CSLR51 – Database Management Systems Laboratory\
#Session: 11 || Date: 24/10/2024 Viva Due: Q. No. 1 (24/10/2024) Moodle Due: 30/10/2024 at 11 PM

1. Extensible Markup Language (XML)
a. Create an XML file which acts as a database with the following nodes and execute the given queries.
<EmployeeDetails>as the root element
Create <Employee> element with the following Child Nodes for at least 5 employee details. EmpNo, EName, Job ,working Hours ,Dept ,DeptNo ,Salary

employee_details.xml

```
<EmployeeDetails>
  <Employee>
    <EmpNo>101</EmpNo>
    <EName>John</EName>
    <Job>Software Engineer</Job>
    <Dept>IT</Dept>
    <DeptNo>1</DeptNo>
    <Hours>8</Hours>
    <Salary>60000</Salary>
  </Employee>
  <Employee>
    <EmpNo>102</EmpNo>
    <EName>Jane</EName>
    <Job>Project Manager</Job>
    <Dept>IT</Dept>
    <DeptNo>1</DeptNo>
    <Hours>9</Hours>
    <Salary>80000</Salary>
  </Employee>
  <Employee>
    <EmpNo>103</EmpNo>
    <EName>Sam</EName>
    <Job>Team Lead</Job>
    <Dept>HR</Dept>
    <DeptNo>2</DeptNo>
    <Hours>10</Hours>
    <Salary>90000</Salary>
  </Employee>
  <Employee>
    <EmpNo>104</EmpNo>
```

```
        <EName>Mary</EName>
        <Job>HR Manager</Job>
        <Dept>Research</Dept>
        <DeptNo>3</DeptNo>
        <Hours>3</Hours>
        <Salary>57000</Salary>
    </Employee>
    <Employee>
        <EmpNo>105</EmpNo>
        <EName>Tina</EName>
        <Job>Clerk</Job>
        <Dept>HR</Dept>
        <DeptNo>2</DeptNo>
        <Hours>1</Hours>
        <Salary>15000</Salary>
    </Employee>
</EmployeeDetails>
```

```
1.reethi@DESKTOP-8744EFO:~/dir1/dbms$ sudo apt-get install xmlstarlet
```

i. Create a xquery to list the salary > 30000.

```
reethi@DESKTOP-8744EFO:~/dir1/dbms$ xmlstarlet sel -t -m "/EmployeeDetails/E
mployee[Salary>30000]" -v "Salary" -n employee_details.xml
60000
80000
90000
57000
```

ii. Get employee numbers of employees whose last name starts with "S".

```
reethi@DESKTOP-8744EFO:~/dir1/dbms$ xmlstarlet sel -t -m "EmployeeDetails/Em
ployee[starts-with(EName,'S')]" -v "EmpNo" -n employee_details.xml
103
```

iii. Get the names of employees in the "Research" department.

```
reethi@DESKTOP-8744EFO:~/dir1/dbms$ xmlstarlet sel -t -m
"EmployeeDetails/Employee[Dept='Research']" -v "EName" -n employee_details.xml
Mary
```

iv. Get all those employees who work for more than 8 hours.

```
reethi@DESKTOP-8744EFO:~/dir1/dbms$ xmlstarlet sel -t -m
"/EmployeeDetails/Employee[Hours>8]" -v "EName" -n employee_details.xml
Jane
Sam
```

v. Display the salary from highest to lowest.

```
reethi@DESKTOP-8744EFO:~/dir1/dbms$ xmlstarlet sel -t -m "EmployeeDetails/Employee" -v
"Salary" -n "employee_details.xml" | sort -nr
90000
80000
60000
57000
15000
```

vi. Display the employee’s name in the alphabetical order.

```
reethi@DESKTOP-8744EFO:~/dir1/dbms$ xmlstarlet sel -t -m "EmployeeDetails/Em
ployee" -v "EName" -n "employee_details.xml" | sort
Jane
John
Mary
Sam
Tina
```

b. Create an XML file which acts as a database with the following nodes and execute the given queries.
<FlightDetails> as the root element
Create <Flight> element with the following Child Nodes for at least 5 employee details.
FlNo, FlName, PilotName, From, To, Date, Departs Time, Arrives Time, Price

flight.xml

```
<FlightDetails>
   <Flight>
      <FlNo>FL101</FlNo>
      <FlName>Flight A</FlName>
      <PilotName>John Smith</PilotName>
      <From>New York</From>
      <To>Los Angeles</To>
      <Date>2024-10-01</Date>
      <DepartsTime>08:00</DepartsTime>
      <ArrivesTime>11:00</ArrivesTime>
      <Price>8300</Price>
   </Flight>
   <Flight>
      <FlNo>FL102</FlNo>
      <FlName>Flight B</FlName>
      <PilotName>Jane Doe</PilotName>
      <From>Chicago</From>
```

```
        <To>Miami</To>
        <Date>2024-10-02</Date>
        <DepartsTime>09:30</DepartsTime>
        <ArrivesTime>12:30</ArrivesTime>
        <Price>1250</Price>
    </Flight>
    <Flight>
        <FlNo>FL103</FlNo>
        <FlName>Flight C</FlName>
        <PilotName>Mark Johnson</PilotName>
        <From>San Francisco</From>
        <To>Seattle</To>
        <Date>2024-10-03</Date>
        <DepartsTime>07:45</DepartsTime>
        <ArrivesTime>09:45</ArrivesTime>
        <Price>2200</Price>
    </Flight>
    <Flight>
        <FlNo>FL104</FlNo>
        <FlName>Flight D</FlName>
        <PilotName>Emily Davis</PilotName>
        <From>Dallas</From>
        <To>Denver</To>
        <Date>2024-10-04</Date>
        <DepartsTime>10:15</DepartsTime>
        <ArrivesTime>11:30</ArrivesTime>
        <Price>5200</Price>
    </Flight>
    <Flight>
        <FlNo>FL105</FlNo>
        <FlName>Flight E</FlName>
        <PilotName>Michael Brown</PilotName>
        <From>Boston</From>
        <To>San Francisco</To>
        <Date>2024-10-05</Date>
        <DepartsTime>15:00</DepartsTime>
        <ArrivesTime>18:00</ArrivesTime>
        <Price>3500</Price>
    </Flight>
</FlightDetails>
```

i. Create a xquery to list the price of journey < 5000.

```
reethi@DESKTOP-8744EFO:~/dir1/dbms$ xmlstarlet sel -t -m "FlightDetails/Flight[Price<5000]"
-v "Price" -n "flight.xml"
1250
2200
3500
```

ii. Create a xquery to find the departing time of a particular flight on a particular date from a particular city.

```
reethi@DESKTOP-8744EFO:~/dir1/dbms$ xmlstarlet sel -t -m "FlightDetails/Flight[FlNo='FL101'
and Date='2024-10-01' and From='New York']" -v "DepartsTime" -n "flight.xml"
08:00
```

iii. Create a xquery to find the flight names handled by a particular pilot.

```
reethi@DESKTOP-8744EFO:~/dir1/dbms$ xmlstarlet sel -t -m
"FlightDetails/Flight[PilotName='Jane Doe']" -v "FlName" -n "flight.xml"
Flight B
```

iv. Create a xquery to find out the number of flight journeys of a particular flight on a particular date.

```
reethi@DESKTOP-8744EFO:~/dir1/dbms$  xmlstarlet sel -t -v
"count(FlightDetails/Flight[FlNo=='FL101' and Date='2024-10-01'])" flight.xml > output.txt
1
```

v. Create a xquery to find the arrival time of a particular flight on a particular date from a particular city.

```
reethi@DESKTOP-8744EFO:~/dir1/dbms$ xmlstarlet sel -t -m
"FlightDetails/Flight[Date='2024-10-01' and From='New York']" -v "ArrivesTime" -n "flight.xml"
11:00
```

2. Procedures and Functions (Use the Employee Schema from Session 03)

a. Create a procedure to display the details of an employee from the employee table for a given employee id.

```
mysql> CREATE PROCEDURE employee_details(IN EId INT)
    -> BEGIN
    -> SELECT * FROM employee WHERE Ssn=EId;
    -> END//
Query OK, 0 rows affected (0.03 sec)

mysql> CALL employee_details('653298100');
+-------+-------+--------+-----------+------------+----------------------+------+----------+-----------+-----+
| Fname | Minit | Lname  | Ssn       | Bdate      | Address              | Sex  | Salary   | Super_ssn |
Dno |
+-------+-------+--------+-----------+------------+----------------------+------+----------+-----------+-----+
| Hiran | L     | Farook | 653298100 | 1962-12-30 | 21 Oak Forest,Katy,TX | M    | 90000.00 |
65329869  |   9 |
+-------+-------+--------+-----------+------------+----------------------+------+----------+-----------+-----+
1 row in set (0.01 sec)

Query OK, 0 rows affected (0.01 sec)
```

b. Create a procedure to add details of a new employee into the employee table.

```
mysql> CREATE PROCEDURE add_emp(Name VARCHAR(15),Init CHAR(1),LName
VARCHAR(15),Ssn CHAR(9),BDate DATE,Address VARCHAR(30),Sex CHAR(1),Salary DE
CIMAL(10,2),Super_ssn CHAR(9),Dno INT)
    -> BEGIN
    -> INSERT INTO employee
VALUES(Name,Init,LName,Ssn,BDate,Address,Sex,Salary,Super_ssn,Dno);
    -> END//
Query OK, 0 rows affected (0.01 sec)

mysql> CALL add_emp("Rajesh","S","Jasthi","123456789","2004-03-12","No-4,Church
Street","M",12000,"234567890",2)//
Query OK, 1 row affected (0.05 sec)
mysql> SELECT * FROM EMPLOYEE;
    -> //
+----------+-------+---------+-----------+------------+-----------------------+------+----------+-----------+-----+
| Fname    | Minit | Lname   | Ssn       | Bdate      | Address               | Sex  | Salary   | Super_ssn
| Dno |
+----------+-------+---------+-----------+------------+-----------------------+------+----------+-----------+-----+
| Rajesh   | S     | Jasthi  | 123456789 | 2004-03-12 | No-4,Church Street     | M    | 12000.00 |
234567890 |   2 |
| mysql    | K     | S*GH    | 653298000 | 1962-01-30 | 1 Oak Forest,Katy,TX   | M    | 39000.00 |
653298653 |   4 |
| Hiran    | L     | Farook  | 653298100 | 1962-12-30 | 21 Oak Forest,Katy,TX  | M    | 90000.00 |
65329869  |   9 |
| Alam XYZ | K     | Marini  | 653298653 | 1962-12-30 | 98 Oak Forest,Katy,TX  | F    | 37000.00 |
653298663 |   4 |
| Allen    | C     | Mar     | 653298654 | 1962-12-30 | 99 Oak Forest,Katy,TX  | M    | 37000.00 |
653298655 |   4 |
| Richson  | K     | Mario   | 653298655 | 1984-12-03 | 100 Oak Forest,Katy,TX | M    | 39000.00 |
653298653 |   4 |
| Rich     | G     | Mario   | 653298656 | 1984-12-03 | 101 Oak Forest,Katy,TX | M    | 39000.00 |
653298653 |   4 |
| Richton  | G     | Mario   | 653298657 | 1984-12-03 | 102 Oak Forest,Katy,TX | F    | 39000.00 |
653298653 |   4 |
| Teju     | G     | Chouhan | 653298660 | 1962-12-30 | 111 Oak Forest,Katy,TX | M    | 39000.00 |
653298653 |   3 |
| Tejas    | G     | Khana   | 653298661 | 1962-12-30 | 112 Oak Forest,Katy,TX | M    | 41000.00 |
653298660 |   2 |
| Kiran    | P     | Yadav   | 653298662 | 1962-12-30 | 114 Oak Forest,Katy,TX | M    | 30000.00 |
653298661 |   1 |
| Mukesh   | H     | Ragav   | 653298663 | 1962-12-30 | 115 Oak Forest,Katy,TX | F    | 70000.00 |
653298653 |   5 |
| Andrea   | G     | Khan    | 653298665 | 1962-12-30 | 192 Oak Forest,Katy,TX | F    | 60000.00 |
653298660 |   5 |
| Ramsay   | K     | Marini  | 65329869  | 1962-12-30 | 98 Oak Forest,Katy,TX  | M    | 38000.00 |
653298653 |   4 |
| Rocky    | H     | Stone   | 653298698 | 1962-12-30 | 201 Oak Forest,Katy,TX | M    | 51000.00 |
653298654 |   9 |
+----------+-------+---------+-----------+------------+------------------------+------+----------+-----------+-----+
15 rows in set (0.00 sec)
```

c. Write a procedure raise_sal which increases the salary of an employee. It accepts an employee id and the hike amount. It shall use the employee id to find the current salary from the EMPLOYEE table and updates the salary.

```
mysql> CREATE PROCEDURE hike_salary(EId CHAR(9),hike INT)
    -> BEGIN
    -> UPDATE employee SET Salary=Salary+hike WHERE Ssn=EId;
    -> END//
Query OK, 0 rows affected (0.01 sec)

mysql> CALL hike_salary('123456789',50);
Query OK, 1 row affected (0.01 sec)
mysql> select * from employee;
    -> //
+----------+-------+---------+-----------+------------+------------------------+------+----------+-----------+-----+
| Fname    | Minit | Lname   | Ssn       | Bdate      | Address                | Sex  | Salary   | Super_ssn
| Dno |
+----------+-------+---------+-----------+------------+------------------------+------+----------+-----------+-----+
| Rajesh   | S     | Jasthi  | 123456789 | 2004-03-12 | No-4,Church Street     | M    | 12050.00 |
234567890 |   2 |
| mysql    | K     | S*GH    | 653298000 | 1962-01-30 | 1 Oak Forest,Katy,TX   | M    | 39000.00 |
653298653 |   4 |
| Hiran    | L     | Farook  | 653298100 | 1962-12-30 | 21 Oak Forest,Katy,TX  | M    | 90000.00 |
65329869  |   9 |
| Alam XYZ | K     | Marini  | 653298653 | 1962-12-30 | 98 Oak Forest,Katy,TX  | F    | 37000.00 |
653298663 |   4 |
| Allen    | C     | Mar     | 653298654 | 1962-12-30 | 99 Oak Forest,Katy,TX  | M    | 37000.00 |
653298655 |   4 |
| Richson  | K     | Mario   | 653298655 | 1984-12-03 | 100 Oak Forest,Katy,TX | M    | 39000.00 |
653298653 |   4 |
| Rich     | G     | Mario   | 653298656 | 1984-12-03 | 101 Oak Forest,Katy,TX | M    | 39000.00 |
653298653 |   4 |
| Richton  | G     | Mario   | 653298657 | 1984-12-03 | 102 Oak Forest,Katy,TX | F    | 39000.00 |
653298653 |   4 |
| Teju     | G     | Chouhan | 653298660 | 1962-12-30 | 111 Oak Forest,Katy,TX | M    | 39000.00 |
653298653 |   3 |
| Tejas    | G     | Khana   | 653298661 | 1962-12-30 | 112 Oak Forest,Katy,TX | M    | 41000.00 |
653298660 |   2 |
| Kiran    | P     | Yadav   | 653298662 | 1962-12-30 | 114 Oak Forest,Katy,TX | M    | 30000.00 |
653298661 |   1 |
| Mukesh   | H     | Ragav   | 653298663 | 1962-12-30 | 115 Oak Forest,Katy,TX | F    | 70000.00 |
653298653 |   5 |
| Andrea   | G     | Khan    | 653298665 | 1962-12-30 | 192 Oak Forest,Katy,TX | F    | 60000.00 |
653298660 |   5 |
| Ramsay   | K     | Marini  | 65329869  | 1962-12-30 | 98 Oak Forest,Katy,TX  | M    | 38000.00 |
653298653 |   4 |
| Rocky    | H     | Stone   | 653298698 | 1962-12-30 | 201 Oak Forest,Katy,TX | M    | 51000.00 |
653298654 |   9 |
+----------+-------+---------+-----------+------------+------------------------+------+----------+-----------+-----+
15 rows in set (0.00 sec)
```

d. Create a procedure to delete a record from the employee table for a given employee name.

```
mysql> CREATE  PROCEDURE del(Name VARCHAR(15))
    -> BEGIN
    -> DELETE FROM employee WHERE Fname=Name;
    -> END//
Query OK, 0 rows affected (0.01 sec)

mysql> CALL del("Rajesh")//
Query OK, 1 row affected (0.05 sec)

mysql> select * from employee//
+----------+-------+---------+-----------+------------+------------------------+------+----------+-----------+-----+
| Fname    | Minit | Lname   | Ssn       | Bdate      | Address                | Sex  | Salary   | Super_ssn
| Dno |
+----------+-------+---------+-----------+------------+------------------------+------+----------+-----------+-----+
| mysql    | K     | S*GH    | 653298000 | 1962-01-30 | 1 Oak Forest,Katy,TX   | M    | 39000.00 |
653298653 |   4 |
| Hiran    | L     | Farook  | 653298100 | 1962-12-30 | 21 Oak Forest,Katy,TX  | M    | 90000.00 |
65329869  |   9 |
| Alam XYZ | K     | Marini  | 653298653 | 1962-12-30 | 98 Oak Forest,Katy,TX  | F    | 37000.00 |
653298663 |   4 |
| Allen    | C     | Mar     | 653298654 | 1962-12-30 | 99 Oak Forest,Katy,TX  | M    | 37000.00 |
653298655 |   4 |
| Richson  | K     | Mario   | 653298655 | 1984-12-03 | 100 Oak Forest,Katy,TX | M    | 39000.00 |
653298653 |   4 |
| Rich     | G     | Mario   | 653298656 | 1984-12-03 | 101 Oak Forest,Katy,TX | M    | 39000.00 |
653298653 |   4 |
| Richton  | G     | Mario   | 653298657 | 1984-12-03 | 102 Oak Forest,Katy,TX | F    | 39000.00 |
653298653 |   4 |
| Teju     | G     | Chouhan | 653298660 | 1962-12-30 | 111 Oak Forest,Katy,TX | M    | 39000.00 |
653298653 |   3 |
| Tejas    | G     | Khana   | 653298661 | 1962-12-30 | 112 Oak Forest,Katy,TX | M    | 41000.00 |
653298660 |   2 |
| Kiran    | P     | Yadav   | 653298662 | 1962-12-30 | 114 Oak Forest,Katy,TX | M    | 30000.00 |
653298661 |   1 |
| Mukesh   | H     | Ragav   | 653298663 | 1962-12-30 | 115 Oak Forest,Katy,TX | F    | 70000.00 |
653298653 |   5 |
| Andrea   | G     | Khan    | 653298665 | 1962-12-30 | 192 Oak Forest,Katy,TX | F    | 60000.00 |
653298660 |   5 |
| Ramsay   | K     | Marini  | 65329869  | 1962-12-30 | 98 Oak Forest,Katy,TX  | M    | 38000.00 |
653298653 |   4 |
| Rocky    | H     | Stone   | 653298698 | 1962-12-30 | 201 Oak Forest,Katy,TX | M    | 51000.00 |
653298654 |   9 |
+----------+-------+---------+-----------+------------+------------------------+------+----------+-----------+-----+
14 rows in set (0.00 sec)
```

e. Write a procedure which takes a dept_no and lists the names of all employees belonging to that department.

```
mysql> CREATE PROCEDURE listing(Number INT)
    -> BEGIN
    -> SELECT Fname,Minit,Lname from employee WHERE Dno=Number;
    -> END//
Query OK, 0 rows affected (0.01 sec)

mysql> CALL listing(1)//
+-------+-------+-------+
| Fname | Minit | Lname |
+-------+-------+-------+
| Kiran | P     | Yadav |
+-------+-------+-------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)
```

f. Write a procedure that lists the highest salary drawn by an employee in each of the departments. It should make use of a named procedure dept_highest which finds the highest salary drawn by an employee for the given department.

```
mysql> CREATE PROCEDURE dept_highest()
    -> BEGIN
    -> SELECT Dno,MAX(Salary) FROM employee GROUP BY Dno;
    -> END//
Query OK, 0 rows affected (0.01 sec)

mysql> CALL dept_highest()//
+-----+-------------+
| Dno | MAX(Salary) |
+-----+-------------+
|   4 |    39000.00 |
|   9 |    90000.00 |
|   3 |    39000.00 |
|   2 |    41000.00 |
|   1 |    30000.00 |
|   5 |    70000.00 |
+-----+-------------+
6 rows in set (0.01 sec)

Query OK, 0 rows affected (0.01 sec)
```

g. Write a function to display the minimum salary of employees from the employee table.

```
mysql> CREATE FUNCTION min_sal()
    -> RETURNS INT READS SQL DATA
    -> BEGIN
    -> DECLARE m INT;
    -> SELECT MIN(Salary) INTO m FROM employee;
    -> RETURN m;
    -> END//
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT min_sal()//
+-----------+
| min_sal() |
+-----------+
|     30000 |
+-----------+
1 row in set (0.01 sec)
```

h. Write a function to display the number of employees working in the organization.

```
mysql> CREATE FUNCTION num_emp()
```

```
    -> RETURNS INT READS SQL DATA
    -> BEGIN
    -> DECLARE m INT;
    -> SELECT COUNT(*) INTO m FROM employee;
    -> RETURN m;
    -> END//
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT num_emp()//
+-----------+
| num_emp() |
+-----------+
|        14 |
+-----------+
1 row in set (0.00 sec)
```

i. Write a function to display salary of an employee with the given employee id.

```
mysql> CREATE FUNCTION display_sal(EId CHAR(9))
    -> RETURNS INT READS SQL DATA
    -> BEGIN
    -> DECLARE m INT
    -> ;
    -> SELECT Salary INTO m FROM employee WHERE Ssn=EId;
    -> RETURN m;
    -> END//
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT display_sal('653298656')//
+--------------------------+
| display_sal('653298656') |
+--------------------------+
|                    39000 |
+--------------------------+
1 row in set (0.01 sec)
```

j. Write a function which takes dept_no and returns the average salary received by the employees in that department.

```
mysql> CREATE FUNCTION display_sal_dept(Number INT)
    -> RETURNS INT READS SQL DATA
    -> BEGIN
    -> DECLARE m INT;
    -> SELECT AVG(Salary) INTO m FROM employee WHERE Dno=Number;
    -> RETURN m;
    -> END//
```

```
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT display_sal_dept(9)//
+---------------------+
| display_sal_dept(9) |
+---------------------+
|               70500 |
+---------------------+
1 row in set (0.00 sec)
```

k. Write a function that will display the number of employees with salary more than 30000.

```
mysql> CREATE FUNCTION display_greater()
    -> RETURNS INT READS SQL DATA
    -> BEGIN
    -> DECLARE m INT;
    -> SELECT COUNT(*) INTO m FROM employee WHERE Salary>30000;
    -> RETURN m;
    -> END//
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT display_greater()//
+-------------------+
| display_greater() |
+-------------------+
|                13 |
+-------------------+
1 row in set (0.00 sec)
```

l. Write a function that will display the count of the employees working in Tiruchirappalli.

```
mysql> CREATE FUNCTION emp_count()
    -> RETURNS INT READS SQL DATA
    -> BEGIN
    -> DECLARE m INT;
    -> DECLARE n INT;
    -> SELECT Dnumber INTO m FROM dept_locations WHERE Dlocation="Tiruchirappalli";
    -> SELECT COUNT(*) INTO n FROM employee WHERE Dno=m;
    -> RETURN n;
    -> END//
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT emp_count()//
+-------------+
| emp_count() |
+-------------+
|           7 |
+-------------+
1 row in set (0.00 sec)
```

3. Commit, Rollback, Save point and Cascade

For the employee schema created, update any one attribute and subsequently show the result of the following transaction operations.

These statements provide control over the use of transactions:

- **START TRANSACTION or BEGIN - start a new transaction.**
- **COMMIT - commits the current transaction, making its changes permanent.**
- **ROLLBACK - rolls back the current transaction, cancelling its changes.**
- **SET - autocommit disables or enables the default autocommit mode for the current session. By default, MySQL runs with autocommit mode enabled.**

To force MySQL not to commit changes automatically, you can use the following statement:

SET autocommit = 0;

To disable autocommit mode implicitly for a single series of statements, use the START TRANSACTION statement.

```
mysql> SET autocommit=0;
Query OK, 0 rows affected (0.01 sec)

mysql> BEGIN;
Query OK, 0 rows affected (0.00 sec)

mysql> UPDATE employee SET Salary=Salary*1.1 WHERE Ssn='653298665';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM employee WHERE Ssn="653298665";
+--------+-------+-------+-----------+------------+-----------------------+------+----------+-----------+-----+
| Fname  | Minit | Lname | Ssn       | Bdate      | Address               | Sex  | Salary   | Super_ssn |
Dno |
+--------+-------+-------+-----------+------------+-----------------------+------+----------+-----------+-----+
| Andrea | G     | Khan  | 653298665 | 1962-12-30 | 192 Oak Forest,Katy,TX | F    | 66000.00 |
653298660 |   5 |
+--------+-------+-------+-----------+------------+-----------------------+------+----------+-----------+-----+
1 row in set (0.00 sec)

mysql> COMMIT;
Query OK, 0 rows affected (0.01 sec)

mysql> START TRANSACTION;
```

```
Query OK, 0 rows affected (0.00 sec)

mysql> UPDATE employee SET Salary=Salary*0.1 WHERE Ssn='653298665';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM employee WHERE Ssn="653298665";
+--------+-------+-------+-----------+------------+------------------------+------+---------+-----------+-----+
| Fname  | Minit | Lname | Ssn       | Bdate      | Address                | Sex  | Salary  | Super_ssn |
Dno |
+--------+-------+-------+-----------+------------+------------------------+------+---------+-----------+-----+
| Andrea | G     | Khan  | 653298665 | 1962-12-30 | 192 Oak Forest,Katy,TX | F    | 6600.00 |
653298660 |   5 |
+--------+-------+-------+-----------+------------+------------------------+------+---------+-----------+-----+
1 row in set (0.00 sec)

mysql> ROLLBACK;
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT * FROM employee WHERE Ssn="653298665";
+--------+-------+-------+-----------+------------+------------------------+------+----------+-----------+-----+
| Fname  | Minit | Lname | Ssn       | Bdate      | Address                | Sex  | Salary   | Super_ssn |
Dno |
+--------+-------+-------+-----------+------------+------------------------+------+----------+-----------+-----+
| Andrea | G     | Khan  | 653298665 | 1962-12-30 | 192 Oak Forest,Katy,TX | F    | 66000.00 |
653298660 |   5 |
+--------+-------+-------+-----------+------------+------------------------+------+----------+-----------+-----+
1 row in set (0.00 sec)
```