Name: ARKA DAS

Roll: 002210503046

Name: MD NAIMUR RAHAMAN

Roll: 002210503040

MCA 1st year 2nd Semester

Session: 2022 - 2024

DBMS assignments

JADAVPUR UNIVERSITY

Assignment: Set - 1

Assignment – 1

Question 1:

Create the following tables:

Tablename: EMPLOYEE

Structure :

EMP\_CODE char(16)

EMP\_NAME char(20)

DEPT\_CODE char(16)

DESIG\_CODE char(16)

SEX char(1)

ADDRESS char (25)

CITY char (20)

STATE char (20)

PIN char (6)

BASIC Number

JN\_DT Date

Primary key is EMP\_CODE

Tablename : DESIGNATION

Structure :

DESIG\_CODE char(16)

DESIG\_DESC char(20)

Primary key is DESIG\_CODE

Tablename : DEPARTMENT

Structure :

DEPT\_CODE char(16)

DEPT\_NAME char(20)

Primary key is DEPT\_CODE.

Solution:

CREATE TABLE EMPLOYEE

(

EMP\_CODE varchar(16),

EMP\_NAME varchar(20),

DEPT\_CODE varchar(16),

DESIG\_CODE varchar(16),

SEX varchar(1),

ADDRESS varchar(25),

CITY varchar(20),

STATE varchar(20),

PIN varchar(6),

BASIC int(7),

JN\_DT DATE,

primary key (EMP\_CODE)

);

CREATE TABLE DESIGNATION

(

DESIG\_CODE varchar(16),

DESIG\_DESC varchar(20),

primary key (DESIG\_CODE)

);

CREATE TABLE DEPARTMENT

(

DEPT\_CODE varchar(16),

DEPT\_NAME varchar(20),

primary key (DEPT\_CODE)

);

Question 2:

Display the structure of each table

Solution:

desc EMPLOYEE;

MariaDB [assignment\_1]> desc EMPLOYEE;

+------------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+------------+-------------+------+-----+---------+-------+

| EMP\_CODE | varchar(16) | NO | PRI | NULL | |

| EMP\_NAME | varchar(20) | YES | | NULL | |

| DEPT\_CODE | varchar(16) | YES | | NULL | |

| DESIG\_CODE | varchar(16) | YES | | NULL | |

| SEX | varchar(1) | YES | | NULL | |

| ADDRESS | varchar(25) | YES | | NULL | |

| CITY | varchar(20) | YES | | NULL | |

| STATE | varchar(20) | YES | | NULL | |

| PIN | varchar(6) | YES | | NULL | |

| BASIC | int(7) | YES | | NULL | |

| JN\_DT | date | YES | | NULL | |

+------------+-------------+------+-----+---------+-------+

desc DESIGNATION;

MariaDB [assignment\_1]> desc DESIGNATION;

+------------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+------------+-------------+------+-----+---------+-------+

| DESIG\_CODE | varchar(16) | NO | PRI | NULL | |

| DESIG\_DESC | varchar(20) | YES | | NULL | |

+------------+-------------+------+-----+---------+-------+

desc DEPARTMENT;

MariaDB [assignment\_1]> desc DEPARTMENT;

+-----------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+-----------+-------------+------+-----+---------+-------+

| DEPT\_CODE | varchar(16) | NO | PRI | NULL | |

| DEPT\_NAME | varchar(20) | YES | | NULL | |

+-----------+-------------+------+-----+---------+-------+

Question 3:

Insert few rows in each table.

[While entering data in EMP table use DESIG\_CODE which exists in DESIGNATION table and DEPT\_CODE which is exists in DEPARTMENT table. In DESIGNATION table, assign code for Manager, Executive, officer, clerk and helper. In DEPARTMENT table, assign code for Personnel, Production, Purchase, Finance, Research departments]

Solution:

INSERT INTO DESIGNATION (DESIG\_CODE, DESIG\_DESC)

VALUES ('MGR', 'Manager'),

('EXE', 'Executive'),

('OFF', 'Officer'),

('CLK', 'Clerk'),

('HLP', 'Helper');

INSERT INTO DEPARTMENT (DEPT\_CODE, DEPT\_NAME)

VALUES ('PER', 'Personnel'),

('PRO', 'Production'),

('PUR', 'Purchase'),

('FIN', 'Finance'),

('RES', 'Research');

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, JN\_DT)

VALUES ('001', 'Adam Smith', 'PUR', 'MGR', 'M', '123 Main St', 'Berkeley', 'CA', '90001', 50000, '2002-01-01');

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, JN\_DT)

VALUES ('002', 'Maddie Morris', 'PRO', 'EXE', 'F', '456 Elm St', 'Denver', 'CO', '80001', 30000, '20015-03-10');

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, JN\_DT)

VALUES ('003', 'Edward Johnson', 'PUR', 'OFF', 'M', '789 Oak St', 'Atlanta', 'GA', '30002', 40000, '2010-06-25');

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, JN\_DT)

VALUES ('004', 'Jenifer Martin', 'MGR', 'OFF', 'F', '245 Perl St', 'Nashville', 'TN', '80001', 37010, '2008-01-24');

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, JN\_DT)

VALUES ('005', 'Sam Joe', 'FIN', 'CLK', 'M', '245 New St', 'Denver', 'CO', '80001', 35000, '2002-02-24');

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, JN\_DT)

VALUES ('006', 'Jessica Altman', 'PUR', 'HLP', 'F', '245 Old St', 'Atlanta', 'GA', '80001', 45000, '2003-05-22');

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, JN\_DT)

VALUES ('007', 'Sarra Jhones', 'PRO', 'CLK', 'F', '160 Saint St', 'Nashville', 'TN', '20001', 48000, '2001-06-18');

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, JN\_DT)

VALUES ('008', 'Maxine', 'FIN', 'CLK', 'F', '160 Perl St', 'Nashville', 'TN', '20521', 48000, '2002-06-18');

MariaDB [assignment\_1]> select \* from employee;

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | DESIG\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT |

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

| 001 | Adam Smith | PUR | MGR | M | 123 Main St | Berkeley | CA | 90001 | 50000 | 2002-01-01 |

| 002 | Maddie Morris | PRO | EXE | F | 456 Elm St | Denver | CO | 80001 | 0 | 0000-00-00 |

| 003 | Edward Johnson | PUR | OFF | M | 789 Oak St | Atlanta | GA | 30002 | 40000 | 2010-06-25 |

| 004 | Jenifer Martin | MGR | OFF | F | 245 Perl St | Nashville | TN | 80001 | 37010 | 2008-01-24 |

| 005 | Sam Joe | FIN | CLK | M | 245 New St | Denver | CO | 80001 | 35000 | 2002-02-24 |

| 006 | Jessica Altman | PUR | HLP | F | 245 Old St | Atlanta | GA | 80001 | 45000 | 2003-05-22 |

| 007 | Sarra Jhones | PRO | CLK | F | 160 Saint St | Nashville | TN | 20001 | 48000 | 2001-06-18 |

| 008 | Maxine | FIN | CLK | F | 160 Perl St | Nashville | TN | 20521 | 48000 | 2002-06-18 |

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

Question 4:

In EMP table insert few rows without DEPT\_CODE and BASIC.

Solution:

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, JN\_DT)

VALUES ('009', 'Sophie Matrin', 'CLK', 'F', '246 Maple St', 'Boston', 'MA', '02101', '2014-04-28');

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, JN\_DT)

VALUES ('010', 'Thomas Brown', 'HLP', 'M', '369 Pine St', 'San Francisco', 'CA', '94101', '2016-10-12');

Question 5:

Find the rows with unassigned DEPT\_CODE

Solution:

select \* from employee where dept\_code is NULL;

MariaDB [assignment\_1]> select \* from employee where dept\_code is NULL;

+----------+---------------+-----------+------------+------+--------------+---------------+-------+-------+-------+------------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | DESIG\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT |

+----------+---------------+-----------+------------+------+--------------+---------------+-------+-------+-------+------------+

| 009 | Sophie Matrin | NULL | CLK | F | 246 Maple St | Boston | MA | 02101 | NULL | 2014-04-28 |

| 010 | Thomas Brown | NULL | HLP | M | 369 Pine St | San Francisco | CA | 94101 | NULL | 2016-10-12 |

+----------+---------------+-----------+------------+------+--------------+---------------+-------+-------+-------+------------+

Question 6:

Find the rows with BASIC equal to zero

Solution:

select \* from employee where basic = 0;

MariaDB [assignment\_1]> select \* from employee where basic = 0;

+----------+---------------+-----------+------------+------+------------+--------+-------+-------+-------+------------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | DESIG\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT |

+----------+---------------+-----------+------------+------+------------+--------+-------+-------+-------+------------+

| 002 | Maddie Morris | PRO | EXE | F | 456 Elm St | Denver | CO | 80001 | 0 | 0000-00-00 |

+----------+---------------+-----------+------------+------+------------+--------+-------+-------+-------+------------+

Question 7:

Find the rows with unassigned Basic.

Solution:

select \* from employee where basic is NULL;

MariaDB [assignment\_1]> select \* from employee where basic is NULL;

+----------+---------------+-----------+------------+------+--------------+---------------+-------+-------+-------+------------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | DESIG\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT |

+----------+---------------+-----------+------------+------+--------------+---------------+-------+-------+-------+------------+

| 009 | Sophie Matrin | NULL | CLK | F | 246 Maple St | Boston | MA | 02101 | NULL | 2014-04-28 |

| 010 | Thomas Brown | NULL | HLP | M | 369 Pine St | San Francisco | CA | 94101 | NULL | 2016-10-12 |

+----------+---------------+-----------+------------+------+--------------+---------------+-------+-------+-------+------------+

Question 8:

Find the average basic of the employees.

Solution:

select avg(basic) as 'Avg\_Basic' from employee;

MariaDB [assignment\_1]> select avg(basic) as 'Avg\_Basic' from employee;

+------------+

| Avg\_Basic |

+------------+

| 37876.2500 |

+------------+

Question 9:

Replace the BASIC with 0 for the rows with unassigned Basic.

Solution:

update employee set basic = 0 where basic is NULL;

MariaDB [assignment\_1]> update employee set basic = 0 where basic is NULL;

Query OK, 2 rows affected (0.004 sec)

Rows matched: 2 Changed: 2 Warnings: 0

Question 10:

Again, find the average Basic. (Note the difference of result obtained in Q.8 & Q.10.)

Solution:

select avg(basic) as 'Avg\_Basic' from employee;

MariaDB [assignment\_1]> select avg(basic) as 'Avg\_Basic' from employee;

+------------+

| Avg\_Basic |

+------------+

| 30301.0000 |

+------------+

Question 11:

Delete the rows with unassigned DEPT\_CODE

Solution:

delete from employee where dept\_code is NULL;

MariaDB [assignment\_1]> delete from employee where dept\_code is NULL;

Query OK, 2 rows affected (0.008 sec)

Question 12:

Say, Net pay of an employee = Basic + HRA + DA where HRA is 50% of the Basic & DA is 40% of Basic. Show the employee name & Net pay for all employees.

Solution:

select emp\_name, basic + (basic \* 0.4) + (basic \* 0.5) as "Net\_Pay" from employee;

MariaDB [assignment\_1]> select emp\_name, basic + (basic \* 0.4) + (basic \* 0.5) as "Net\_Pay" from employee;

+----------------+---------+

| emp\_name | Net\_Pay |

+----------------+---------+

| Adam Smith | 95000.0 |

| Maddie Morris | 0.0 |

| Edward Johnson | 76000.0 |

| Jenifer Martin | 70319.0 |

| Sam Joe | 66500.0 |

| Jessica Altman | 85500.0 |

| Sarra Jhones | 91200.0 |

| Maxine | 91200.0 |

+----------------+---------+

Question 13:

Show the EMP\_NAME & BASIC in the ascending order of DEPT\_CODE. The employee name must appear in uppercase.

Solution:

select upper(emp\_name), basic from employee order by dept\_code;

MariaDB [assignment\_1]> select upper(emp\_name), basic from employee order by dept\_code;

+-----------------+-------+

| upper(emp\_name) | basic |

+-----------------+-------+

| SAM JOE | 35000 |

| Maxine | 48000 |

| JENIFER MARTIN | 37010 |

| MADDIE MORRIS | 0 |

| SARRA JHONES | 48000 |

| ADAM SMITH | 50000 |

| EDWARD JOHNSON | 40000 |

| JESSICA ALTMAN | 45000 |

+-----------------+-------+

Question 14:

Find the employees who have joined after 1st January 2010.

Solution:

select \* from employee where jn\_dt > '2010-01-01';

MariaDB [assignment\_1]> select \* from employee where jn\_dt > '2010-01-01';

+----------+----------------+-----------+------------+------+------------+---------+-------+-------+-------+------------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | DESIG\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT |

+----------+----------------+-----------+------------+------+------------+---------+-------+-------+-------+------------+

| 003 | Edward Johnson | PUR | OFF | M | 789 Oak St | Atlanta | GA | 30002 | 40000 | 2010-06-25 |

+----------+----------------+-----------+------------+------+------------+---------+-------+-------+-------+------------+

Question 15:

Find, how many employees have joined in the month of January?

Solution:

select count(emp\_code) as 'Joiners on January' from employee where month(jn\_dt) = '01';

MariaDB [assignment\_1]> select count(emp\_code) as 'Joiners on January' from employee where month(jn\_dt) = '01';

+--------------------+

| Joiners on January |

+--------------------+

| 2 |

+--------------------+

Question 16:

Find the maximum & minimum Basic.

Solution:

select max(basic) as 'Max\_Basic', min(basic) as 'Min\_Basic' from employee;

MariaDB [assignment\_1]> select max(basic) as 'Max\_Basic', min(basic) as 'Min\_Basic' from employee;

+-----------+-----------+

| Max\_Basic | Min\_Basic |

+-----------+-----------+

| 50000 | 0 |

+-----------+-----------+

Question 17:

Find how many Female employees are there?

Solution:

select \* from employee where sex = 'F';

MariaDB [assignment\_1]> select \* from employee where sex = 'F';

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | DESIG\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT |

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

| 002 | Maddie Morris | PRO | EXE | F | 456 Elm St | Denver | CO | 80001 | 0 | 0000-00-00 |

| 004 | Jenifer Martin | MGR | OFF | F | 245 Perl St | Nashville | TN | 80001 | 37010 | 2008-01-24 |

| 006 | Jessica Altman | PUR | HLP | F | 245 Old St | Atlanta | GA | 80001 | 45000 | 2003-05-22 |

| 007 | Sarra Jhones | PRO | CLK | F | 160 Saint St | Nashville | TN | 20001 | 48000 | 2001-06-18 |

| 008 | Maxine | FIN | CLK | F | 160 Perl St | Nashville | TN | 20521 | 48000 | 2002-06-18 |

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

Question 18:

Replace CITY with existing value converted into uppercase for all rows.

Solution:

update employee set city = upper(city);

MariaDB [assignment\_1]> update employee set city = upper(city);

Query OK, 8 rows affected (0.003 sec)

Rows matched: 8 Changed: 8 Warnings: 0

MariaDB [assignment\_1]> select \* from employee;

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | DESIG\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT |

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

| 001 | Adam Smith | PUR | MGR | M | 123 Main St | BERKELEY | CA | 90001 | 50000 | 2002-01-01 |

| 002 | Maddie Morris | PRO | EXE | F | 456 Elm St | DENVER | CO | 80001 | 0 | 0000-00-00 |

| 003 | Edward Johnson | PUR | OFF | M | 789 Oak St | ATLANTA | GA | 30002 | 40000 | 2010-06-25 |

| 004 | Jenifer Martin | MGR | OFF | F | 245 Perl St | NASHVILLE | TN | 80001 | 37010 | 2008-01-24 |

| 005 | Sam Joe | FIN | CLK | M | 245 New St | DENVER | CO | 80001 | 35000 | 2002-02-24 |

| 006 | Jessica Altman | PUR | HLP | F | 245 Old St | ATLANTA | GA | 80001 | 45000 | 2003-05-22 |

| 007 | Sarra Jhones | PRO | CLK | F | 160 Saint St | NASHVILLE | TN | 20001 | 48000 | 2001-06-18 |

| 008 | Maxine | FIN | CLK | F | 160 Perl St | NASHVILLE | TN | 20521 | 48000 | 2002-06-18 |

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

Question 19:

Find in how many different cities various employees are residing?

Solution:

select count(distinct(city)) as 'Unique cities' from employee;

MariaDB [assignment\_1]> select count(distinct(city)) as 'Unique cities' from employee;

+---------------+

| Unique cities |

+---------------+

| 4 |

+---------------+

Question 20:

Display the employee information in the ascending order of DEPT\_CODE and within a Department, it should be in the descending order of BASIC.

Solution:

select \* from employee order by dept\_code, basic desc;

MariaDB [assignment\_1]> select \* from employee order by dept\_code, basic desc;

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | DESIG\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT |

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

| 008 | Maxine | FIN | CLK | F | 160 Perl St | NASHVILLE | TN | 20521 | 48000 | 2002-06-18 |

| 005 | Sam Joe | FIN | CLK | M | 245 New St | DENVER | CO | 80001 | 35000 | 2002-02-24 |

| 004 | Jenifer Martin | MGR | OFF | F | 245 Perl St | NASHVILLE | TN | 80001 | 37010 | 2008-01-24 |

| 007 | Sarra Jhones | PRO | CLK | F | 160 Saint St | NASHVILLE | TN | 20001 | 48000 | 2001-06-18 |

| 002 | Maddie Morris | PRO | EXE | F | 456 Elm St | DENVER | CO | 80001 | 0 | 0000-00-00 |

| 001 | Adam Smith | PUR | MGR | M | 123 Main St | BERKELEY | CA | 90001 | 50000 | 2002-01-01 |

| 006 | Jessica Altman | PUR | HLP | F | 245 Old St | ATLANTA | GA | 80001 | 45000 | 2003-05-22 |

| 003 | Edward Johnson | PUR | OFF | M | 789 Oak St | ATLANTA | GA | 30002 | 40000 | 2010-06-25 |

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

Assignment: Set - 2

Assignment – 2

Question 1:

From the EMP table show the minimum, maximum and average basic for each department (show dept. Code).

Solution:

select dept\_code, max(basic), min(basic), avg(basic) from employee group by dept\_code;

MariaDB [assignment\_1]> select dept\_code, max(basic), min(basic), avg(basic) from employee group by dept\_code;

+-----------+------------+------------+------------+

| dept\_code | max(basic) | min(basic) | avg(basic) |

+-----------+------------+------------+------------+

| FIN | 48000 | 35000 | 41500.0000 |

| MGR | 37010 | 37010 | 37010.0000 |

| PRO | 48000 | 32500 | 40250.0000 |

| PUR | 50000 | 40000 | 45000.0000 |

+-----------+------------+------------+------------+

Question 2:

Find the number of female employees in each department (show dept. Code).

Solution:

select dept\_code, count(emp\_code) as 'Count of Female' from employee where sex = 'F' group by dept\_code;

MariaDB [assignment\_1]> select dept\_code, count(emp\_code) as 'Count of Female' from employee where sex = 'F' group by dept\_code;

+-----------+-----------------+

| dept\_code | Count of Female |

+-----------+-----------------+

| FIN | 1 |

| MGR | 1 |

| PRO | 2 |

| PUR | 1 |

+-----------+-----------------+

Question 3:

Find the city wise no. of employees for each department (show dept. Code).

Solution:

select distinct(dept\_code), city, count(\*) from employee group by city;

MariaDB [assignment\_1]> select distinct(dept\_code), city, count(\*) from employee group by city;

+-----------+-----------+----------+

| dept\_code | city | count(\*) |

+-----------+-----------+----------+

| PUR | ATLANTA | 2 |

| PUR | BERKELEY | 1 |

| PRO | DENVER | 2 |

| MGR | NASHVILLE | 3 |

+-----------+-----------+----------+

Question 4:

Show the designation wise no of employees who have joined in the year 2000 in each department. The listing should appear in the ascending order of no. of employees.

Solution:

select desig\_code, dept\_code, count(emp\_code) from employee where year(jn\_dt) = '2002' group by desig\_code;

MariaDB [assignment\_1]> select desig\_code, dept\_code, count(emp\_code) from employee where year(jn\_dt) = '2000' group by desig\_code;

+------------+-----------+-----------------+

| desig\_code | dept\_code | count(emp\_code) |

+------------+-----------+-----------------+

| CLK | FIN | 2 |

| MGR | PUR | 1 |

+------------+-----------+-----------------+

Question 5:

Find the department code wise total basic of male employees only for the departments for which such total is more than 50,000 and the listing should appear in the descending order of total basic.

Solution:

select dept\_code, sum(basic) from employee where sex = 'M' group by dept\_code having sum(basic) > 50000;

MariaDB [assignment\_1]> select dept\_code, sum(basic) from employee where sex = 'M' group by dept\_code having sum(basic) > 50000;

+-----------+------------+

| dept\_code | sum(basic) |

+-----------+------------+

| PUR | 90000 |

+-----------+------------+

Question 6:

Show the employee name, Designation description and basic for all employees.

Solution:

select e.emp\_name, d.desig\_desc from employee e, designation d

where e.desig\_code = d.desig\_code;

MariaDB [assignment\_1]> select e.emp\_name, d.desig\_desc from employee e,designation d

-> where e.desig\_code = d.desig\_code;

+----------------+------------+

| emp\_name | desig\_desc |

+----------------+------------+

| Adam Smith | Manager |

| Maddie Morris | Executive |

| Edward Johnson | Officer |

| Jenifer Martin | Officer |

| Sam Joe | Clerk |

| Jessica Altman | Helper |

| Sarra Jhones | Clerk |

| Maxine | Clerk |

+----------------+------------+

Question 7:

Show the employee name, Designation description, Department Name & Basic for all employees.

Solution:

select e.emp\_name, d.desig\_code, dp.dept\_name, e.basic

from employee e, designation d, department dp

where e.desig\_code = d.desig\_code and e.dept\_code = dp.dept\_code;

MariaDB [assignment\_1]> select e.emp\_name, d.desig\_code, dp.dept\_name, e.basic

-> from employee e, designation d, department dp

-> where e.desig\_code = d.desig\_code and e.dept\_code = dp.dept\_code;

+----------------+------------+------------+-------+

| emp\_name | desig\_code | dept\_name | basic |

+----------------+------------+------------+-------+

| Sam Joe | CLK | Finance | 35000 |

| Maxine | CLK | Finance | 48000 |

| Maddie Morris | EXE | Production | 32500 |

| Sarra Jhones | CLK | Production | 48000 |

| Adam Smith | MGR | Purchase | 50000 |

| Edward Johnson | OFF | Purchase | 40000 |

| Jessica Altman | HLP | Purchase | 45000 |

+----------------+------------+------------+-------+

Question 8:

Find the department Codes in which no employee works.

Solution:

select dept\_code from department where dept\_code

not in (select dept\_code from employee);

MariaDB [assignment\_1]> select dept\_code from department where dept\_code

-> not in (select dept\_code from employee);

+-----------+

| dept\_code |

+-----------+

| PER |

| RES |

+-----------+

Question 9:

Find the department names where at least one employee works.

Solution:

select dept\_name from department

where dept\_code in

(select dept\_code from employee);

MariaDB [assignment\_1]> select dept\_name from department

-> where dept\_code in

-> (select dept\_code from employee);

+------------+

| dept\_name |

+------------+

| Finance |

| Production |

| Purchase |

+------------+

Question 10:

Find the department names where at least 10 employees work.

Solution:

select d.dept\_name from department d

where (select count(\*) from employee e where e.dept\_code = d.dept\_code) > 9;

MariaDB [assignment\_1]> select d.dept\_name from department d

-> where (select count(\*) from employee e where e.dept\_code = d.dept\_code) > 9;

Empty set

Question 11:

Find the department code in which employee with highest Basic works.

Solution:

select dept\_code, basic as MAX\_BASIC from employee

where basic = (select max(basic) from employee);

MariaDB [assignment\_1]> select dept\_code, basic as MAX\_BASIC from employee

-> where basic = (select max(basic) from employee);

+-----------+-----------+

| dept\_code | MAX\_BASIC |

+-----------+-----------+

| PUR | 50000 |

+-----------+-----------+

Question 12:

Find the Designation description of the employee with highest basic.

Solution:

select d.desig\_desc, e.basic as MAX\_BASIC

from employee e, designation d

where e.desig\_code = d.desig\_code and

e.basic = (select max(basic) from employee);

MariaDB [assignment\_1]> select d.desig\_desc, e.basic as MAX\_BASIC

-> from employee e, designation d

-> where e.desig\_code = d.desig\_code and

-> e.basic = (select max(basic) from employee);

+------------+-----------+

| desig\_desc | MAX\_BASIC |

+------------+-----------+

| Manager | 50000 |

+------------+-----------+

Question 13:

Find the no. of managers in each department.

Solution:

select d.dept\_name, count(\*) as NO\_OF\_MANAGER

from employee e, department d where e.dept\_code = d.dept\_code

and e.desig\_code in (select desig\_code from designation

where desig\_desc = 'Manager');

MariaDB [assignment\_1]> select d.dept\_name, count(\*) as NO\_OF\_MANAGER

-> from employee e, department d where e.dept\_code = d.dept\_code

-> and e.desig\_code in (select desig\_code from designation

-> where desig\_desc = 'Manager');

+-----------+---------------+

| dept\_name | NO\_OF\_MANAGER |

+-----------+---------------+

| Purchase | 1 |

+-----------+---------------+

Question 14:

Find the maximum basic from EMP table without using MAX().

Solution:

select distinct basic as Max\_Basic from employee e

where e.basic >= all(select basic from employee);

MariaDB [assignment\_1]> select distinct basic as Max\_Basic from employee e

-> where e.basic >= all(select basic from employee);

+-----------+

| Max\_Basic |

+-----------+

| 50000 |

+-----------+

Question 15:

Find the minimum basic from EMP table without using MIN().

Solution:

select distinct basic as Min\_Basic from employee e

where e.basic <= all(select basic from employee);

MariaDB [assignment\_1]> select distinct basic as Min\_Basic from employee e

-> where e.basic <= all(select basic from employee);

+-----------+

| Min\_Basic |

+-----------+

| 32500 |

+-----------+

Question 16:

Find the name of the department with highest total basic. Do the same for highest average basic and maximum no. of employee.

Solution:

Part – 1

select d.dept\_name, e.basic from employee e, department d

where e.dept\_code = d.dept\_code

and e.basic >= all(select basic from employee);

MariaDB [assignment\_1]> select d.dept\_name, e.basic from employee e, department d

-> where e.dept\_code = d.dept\_code

-> and e.basic >= all(select basic from employee);

+-----------+-------+

| dept\_name | basic |

+-----------+-------+

| Purchase | 50000 |

+-----------+-------+

Part - 2

select dept\_name, basic as AVERAGE\_BASIC from (

select dept\_name, avg(basic) as Basic from employee e, department d

where e.dept\_code = d.dept\_code group by dept\_name)

where basic = (select max(basic) from (

select dept\_name, avg(basic) as Basic from employee e, department d

where e.dept\_code = d.dept\_code group by dept\_name)

);

+-----------+-----------+

| dept\_name | avg\_basic |

+-----------+-----------+

| Purchase | 45000 |

+-----------+-----------+

Part – 3

select dept\_name, count(emp\_code) as Count\_of\_Employee from employee, department

where employee.dept\_code = department.dept\_code

group by employee.dept\_code order by count(emp\_code) desc limit 1;

MariaDB [assignment\_1]> select dept\_name, count(emp\_code) as Count\_of\_Employee from employee, department

-> where employee.dept\_code = department.dept\_code

-> group by employee.dept\_code order by count(emp\_code) desc limit 1;

+-----------+-------------------+

| dept\_name | Count\_of\_Employee |

+-----------+-------------------+

| Purchase | 3 |

+-----------+-------------------+

Question 17:

Insert same rows into EMP table with designation code not existing in DESIGNATION table.

Solution:

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, JN\_DT)

VALUES ('009', 'Adam Smith', 'PUR', 'Manager', 'M', '123 Main St', 'Berkeley', 'CA', '90001', 50000, '2002-01-01');

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, JN\_DT)

VALUES ('010', 'Adam Smith', 'PUR', 'Worker', 'M', '123 Main St', 'Berkeley', 'CA', '90001', 50000, '2002-01-01');

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, JN\_DT)

VALUES ('011', 'Adam Smith', 'PUR', 'Client', 'M', '123 Main St', 'Berkeley', 'CA', '90001', 50000, '2002-01-01');

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, DESIG\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, JN\_DT)

VALUES ('012', 'Adam Smith', 'PUR', 'Executive', 'M', '123 Main St', 'Berkeley', 'CA', '90001', 50000, '2002-01-01');

MariaDB [assignment\_1]> select \* from employee;

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | DESIG\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT |

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

| 001 | Adam Smith | PUR | MGR | M | 123 Main St | BERKELEY | CA | 90001 | 50000 | 2002-01-01 |

| 002 | Maddie Morris | PRO | EXE | F | 456 Elm St | DENVER | CO | 80001 | 32500 | 0000-00-00 |

| 003 | Edward Johnson | PUR | OFF | M | 789 Oak St | ATLANTA | GA | 30002 | 40000 | 2010-06-25 |

| 004 | Jenifer Martin | MGR | OFF | F | 245 Perl St | NASHVILLE | TN | 80001 | 37010 | 2008-01-24 |

| 005 | Sam Joe | FIN | CLK | M | 245 New St | DENVER | CO | 80001 | 35000 | 2002-02-24 |

| 006 | Jessica Altman | PUR | HLP | F | 245 Old St | ATLANTA | GA | 80001 | 45000 | 2003-05-22 |

| 007 | Sarra Jhones | PRO | CLK | F | 160 Saint St | NASHVILLE | TN | 20001 | 48000 | 2001-06-18 |

| 008 | Maxine | FIN | CLK | F | 160 Perl St | NASHVILLE | TN | 20521 | 48000 | 2002-06-18 |

| 009 | Adam Smith | PUR | Manager | M | 123 Main St | Berkeley | CA | 90001 | 50000 | 2002-01-01 |

| 010 | Adam Smith | PUR | Worker | M | 123 Main St | Berkeley | CA | 90001 | 50000 | 2002-01-01 |

| 011 | Adam Smith | PUR | Client | M | 123 Main St | Berkeley | CA | 90001 | 50000 | 2002-01-01 |

| 012 | Adam Smith | PUR | Executive | M | 123 Main St | Berkeley | CA | 90001 | 50000 | 2002-01-01 |

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

Question 18:

Delete the rows from EMP table with invalid DESIG\_CODE.

Solution:

delete from employee where desig\_code not in

(select desig\_code from designation);

MariaDB [assignment\_1]> delete from employee where desig\_code not in

-> (select desig\_code from designation);

Query OK, 4 rows affected (0.005 sec)

MariaDB [assignment\_1]> select \* from employee;

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | DESIG\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT |

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

| 001 | Adam Smith | PUR | MGR | M | 123 Main St | BERKELEY | CA | 90001 | 50000 | 2002-01-01 |

| 002 | Maddie Morris | PRO | EXE | F | 456 Elm St | DENVER | CO | 80001 | 32500 | 0000-00-00 |

| 003 | Edward Johnson | PUR | OFF | M | 789 Oak St | ATLANTA | GA | 30002 | 40000 | 2010-06-25 |

| 004 | Jenifer Martin | MGR | OFF | F | 245 Perl St | NASHVILLE | TN | 80001 | 37010 | 2008-01-24 |

| 005 | Sam Joe | FIN | CLK | M | 245 New St | DENVER | CO | 80001 | 35000 | 2002-02-24 |

| 006 | Jessica Altman | PUR | HLP | F | 245 Old St | ATLANTA | GA | 80001 | 45000 | 2003-05-22 |

| 007 | Sarra Jhones | PRO | CLK | F | 160 Saint St | NASHVILLE | TN | 20001 | 48000 | 2001-06-18 |

| 008 | Maxine | FIN | CLK | F | 160 Perl St | NASHVILLE | TN | 20521 | 48000 | 2002-06-18 |

+----------+----------------+-----------+------------+------+--------------+-----------+-------+-------+-------+------------+

Question 19:

Find the name of the female employees with basic greater than the average basic of their respective department.

Solution:

select e.emp\_name, e.basic from employee e, department d where

e.dept\_code = d.dept\_code and basic > (

select avg(basic) from employee e, department d where

e.dept\_code = d.dept\_code and e.sex = 'F'

) and e.sex = 'F';

MariaDB [assignment\_1]> select e.emp\_name, e.basic from employee e, department d where

-> e.dept\_code = d.dept\_code and basic > (

-> select avg(basic) from employee e, department d where

-> e.dept\_code = d.dept\_code and e.sex = 'F'

-> ) and e.sex = 'F';

+----------------+-------+

| emp\_name | basic |

+----------------+-------+

| Jessica Altman | 45000 |

| Sarra Jhones | 48000 |

| Maxine | 48000 |

+----------------+-------+

Question 20:

Find the number of female managers.

Solution:

select count(\*) as NO\_OF\_FEMALE\_MANAGER from employee e, designation d

where e.desig\_code = d.desig\_code and e.sex = 'F'

and d.desig\_code = "MGR";

MariaDB [assignment\_1]> select count(\*) as NO\_OF\_FEMALE\_MANAGER from employee e, designation d

-> where e.desig\_code = d.desig\_code and e.sex = 'F'

-> and d.desig\_code = "MGR";

+----------------------+

| NO\_OF\_FEMALE\_MANAGER |

+----------------------+

| 1 |

+----------------------+

Assignment: Set - 3

Assignment – 3

Question 1:

In an organization, number of departments exists. Each department has a name &amp; unique code. Number of employees work in each department. Each employee has unique employee code. Detailed information like name, address, city, basic, date of join are also stored. In a leave register for each employee leave records are kept showing leave type (CL/EL/ML etc.), from-date and to-date. When an employee retires or resigns then all the leave information pertaining to him are also deleted. Basic salary must be within Rs.5000 to Rs.9000. A department cannot be deleted if any employee record refers to it. Valid grades are A/B/C. Employee name must be in uppercase only. Default value for joining date is system date.

Design &amp; implement the tables with necessary constraints to support the scenario depicted above.

Solution:

Creating Department table:

CREATE TABLE DEPARTMENT (

DEPT\_CODE VARCHAR(16) PRIMARY KEY,

DEPT\_NAME VARCHAR(50) NOT NULL

);

MariaDB [assignment\_3]> DESC DEPARTMENT;

+-----------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+-----------+-------------+------+-----+---------+-------+

| DEPT\_CODE | varchar(16) | NO | PRI | NULL | |

| DEPT\_NAME | varchar(50) | NO | | NULL | |

+-----------+-------------+------+-----+---------+-------+

Creating Department table:

CREATE TABLE EMPLOYEE (

EMP\_CODE VARCHAR(16) PRIMARY KEY,

EMP\_NAME VARCHAR(50) CHECK (EMP\_NAME = upper(EMP\_NAME)),

DEPT\_CODE VARCHAR(16) NOT NULL,

SEX CHAR(1) CHECK (SEX IN ('M', 'F')),

ADDRESS VARCHAR(100),

CITY VARCHAR(50),

STATE VARCHAR(50),

PIN INTEGER,

BASIC INTEGER CHECK (BASIC BETWEEN 5000 AND 9000),

JN\_DT DATE DEFAULT (current\_date()),

GRADE CHAR(1) CHECK (GRADE IN ('A', 'B', 'C')),

FOREIGN KEY (DEPT\_CODE)

REFERENCES DEPARTMENT (DEPT\_CODE)

ON DELETE RESTRICT

);

MariaDB [assignment\_3]> DESC EMPLOYEE;

+-----------+--------------+------+-----+-----------+-------+

| Field | Type | Null | Key | Default | Extra |

+-----------+--------------+------+-----+-----------+-------+

| EMP\_CODE | varchar(16) | NO | PRI | NULL | |

| EMP\_NAME | varchar(50) | YES | | NULL | |

| DEPT\_CODE | varchar(16) | NO | MUL | NULL | |

| SEX | char(1) | YES | | NULL | |

| ADDRESS | varchar(100) | YES | | NULL | |

| CITY | varchar(50) | YES | | NULL | |

| STATE | varchar(50) | YES | | NULL | |

| PIN | int(11) | YES | | NULL | |

| BASIC | int(11) | YES | | NULL | |

| JN\_DT | date | YES | | curdate() | |

| GRADE | char(1) | YES | | NULL | |

+-----------+--------------+------+-----+-----------+-------+

Creating Leave\_Record table:

CREATE TABLE LEAVE\_RECORD (

LEAVE\_ID INTEGER PRIMARY KEY,

EMP\_CODE VARCHAR(16) NOT NULL,

LEAVE\_TYPE VARCHAR(10) CHECK (LEAVE\_TYPE IN ('CL', 'EL', 'ML')),

FROM\_DATE DATE,

TO\_DATE DATE,

FOREIGN KEY (EMP\_CODE)

REFERENCES EMPLOYEE (EMP\_CODE)

ON DELETE CASCADE

);

MariaDB [assignment\_3]> DESC LEAVE\_RECORD;

+------------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+------------+-------------+------+-----+---------+-------+

| LEAVE\_ID | int(11) | NO | PRI | NULL | |

| EMP\_CODE | varchar(16) | NO | MUL | NULL | |

| LEAVE\_TYPE | varchar(10) | YES | | NULL | |

| FROM\_DATE | date | YES | | NULL | |

| TO\_DATE | date | YES | | NULL | |

+------------+-------------+------+-----+---------+-------+

Inserting data in Department table:

INSERT INTO DEPARTMENT (DEPT\_CODE, DEPT\_NAME)

VALUES

('D001', 'Sales'),

('D002', 'Marketing'),

('D003', 'Finance');

MariaDB [assignment\_3]> SELECT \* FROM DEPARTMENT;

+-----------+-----------+

| DEPT\_CODE | DEPT\_NAME |

+-----------+-----------+

| D001 | Sales |

| D002 | Marketing |

| D003 | Finance |

+-----------+-----------+

Inserting data in Employee table:

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, SEX, ADDRESS, CITY, STATE, PIN, BASIC, GRADE)

VALUES

('E0010', 'John Doe', 'D001', 'M', '123 Main St', 'New York', 'NY', 12345, 6000, 'A'),

('E0011', 'Jane Smith', 'D001', 'F', '456 Elm St', 'Los Angeles', 'CA', 67890, 7000, 'B'),

('E0012', 'Michael Johnson', 'D002', 'M', '789 Oak St', 'Chicago', 'IL', 98765, 8000, 'C'),

('E0013', 'Emily Brown', 'D002', 'F', '321 Pine St', 'Houston', 'TX', 54321, 5500, 'A'),

('E0014', 'David Wilson', 'D003', 'M', '654 Cedar St', 'Miami', 'FL', 13579, 9000, 'B');

MariaDB [assignment\_3]> SELECT \* FROM EMPLOYEE;

+----------+-----------------+-----------+------+--------------+-------------+-------+-------+-------+------------+-------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT | GRADE |

+----------+-----------------+-----------+------+--------------+-------------+-------+-------+-------+------------+-------+

| E0010 | John Doe | D001 | M | 123 Main St | New York | NY | 12345 | 6000 | 2023-06-10 | A |

| E0011 | Jane Smith | D001 | F | 456 Elm St | Los Angeles | CA | 67890 | 7000 | 2023-06-10 | B |

| E0012 | Michael Johnson | D002 | M | 789 Oak St | Chicago | IL | 98765 | 8000 | 2023-06-10 | C |

| E0013 | Emily Brown | D002 | F | 321 Pine St | Houston | TX | 54321 | 5500 | 2023-06-10 | A |

| E0014 | David Wilson | D003 | M | 654 Cedar St | Miami | FL | 13579 | 9000 | 2023-06-10 | B |

+----------+-----------------+-----------+------+--------------+-------------+-------+-------+-------+------------+-------+

Inserting data in Leave\_record table:

INSERT INTO LEAVE\_RECORD (LEAVE\_ID, EMP\_CODE, LEAVE\_TYPE, FROM\_DATE, TO\_DATE)

VALUES

(1, 'E0010', 'CL', '2023-02-01', '2023-02-02'),

(2, 'E0011', 'EL', '2023-03-15', '2023-03-20'),

(3, 'E0012', 'ML', '2023-04-10', '2023-04-15'),

(4, 'E0013', 'CL', '2023-05-10', '2023-05-10'),

(5, 'E0014', 'EL', '2023-06-01', '2023-06-05');

MariaDB [assignment\_3]> SELECT \* FROM LEAVE\_RECORD;

+----------+----------+------------+------------+------------+

| LEAVE\_ID | EMP\_CODE | LEAVE\_TYPE | FROM\_DATE | TO\_DATE |

+----------+----------+------------+------------+------------+

| 1 | E0010 | CL | 2023-02-01 | 2023-02-02 |

| 2 | E0011 | EL | 2023-03-15 | 2023-03-20 |

| 3 | E0012 | ML | 2023-04-10 | 2023-04-15 |

| 4 | E0013 | CL | 2023-05-10 | 2023-05-10 |

| 5 | E0014 | EL | 2023-06-01 | 2023-06-05 |

+----------+----------+------------+------------+------------+

Question 2:

Try to violate the constraints that you have implemented in the table &amp; note, what happens. [Try with suitable INSERT/UPDATE/DELETE instruction]

Solution:

Violate foreign key integrity constraint:

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, SEX, BASIC)

VALUES ('E002', 'john doe', 'D0010', 'M', 5500);

ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails

(`assignment\_3`.`employee`, CONSTRAINT `employee\_ibfk\_1` FOREIGN KEY (`DEPT\_CODE`) REFERENCES `department` (`DEPT\_CODE`))

Violate salary range constraint:

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, SEX, BASIC)

VALUES ('E002', 'Jane Smith', 'D002', 'F', 9500);

ERROR 4025 (23000): CONSTRAINT `employee.BASIC` failed for `assignment\_3`.`employee`

Violate grade range constraint:

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, SEX, BASIC, GRADE)

VALUES ('E002', 'Jane Smith', 'D002', 'F', 9500, 'D');

ERROR 4025 (23000): CONSTRAINT `employee.BASIC` failed for `assignment\_3`.`employee`

Violate salary range constraint:

INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, SEX, BASIC, GRADE)

VALUES ('E002', 'jane Smith', 'D002', 'F', 3000, 'C');

ERROR 4025 (23000): CONSTRAINT `employee.BASIC` failed for `assignment\_3`.`employee`

Using update instruction:

select \* from employee;

update employee

set DEPT\_CODE = 'D010'

where emp\_code = 'E0010';

ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails

(`assignment\_3`.`employee`, CONSTRAINT `employee\_ibfk\_1` FOREIGN KEY (`DEPT\_CODE`) REFERENCES `department` (`DEPT\_CODE`))

Using delete instruction:

DELETE FROM DEPARTMENT WHERE DEPT\_CODE = 'D0010';

MariaDB [assignment\_3]> DELETE FROM DEPARTMENT WHERE DEPT\_CODE = 'D0010';

Query OK, 0 rows affected (0.001 sec)

Question 3:

1. Create a view showing employee code, name, dcode &amp; Basic For a particular department.
2. Try to ensure a row into the view with valid department &amp; also with invalid ones.
3. Find the newly inserted row in the table from which view was created.
4. Try to increment basic by Rs.100/.
5. Check it in the original table.
6. Delete the view.

Solution:

Part a:

MariaDB [assignment\_3]> CREATE VIEW EmployeeView AS

-> SELECT EMP\_CODE, EMP\_NAME, DEPT\_CODE, BASIC

-> FROM EMPLOYEE

-> WHERE DEPT\_CODE = 'D001';

Query OK, 0 rows affected

MariaDB [assignment\_3]> SELECT \* FROM EMPLOYEEVIEW;

+----------+------------+-----------+-------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | BASIC |

+----------+------------+-----------+-------+

| E0010 | John Doe | D001 | 6000 |

| E0011 | Jane Smith | D001 | 7000 |

+----------+------------+-----------+-------+

Part b:

MariaDB [assignment\_3]> INSERT INTO EMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_CODE, SEX, BASIC,

-> VALUES ('E003', 'janey Smith', 'D001', 'F', 8000, 'C');

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds

to your MariaDB server version for the right syntax to use near 'VALUES

('E003', 'janey Smith', 'D001', 'F', 8000, 'C')' at line 2

MariaDB [assignment\_3]> SELECT \* FROM EMPLOYEEVIEW;

+----------+------------+-----------+-------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | BASIC |

+----------+------------+-----------+-------+

| E0010 | John Doe | D001 | 6100 |

| E0011 | Jane Smith | D001 | 7000 |

+----------+------------+-----------+-------+

Part c:

MariaDB [assignment\_3]> SELECT \* FROM EMPLOYEE, EMPLOYEEVIEW

-> WHERE EMPLOYEE.EMP\_CODE = EMPLOYEEVIEW.EMP\_CODE;

+----------+------------+-----------+------+-------------+-------------+-------+-------+-------+------------+-------+----------+------------+-----------+-------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT | GRADE | EMP\_CODE | EMP\_NAME | DEPT\_CODE | BASIC |

+----------+------------+-----------+------+-------------+-------------+-------+-------+-------+------------+-------+----------+------------+-----------+-------+

| E0010 | John Doe | D001 | M | 123 Main St | New York | NY | 12345 | 6000 | 2023-06-10 | A | E0010 | John Doe | D001 | 6000 |

| E0011 | Jane Smith | D001 | F | 456 Elm St | Los Angeles | CA | 67890 | 7000 | 2023-06-10 | B | E0011 | Jane Smith | D001 | 7000 |

+----------+------------+-----------+------+-------------+-------------+-------+-------+-------+------------+-------+----------+------------+-----------+-------+

Part d:

MariaDB [assignment\_3]> UPDATE EMPLOYEE SET BASIC = BASIC + 100

-> WHERE DEPT\_CODE = 'D001';

Query OK, 2 rows affected

Rows matched: 2 Changed: 2 Warnings: 0

Part e:

MariaDB [assignment\_3]> SELECT \* FROM EMPLOYEE;

+----------+-----------------+-----------+------+--------------+-------------+-------+-------+-------+------------+-------+

| EMP\_CODE | EMP\_NAME | DEPT\_CODE | SEX | ADDRESS | CITY | STATE | PIN | BASIC | JN\_DT | GRADE |

+----------+-----------------+-----------+------+--------------+-------------+-------+-------+-------+------------+-------+

| E0010 | John Doe | D001 | M | 123 Main St | New York | NY | 12345 | 6100 | 2023-06-10 | A |

| E0011 | Jane Smith | D001 | F | 456 Elm St | Los Angeles | CA | 67890 | 7100 | 2023-06-10 | B |

| E0012 | Michael Johnson | D002 | M | 789 Oak St | Chicago | IL | 98765 | 8000 | 2023-06-10 | C |

| E0013 | Emily Brown | D002 | F | 321 Pine St | Houston | TX | 54321 | 5500 | 2023-06-10 | A |

| E0014 | David Wilson | D003 | M | 654 Cedar St | Miami | FL | 13579 | 9000 | 2023-06-10 | B |

+----------+-----------------+-----------+------+--------------+-------------+-------+-------+-------+------------+-------+

Part f:

DROP VIEW EmployeeView;

Question 4:

1. Create a view Showing empcode, name, deptname, basic, leave type, from date &amp; to date.
2. Try to insert a row in the view. Check what happens?
3. Try to increment basic by Rs.100.
4. Delete the view.

Solution:

Part a:

CREATE VIEW EMPLOYEELEAVEVIEW AS

SELECT EMPLOYEE.EMP\_CODE, EMPLOYEE.EMP\_NAME, DEPARTMENT.DEPT\_NAME, EMPLOYEE.BASIC,

LEAVE\_RECORD.LEAVE\_TYPE, LEAVE\_RECORD.FROM\_DATE, LEAVE\_RECORD.TO\_DATE

FROM EMPLOYEE, DEPARTMENT, LEAVE\_RECORD

WHERE EMPLOYEE.DEPT\_CODE = DEPARTMENT.DEPT\_CODE

AND EMPLOYEE.EMP\_CODE = LEAVE\_RECORD.EMP\_CODE;

MariaDB [assignment\_3]> select \* from EMPLOYEELEAVEVIEW;

+----------+-----------------+-----------+-------+------------+------------+------------+

| EMP\_CODE | EMP\_NAME | DEPT\_NAME | BASIC | LEAVE\_TYPE | FROM\_DATE | TO\_DATE |

+----------+-----------------+-----------+-------+------------+------------+------------+

| E0010 | John Doe | Sales | 6100 | CL | 2023-02-01 | 2023-02-02 |

| E0011 | Jane Smith | Sales | 7100 | EL | 2023-03-15 | 2023-03-20 |

| E0012 | Michael Johnson | Marketing | 8000 | ML | 2023-04-10 | 2023-04-15 |

| E0013 | Emily Brown | Marketing | 5500 | CL | 2023-05-10 | 2023-05-10 |

| E0014 | David Wilson | Finance | 9000 | EL | 2023-06-01 | 2023-06-05 |

+----------+-----------------+-----------+-------+------------+------------+------------+

Part b:

MariaDB [assignment\_3]> insert into EmployeeLeaveView (EMP\_CODE, EMP\_NAME, DEPT\_NAME, BASIC, LEAVE\_TYPE, FROM\_DATE, TO\_DATE)

-> values

-> ('E0015', 'jon snow', 'Sales', 7000, 'CL', '2023-02-01', '2023-02-02' );

ERROR 1393 (HY000): Can not modify more than one base table through a join view 'assignment\_3.employeeleaveview'

Part c:

MariaDB [assignment\_3]> UPDATE EMPLOYEE SET BASIC = BASIC + 100;

ERROR 4025 (23000): CONSTRAINT `employee.BASIC` failed for `assignment\_3`.`employee`

Part d:

DROP VIEW EmployeeLeaveView;

Question 5:

1. Create a table having empcode , Name, deptname, &amp; basic From the existing tables along with the records of the employee who are in a particular department (say, d1) and with a basic Rs. 7000/-
2. From the existing table, add the employees with the basic salary greater than or equal to 7000/-
3. Alter the table to add a net pay column.
4. Replace net pay with 1.5\* Basic.
5. Try to remove the net pay column.

[It may require no. of steps]

Solution:

Part a:

CREATE TABLE FILTEREDEMPLOYEE AS

SELECT EMPLOYEE.EMP\_CODE, EMPLOYEE.EMP\_NAME, DEPARTMENT.DEPT\_NAME, EMPLOYEE.BASIC

FROM EMPLOYEE, DEPARTMENT

WHERE EMPLOYEE.DEPT\_CODE = DEPARTMENT.DEPT\_CODE

AND DEPARTMENT.DEPT\_CODE = 'D001'

AND EMPLOYEE.BASIC = 7000;

MariaDB [assignment\_3]> SELECT \* FROM FILTEREDEMPLOYEE;

+----------+------------+-----------+-------+

| EMP\_CODE | EMP\_NAME | DEPT\_NAME | BASIC |

+----------+------------+-----------+-------+

| E0011 | Jane Smith | Sales | 7000 |

+----------+------------+-----------+-------+

Part b:

INSERT INTO FILTEREDEMPLOYEE (EMP\_CODE, EMP\_NAME, DEPT\_NAME, BASIC)

SELECT EMPLOYEE.EMP\_CODE, EMPLOYEE.EMP\_NAME, DEPARTMENT.DEPT\_NAME, EMPLOYEE.BASIC

FROM EMPLOYEE, DEPARTMENT

WHERE EMPLOYEE.DEPT\_CODE = DEPARTMENT.DEPT\_CODE AND BASIC >= 7000;

MariaDB [assignment\_3]> SELECT \* FROM FILTEREDEMPLOYEE;

+----------+-----------------+-----------+-------+

| EMP\_CODE | EMP\_NAME | DEPT\_NAME | BASIC |

+----------+-----------------+-----------+-------+

| E0011 | Jane Smith | Sales | 7000 |

| E0011 | Jane Smith | Sales | 7000 |

| E0012 | Michael Johnson | Marketing | 8000 |

| E0014 | David Wilson | Finance | 9000 |

+----------+-----------------+-----------+-------+

Part c:

ALTER TABLE FILTEREDEMPLOYEE

ADD COLUMN NET\_PAY INTEGER;

MariaDB [assignment\_3]> SELECT \* FROM FILTEREDEMPLOYEE;

+----------+-----------------+-----------+-------+---------+

| EMP\_CODE | EMP\_NAME | DEPT\_NAME | BASIC | NET\_PAY |

+----------+-----------------+-----------+-------+---------+

| E0011 | Jane Smith | Sales | 7000 | NULL |

| E0011 | Jane Smith | Sales | 7000 | NULL |

| E0012 | Michael Johnson | Marketing | 8000 | NULL |

| E0014 | David Wilson | Finance | 9000 | NULL |

+----------+-----------------+-----------+-------+---------+

Part d:

UPDATE FILTEREDEMPLOYEE

SET NET\_PAY = 1.5 \* BASIC;

MariaDB [assignment\_3]> SELECT \* FROM FILTEREDEMPLOYEE;

+----------+-----------------+-----------+-------+---------+

| EMP\_CODE | EMP\_NAME | DEPT\_NAME | BASIC | NET\_PAY |

+----------+-----------------+-----------+-------+---------+

| E0011 | Jane Smith | Sales | 7000 | 10500 |

| E0011 | Jane Smith | Sales | 7000 | 10500 |

| E0012 | Michael Johnson | Marketing | 8000 | 12000 |

| E0014 | David Wilson | Finance | 9000 | 13500 |

+----------+-----------------+-----------+-------+---------+

Part e:

ALTER TABLE FILTEREDEMPLOYEE

DROP COLUMN NET\_PAY;

MariaDB [assignment\_3]> SELECT \* FROM FILTEREDEMPLOYEE;

+----------+-----------------+-----------+-------+

| EMP\_CODE | EMP\_NAME | DEPT\_NAME | BASIC |

+----------+-----------------+-----------+-------+

| E0011 | Jane Smith | Sales | 7000 |

| E0011 | Jane Smith | Sales | 7000 |

| E0012 | Michael Johnson | Marketing | 8000 |

| E0014 | David Wilson | Finance | 9000 |

+----------+-----------------+-----------+-------+

Question 6:

Drop all the tables that you have created.

Solution:

DROP TABLE FILTEREDEMPLOYEE;

DROP TABLE LEAVE\_RECORD;

DROP TABLE EMPLOYEE;

DROP TABLE DEPARTMENT;

Assignment: Set - 4

Assignment – 4

Question 1:

1. Create EMP table with ECODE (primary key), ENAME, DCODE, GRADE, BASIC &amp; JN-DT as the columns. [Except BASIC &amp; JN-DT, all columns are of char type and site of Grade is 1.]
2. Insert number of rows.

Solution:

CREATE TABLE EMP (

ECODE VARCHAR(16) PRIMARY KEY,

ENAME VARCHAR(50),

DCODE VARCHAR(16),

GRADE CHAR(1),

BASIC INTEGER,

JN\_DT DATE

);

INSERT INTO EMP (ECODE, ENAME, DCODE, GRADE, BASIC, JN\_DT)

VALUES

('E001', 'John Doe', 'D001', 'A', 5000, '2022-01-01'),

('E002', 'Jane Smith', 'D002', 'B', 6000, '2022-02-15'),

('E003', 'Mike Johnson', 'D001', 'A', 5500, '2022-03-10'),

('E004', 'Emily Davis', 'D002', 'B', 7000, '2022-04-05'),

('E005', 'David Wilson', 'D003', 'A', 6500, '2022-05-20'),

('E006', 'Sarah Thompson', 'D003', 'B', 5500, '2022-06-12'),

('E007', 'Michael Brown', 'D001', 'C', 4800, '2022-07-03');

MariaDB [assignment\_4]> SELECT \* FROM EMP;

+-------+----------------+-------+-------+-------+------------+

| ECODE | ENAME | DCODE | GRADE | BASIC | JN\_DT |

+-------+----------------+-------+-------+-------+------------+

| E001 | John Doe | D001 | A | 5000 | 2022-01-01 |

| E002 | Jane Smith | D002 | B | 6000 | 2022-02-15 |

| E003 | Mike Johnson | D001 | A | 5500 | 2022-03-10 |

| E004 | Emily Davis | D002 | B | 7000 | 2022-04-05 |

| E005 | David Wilson | D003 | A | 6500 | 2022-05-20 |

| E006 | Sarah Thompson | D003 | B | 5500 | 2022-06-12 |

| E007 | Michael Brown | D001 | C | 4800 | 2022-07-03 |

+-------+----------------+-------+-------+-------+------------+

Question 2:

Change the column heading as shown below, So that in subsequent SELECT statement newly set heading will be shown:

ECODE EMPLOYEE CODE

ENAME NAME

DCODE DEPT.CODE

JN-DT JONING DATE

Solution:

ALTER TABLE EMP

RENAME COLUMN ECODE TO `EMPLOYEE CODE`,

RENAME COLUMN ENAME TO `NAME`,

RENAME COLUMN DCODE TO `DEPT.CODE`,

RENAME COLUMN JN\_DT TO `JOINING DATE`;

MariaDB [assignment\_4]> SELECT \* FROM EMP;

+---------------+----------------+-----------+-------+-------+----------------+

| EMPLOYEE CODE | NAME | DEPT.CODE | GRADE | BASIC | JOINING DATE |

+---------------+----------------+-----------+-------+-------+----------------+

| E001 | John Doe | D001 | A | 5000 | 2022-01-01 |

| E002 | Jane Smith | D002 | B | 6000 | 2022-02-15 |

| E003 | Mike Johnson | D001 | A | 5500 | 2022-03-10 |

| E004 | Emily Davis | D002 | B | 7000 | 2022-04-05 |

| E005 | David Wilson | D003 | A | 6500 | 2022-05-20 |

| E006 | Sarah Thompson | D003 | B | 5500 | 2022-06-12 |

| E007 | Michael Brown | D001 | C | 4800 | 2022-07-03 |

+---------------+----------------+-----------+-------+-------+----------------+

Question 3:

Set the format of columns as mentioned below, So that in subsequent SELECT statement, values appear in the specified format:

\*format of BASIC is such that a value of 7000 will be shown as7,000

\*Format of GRADE will be such that full column name appears in the display.

\*For JN-DT format is such that 01-JAN-00 will be shown as JANURY 01, 2000

Solution:

SELECT

ECODE,

ENAME,

DCODE,

FORMAT(BASIC, 0) as BASIC,

GRADE,

date\_format(JN\_DT, '%M-%d-%Y') as JN-DT

FROM EMP;

MariaDB [assignment\_4]> SELECT ECODE, ENAME, DCODE,

-> FORMAT(BASIC, 0) AS BASIC,

-> GRADE,

-> DATE\_FORMAT(JN\_DT, '%M-%d-%Y') AS JN\_DT

-> FROM EMP;

+-------+----------------+-------+-------+-------+------------------+

| ECODE | ENAME | DCODE | BASIC | GRADE | JN\_DT |

+-------+----------------+-------+-------+-------+------------------+

| E001 | John Doe | D001 | 5,000 | A | January-01-2022 |

| E002 | Jane Smith | D002 | 6,000 | B | February-15-2022 |

| E003 | Mike Johnson | D001 | 5,500 | A | March-10-2022 |

| E004 | Emily Davis | D002 | 7,000 | B | April-05-2022 |

| E005 | David Wilson | D003 | 6,500 | A | May-20-2022 |

| E006 | Sarah Thompson | D003 | 5,500 | B | June-12-2022 |

| E007 | Michael Brown | D001 | 4,800 | C | July-03-2022 |

+-------+----------------+-------+-------+-------+------------------+

Question 4:

1. Show the display attributes of all the columns.
2. Show the display attributes of particular column.
3. Suppress the newly set attributes of JN-DT .Try a select statement.
4. Reset the newly set attributes of JN-DT
5. Reset the newly set attributes of all columns.
6. Shown the display attributes of all columns.

Solution:

DESC EMP;

MariaDB [assignment\_4]> DESC EMP;

+-------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+-------+-------------+------+-----+---------+-------+

| ECODE | varchar(16) | NO | PRI | NULL | |

| ENAME | varchar(50) | YES | | NULL | |

| DCODE | varchar(16) | YES | | NULL | |

| GRADE | char(1) | YES | | NULL | |

| BASIC | int(11) | YES | | NULL | |

| JN\_DT | date | YES | | NULL | |

+-------+-------------+------+-----+---------+-------+

DESC EMP ENAME;

MariaDB [assignment\_4]> DESC EMP ENAME;

+-------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+-------+-------------+------+-----+---------+-------+

| ENAME | varchar(50) | YES | | NULL | |

+-------+-------------+------+-----+---------+-------+

ALTER TABLE EMP MODIFY COLUMN JOINING\_DATE DATE FORMAT 'YYYY-MM-DD';

ALTER TABLE EMP MODIFY COLUMN EMPLOYEE\_CODE INT;

ALTER TABLE EMP MODIFY COLUMN NAME VARCHAR(20);

ALTER TABLE EMP MODIFY COLUMN DEPARTMENT\_CODE CHAR(1);

ALTER TABLE EMP MODIFY COLUMN GRADE CHAR(1);

ALTER TABLE EMP MODIFY COLUMN BASIC FLOAT;

ALTER TABLE EMP MODIFY COLUMN JOINING\_DATE DATE;

MariaDB [ASSIGNMENT\_4]> DESC EMP;

+-------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+-------+-------------+------+-----+---------+-------+

| ECODE | int(1) | NO | PRI | NULL | |

| ENAME | varchar(20) | YES | | NULL | |

| DCODE | varchar(10) | YES | | NULL | |

| GRADE | char(1) | YES | | NULL | |

| BASIC | float | YES | | NULL | |

| JN\_DT | date | YES | | NULL | |

+-------+-------------+------+-----+---------+-------+