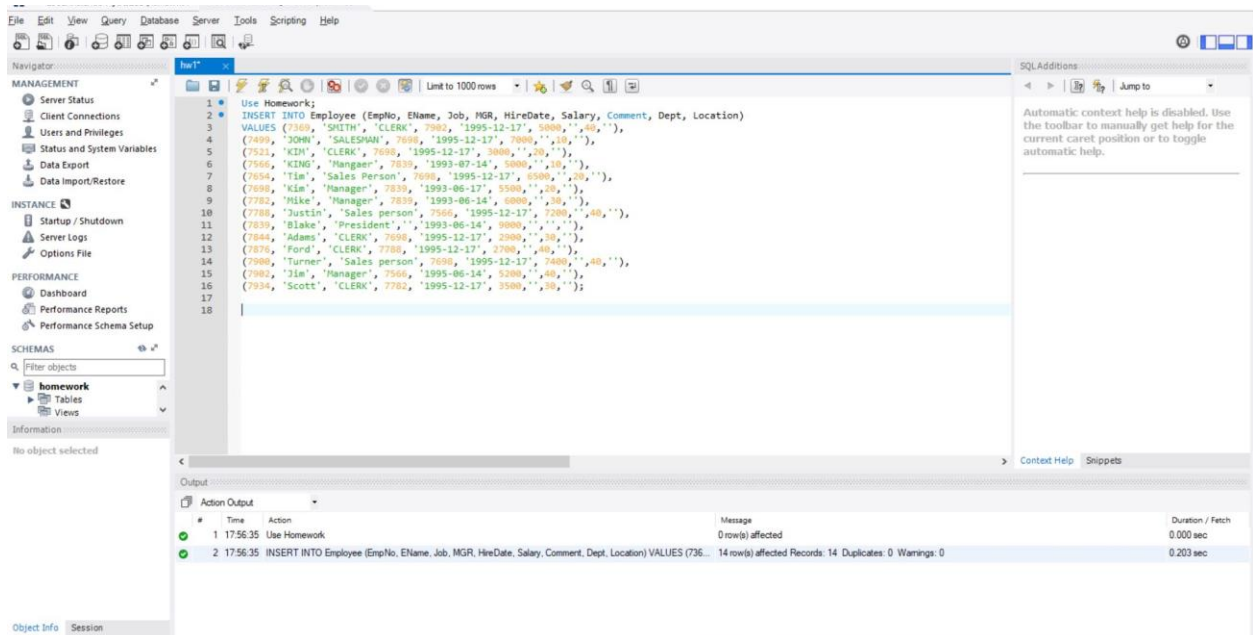


c. Use Homework;

```

INSERT INTO Employee (EmpNo, EName, Job, MGR, HireDate, Salary, Comment, Dept, Location)
VALUES (7369, 'SMITH', 'CLERK', 7902, '1995-12-17', 5000,"40,"),
(7499, 'JOHN', 'SALESMAN', 7698, '1995-12-17', 7000,"10,"),
(7521, 'KIM', 'CLERK', 7698, '1995-12-17', 3000,"20,"),
(7566, 'KING', 'Manager', 7839, '1993-07-14', 5000,"10,"),
(7654, 'Tim', 'Sales Person', 7698, '1995-12-17', 6500,"20,"),
(7698, 'Kim', 'Manager', 7839, '1993-06-17', 5500,"20,"),
(7782, 'Mike', 'Manager', 7839, '1993-06-14', 6000,"30,"),
(7788, 'Justin', 'Sales person', 7566, '1995-12-17', 7200,"40,"),
(7839, 'Blake', 'President',"1993-06-14', 9000,"",""),
(7844, 'Adams', 'CLERK', 7698, '1995-12-17', 2900,"30,"),
(7876, 'Ford', 'CLERK', 7788, '1995-12-17', 2700,"40,"),
(7900, 'Turner', 'Sales person', 7698, '1995-12-17', 7400,"40,"),
(7902, 'Jim', 'Manager', 7566, '1995-06-14', 5200,"40,"),
(7934, 'Scott', 'CLERK', 7782, '1995-12-17', 3500,"30,");

```



Use Homework;

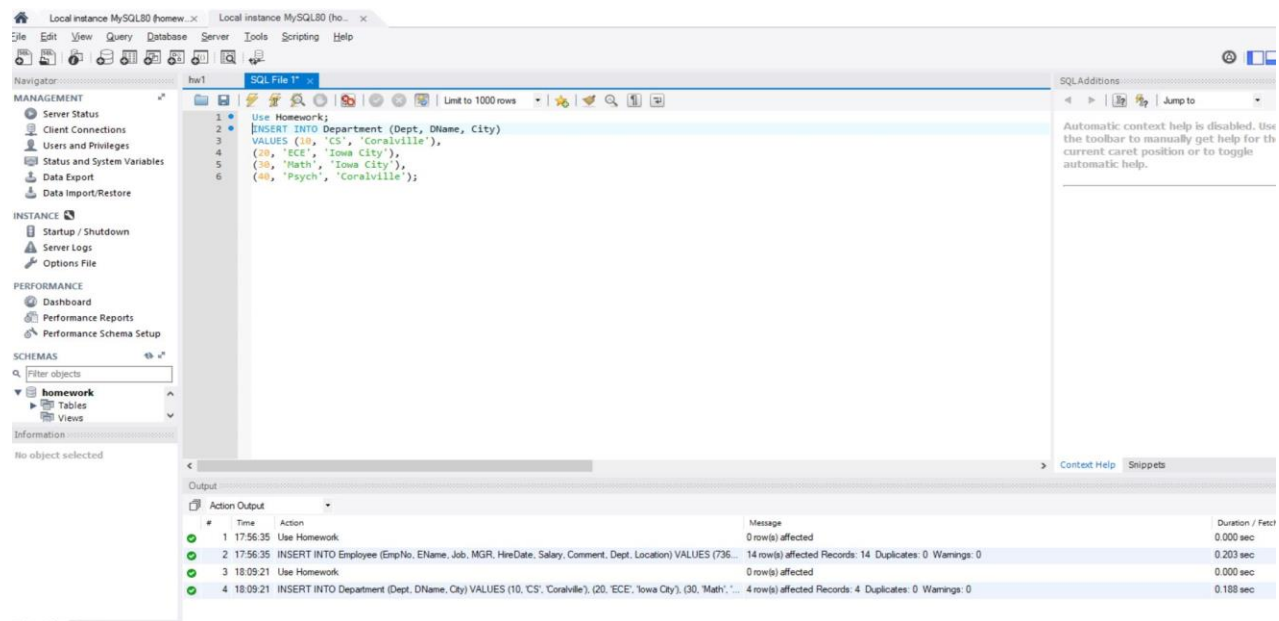
INSERT INTO Department (Dept, DName, City)

VALUES (10, 'CS', 'Coralville'),

(20, 'ECE', 'Iowa City'),

(30, 'Math', 'Iowa City'),

(40, 'Psych', 'Coralville');



2. Three types of database systems -

- Relational
- Hierarchical
- Network

3. Problems with File based access

- Data redundancy** - Data is stored more than once in different files, that means duplicate data may occur in all these files.
- Data inconsistency** - In file processing system, various copies of same data may contain different values. Data is not consistent in this system, it means if a data item needs to be changed then all the files containing that data need to be modified.

- c. Security is problematic - There is very less security in File based access as anyone can easily modify and change the data stored in the files. All the users must have some restriction of accessing data up to a level.
- d. Concurrency issue - Concurrency is the ability of the database to allow multiple users access to the same record without adversely affecting transaction processing. In a file-based system, when an application opens a file, that file is locked. This means that no one else has access to the file at the same time.
- e. Incompatible file format - In file-based processing systems, the structure of data file is coupled with the application program and the structure of data file is dependent on the programming languages in which the application program was developed.

4. Benefits of Relational DBMS

- a. Reduced Data redundancy – Data is typically stored once in Database. The database approach attempts to eliminate the redundancy by integrating the files so that multiple copies of the same data are not stored.
- b. Increased data consistency - If an update is made by one or more users, it is made entirely and in an orderly manner and such updates are available to all users. If a data item is stored more than once and the system is aware of this, the system can ensure that all copies of the item are kept consistent.
- c. Improved data integrity - Database integrity refers to the validity and consistency of stored data. Integrity is usually expressed in terms of constraints, which are consistency rules that the database is not permitted to violate.
- d. Improved security - Database security is the protection of the database from unauthorized users. Integration allows the DBA to define database security, and the DBMS to enforce it.
- a. Enforcement of standards - Integration allows the DBA to define and the DBMS to enforce the necessary standards.

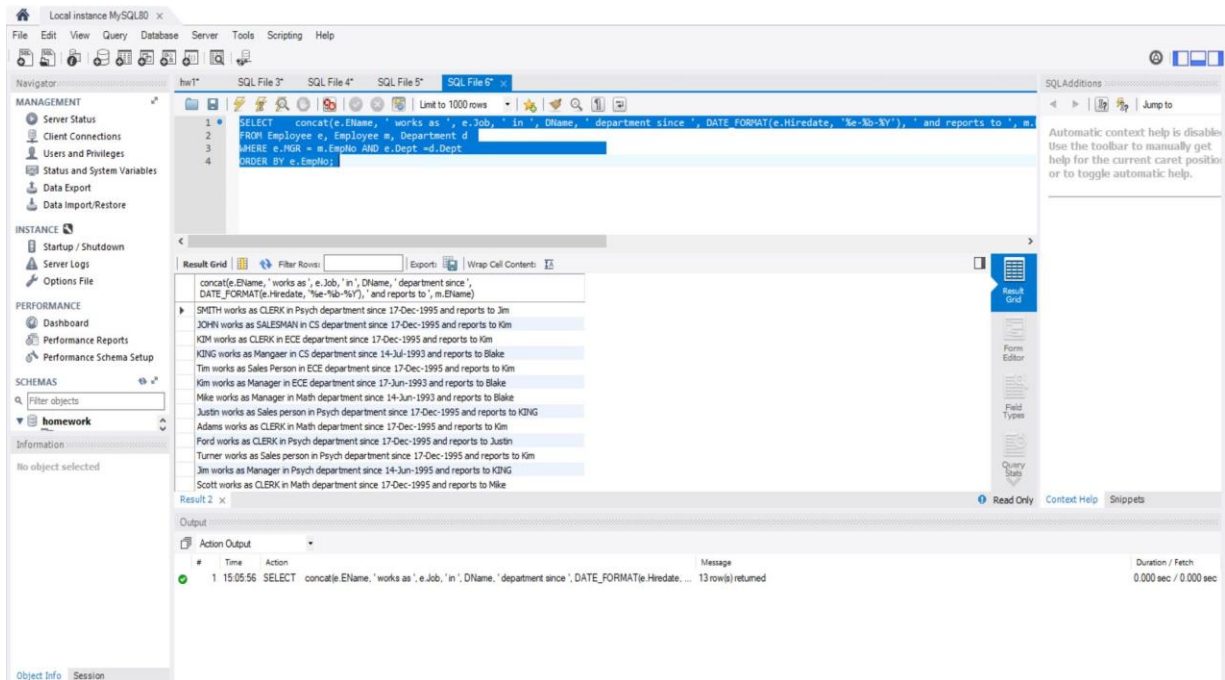
5. Problems with RDBMS

- a. Complexity - RDBMS is an extremely complex piece of software. Failure to understand the system can lead to bad design decisions, which can have serious consequences.
- b. Size - The complexity and breadth of functionality makes the RDBMS an extremely large piece of software, occupying many megabytes of disk space and requiring substantial amounts of memory to run efficiently.
- a. Additional hardware costs - The disk storage requirements for the DBMS and the database may necessitate the purchase of additional storage space.

6.

- a. `SELECT concat(e.ENAME, ' works as ', e.Job, ' in ', DName, ' department since ', DATE_FORMAT(e.Hiredate, '%e-%b-%Y'), ' and reports to ', m.ENAME) FROM Employee e, Employee m, Department d WHERE e.MGR = m.EmpNo AND e.Dept = d.Dept`

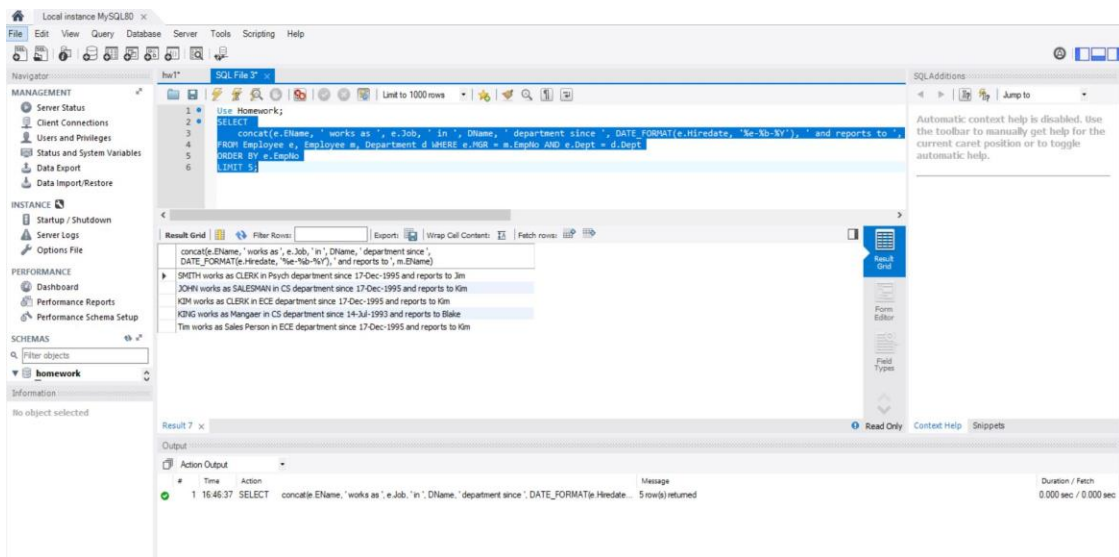
`ORDER BY e.EmpNo;`



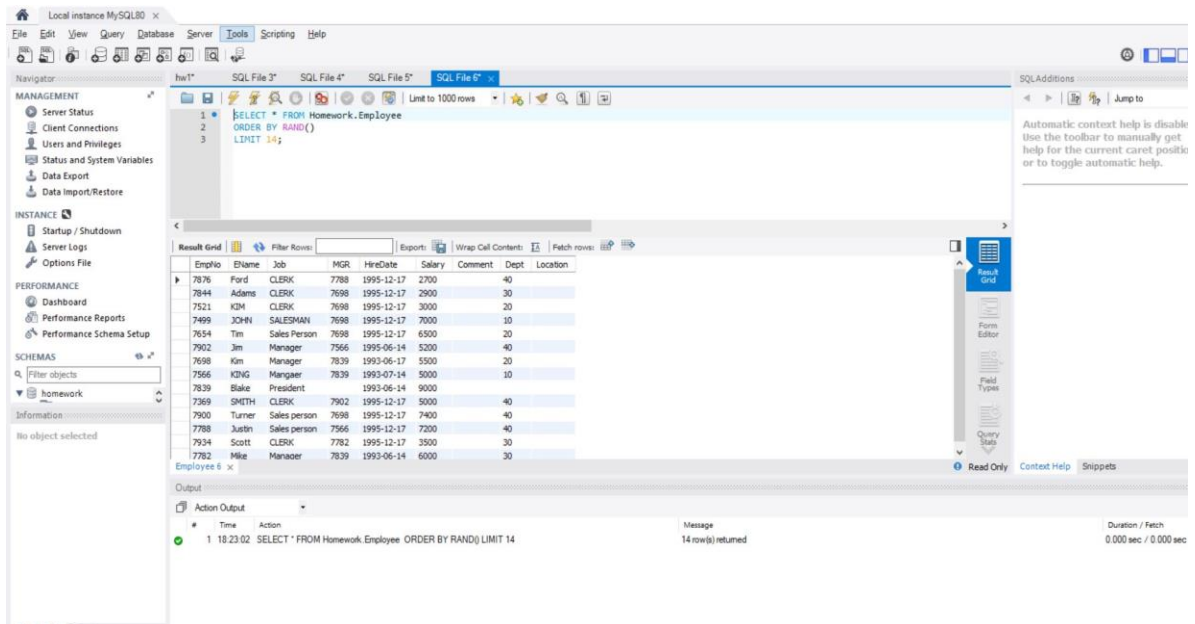
- b. `SELECT concat(e.ENAME, ' works as ', e.Job, ' in ', DName, ' department since ', DATE_FORMAT(e.Hiredate, '%e-%b-%Y'), ' and reports to ', m.ENAME) FROM Employee e, Employee m, Department d WHERE e.MGR = m.EmpNo AND e.Dept = d.Dept`

`ORDER BY e.EmpNo`

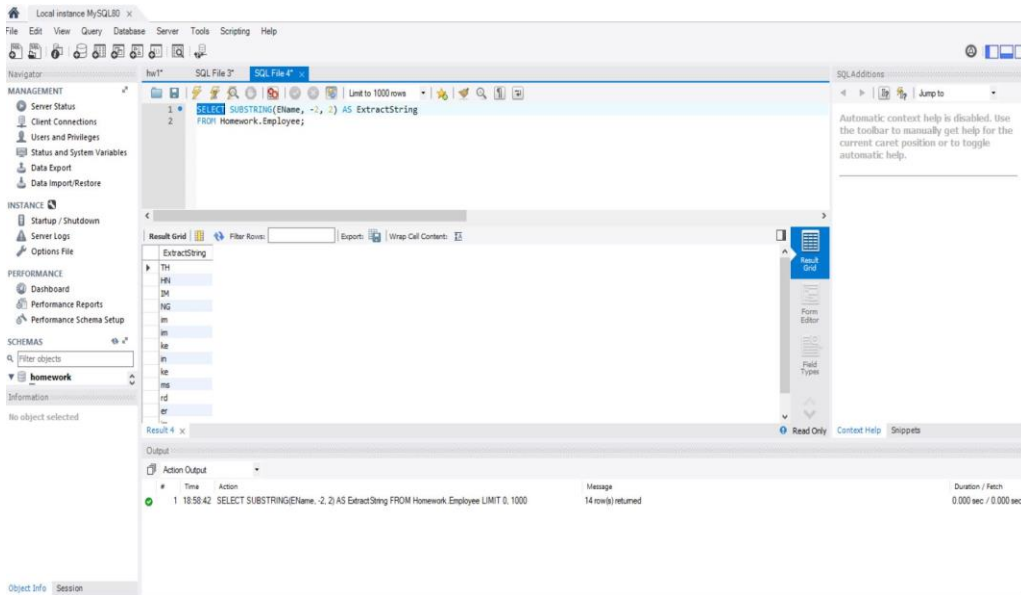
`LIMIT 5;`



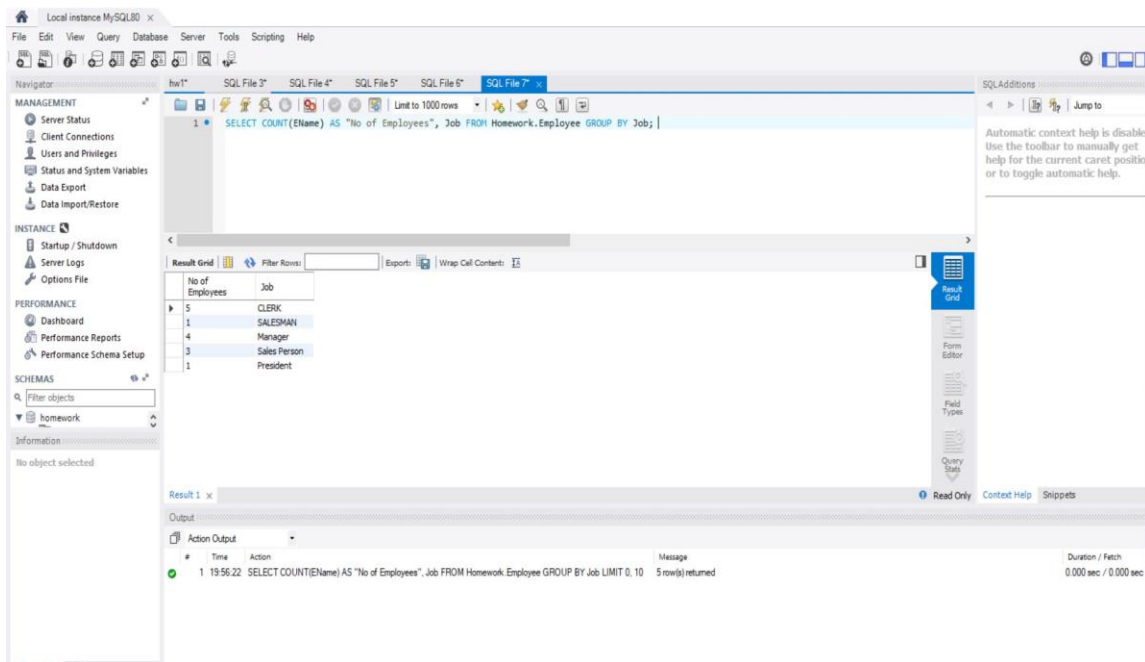
c. `SELECT * FROM Homework.Employee ORDER BY RAND()
LIMIT 14;`



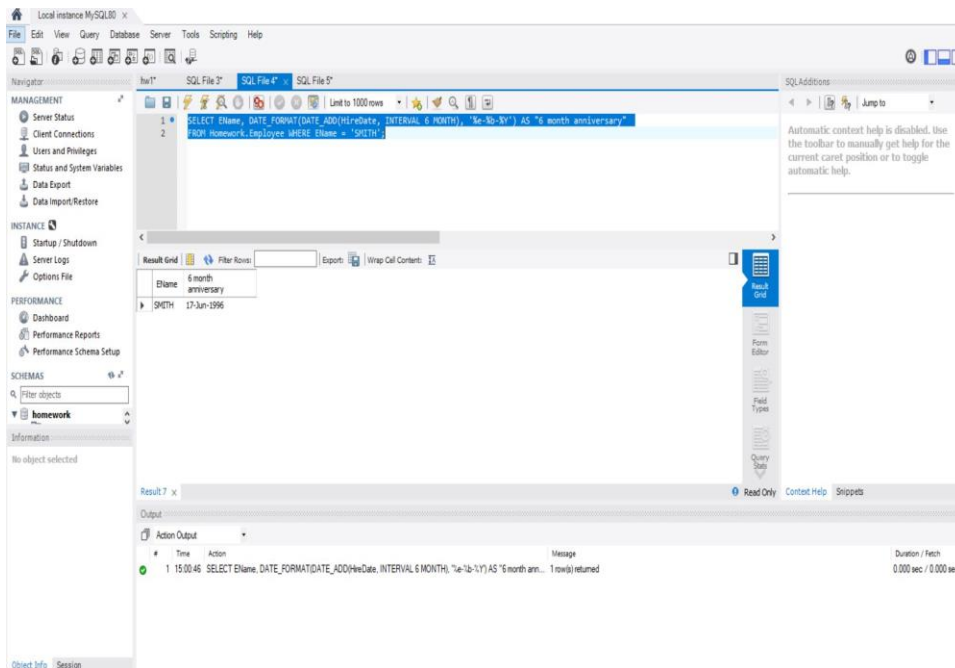
d. `SELECT SUBSTRING(EName, -2, 2) AS ExtractString FROM Homework.Employee;`



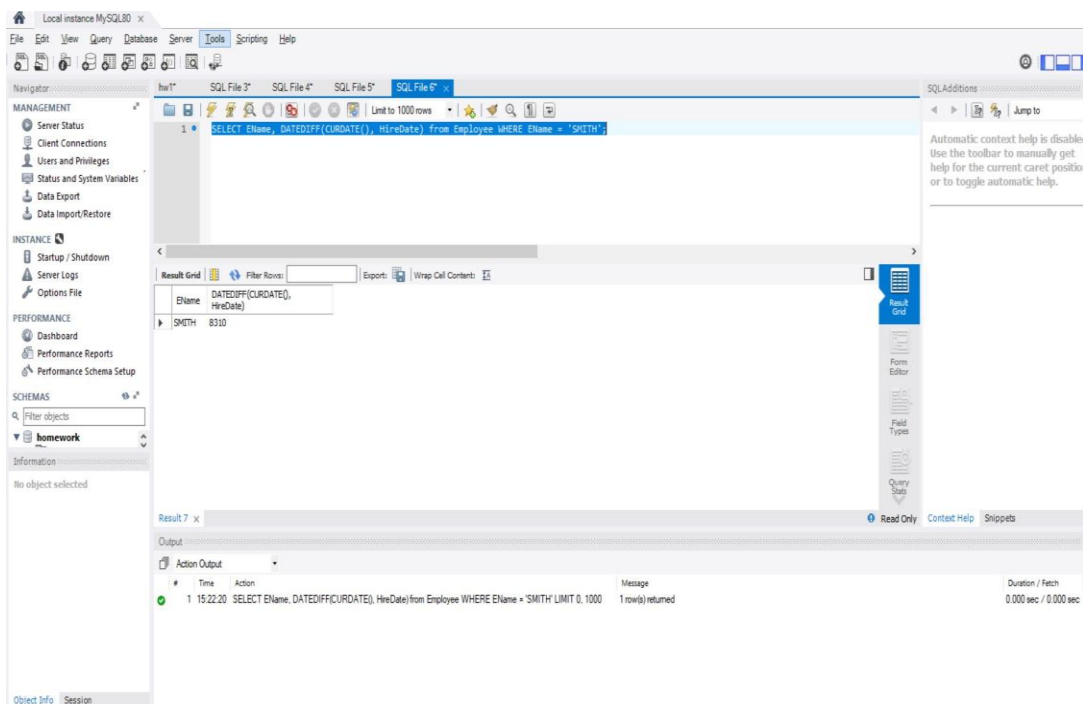
e. SELECT COUNT(EName) AS "No of Employees", Job FROM Homework.Employee GROUP BY Job;



f. SELECT EName, DATE_FORMAT(DATE_ADD(HireDate, INTERVAL 6 MONTH), '%e-%b-%Y') AS "6 month anniversary" FROM Homework.Employee WHERE EName = 'SMITH';



g. SELECT EName, DATEDIFF(CURDATE(), HireDate) from Employee WHERE EName = 'SMITH';



h. SELECT Ename, Salary FROM Homework.Employee
WHERE Salary=(SELECT MAX(Salary) FROM Employee)
OR Salary=(SELECT MIN(Salary) FROM Employee);

The screenshot shows the MySQL Workbench interface for a local instance of MySQL 8.0. The SQL editor contains the following query:

```
1 SELECT Ename, Salary
2 FROM Homework.Employee
3 WHERE Salary=(SELECT MAX(Salary) FROM Employee)
4 OR Salary=(SELECT MIN(Salary) FROM Employee);
```

The query results are displayed in the Result Grid below the editor:

Ename	Salary
Blake	9000
Ford	2700

The Output pane at the bottom shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	14:47:19	SELECT Ename, Salary FROM Homework.Employee WHERE Salary=(SELECT MAX(Salary) FROM Employee) ...	2 row(s) returned	0.000 sec / 0.000 sec