Assignment 4

1. Create a database with Employee System.

- CREATE DATABASE EmployeeSystem.

2. Creating tables based on ER diagram.

- CREATE TABLE Employee (employee_id int NOT NULL AUTO_INCREMENT, first_name char(50) NOT NULL, last_name char(50) NOT NULL, gender char(6) NOT NULL, age int NOT NULL, email char(50) NOT NULL, designation char(50) NOT NULL, hire_date date NOT NULL, resigned_date date, address char(100), PRIMARY KEY (employee_id));
- CREATE TABLE Department (department_id int NOT NULL AUTO_INCREMENT, name char(50) NOT NULL, description char(100), PRIMARY KEY (department_id));
- CREATE TABLE Project (project_id int NOT NULL AUTO_INCREMENT, name char(50) NOT NULL, description char(100), PRIMARY KEY (project_id));
- CREATE TABLE Salary (salary_id int NOT NULL AUTO_INCREMENT, issue_date DATE NOT NULL, amount float NOT NULL, bonus float, PRIMARY KEY (salary_id));

3. Adding 20 employees.

- INSERT INTO Employee (first_name, last_name, gender, age, email, designation,hire_date,resigned_date,address)
  values ('Ramita', 'Shrestha', 'female', '35', 'Ramita@gmail.com', 'Prodct Engineer','2021-5-20',NULL,'birgunj'),
  ('Tom', 'Hamal', 'male', '24', 'Tom@gmail.com', 'Mechanical Engineer','2021-6-20',NULL,'birgunj'),
  ('Komal', 'Panday', 'female', '28', 'Komal@gmail.com', 'Software Engineer','2021-7-20',NULL,'pokhara'),
  ('Madhu', 'Panday', 'female', '21', 'madhu@gmail.com', 'Software Engineer','2021-8-20',NULL,'birgunj'),
  ('Suman', 'Rana', 'male', '25', 'suman@gmail.com', 'Data Scientist','2000-08-30.',NULL,'Kathmandu'),
  ('Tp', 'gupta', 'male', '21', 'tp@gmail.com', 'Tester','2021-11-20',NULL,'Kathmandu'),
  ('ramesh', 'Shrestha', 'male', '23', 'ramesh@gmail.com', 'Jr. Software Engineer','2021-6-14',NULL,'birgunj'),
  ('Ramaneshwor', 'gupta', 'male', '23', 'Ramaneshwor@gmail.com', 'Senior Software Engineer','2021-2-20',NULL,'birgunj'),

('rose', 'lama', 'female', '29', 'rose@gmail.com', ' Manager','2000-08-30',NULL,'birgunj'),
('rashi', 'gupta', 'female', '26', 'rashi@gmail.com', 'Tester','2021-9-20',NULL,'birgunj'),
('Aishwarya', 'Gurung', 'female', '21', 'aishwarya@gmail.com', 'Manager','2021-6-20',NULL,'birgunj'),
('Tapu', 'gupta', 'male', '29', 'tapu@gmail.com', 'Developer','2021-9-20',NULL,'birgunj'),
('Dinesh', 'Gajurel', 'male', '21', 'dinesh531@gmail.com', 'Coder','2021-5-20',NULL,'birgunj'),
('Reshma', 'Gurung', 'female', '21', 'reshma@gmail.com', 'Tester','2021-7-20',NULL,'birgunj'),
('Tommy', 'Sharma', 'male', '21', 'tommy@gmail.com', 'Doctor','2021-8-20',NULL,'birgunj'),
('Tushar', 'gupta', 'male', '21', 'tushar531@gmail.com', 'Software Engineer','2021-7-20','2021-9-14','birgunj'),
('Ajay', 'gupta', 'male', '21', 'ajay531@gmail.com', 'Software Engineer','2021-11-14',NULL,'birgunj'),
('ratan', 'gupta', 'male', '21', 'ratan531@gmail.com', 'Manger','2021-2-20',NULL,'bhaktapur'),
('Ram', 'gupta', 'male', '21', 'Ram531@gmail.com', 'Manger','2021-2-20',NULL,'bhaktapur'),
('Shyam', 'gupta', 'male', '21', 'shyam531@gmail.com', 'Manger','2021-2-20',NULL,'bhaktapur');

4. Adding salary of each employee.

- ALTER TABLE Salary ADD employee_id int;
  ALTER TABLE Salary ADD FOREIGN KEY (employee_id) REFERENCES Employee(employee_id);

  INSERT INTO Salary (issue_date,amount,bonus,employee_id)
  values ('2019-8-20',15000,NULL,1),
  ('2020-11-12',20000,NULL,2),
  ('2020-9-12',12000,NULL,3),
  ('2020-10-12',35000,NULL,4),
  ('2020-10-12',14000,NULL,5),
  ('2020-10-12',20000,NULL,6),
  ('2020-10-12',7000,NULL,7),
  ('2020-10-12',20000,NULL,8),
  ('2020-7-12',20000,NULL,9),
  ('2020-10-12',25000,NULL,10),
  ('2020-10-12',20000,NULL,11),
  ('2020-10-12',20000,NULL,12),

('2020-10-12',20000,NULL,13),
('2020-10-12',20000,NULL,14),
('2020-10-12',20000,NULL,15),
('2020-10-12',20000,NULL,16),
('2020-10-12',20000,NULL,17),
('2020-10-12',20000,NULL,18),
('2020-10-12',20000,NULL,19),
('2020-10-12',20000,NULL,20);

5. Adding departments with employees working in it.

- ALTER TABLE Employee ADD dep_id int;

  ALTER TABLE Employee ADD FOREIGN KEY (dep_id) REFERENCES Department(department_id);

  INSERT INTO Department (name,description) values ('Marketting','This is marketting Department'),('IT', 'This is IT Department');

  update Employee set dep_id = 2 where employee_id BETWEEN 1 AND 10;
  update Employee set dep_id = 1 where employee_id BETWEEN 11 AND 20;

6. Adding 7 projects.

- ALTER TABLE Project ADD employee_id int;
  ALTER TABLE Project ADD FOREIGN KEY (employee_id) REFERENCES Employee(employee_id);

  INSERT INTO Project (name,description,employee_id)
  values ('Anta project',NULL,1),
  ('Ajax project','Develop A website',1),
  ('gopikrishna website',NULL,2),
  ('nashville website','debvelop a website for nashvill',2),
  ('front',NULL,20),
  ('software',NULL,17),
  ('back',NULL,5);

7. Moving 3 employees to another department.

- update Employee set dep_id = 1 where employee_id=1;
  update Employee set dep_id = 1 where employee_id=2;
  update Employee set dep_id = 1 where employee_id=3;

8. Adding resigned date for 2 employees.

- update Employee set resigned_date = '2021-10-22' where employee_id=1;

update Employee set resigned_date = '2021-10-22' where employee_id=2;

9. Showing details of employees whose first name start with ‘R’ or ‘r’.
   - SELECT * FROM Employee WHERE first_name LIKE 'R%' OR first_name LIKE 'r%';

10. Showing details of employees who works in more than one project.
    - SELECT emp.*,COUNT(p.employee_id) FROM Employee AS emp JOIN Project AS p ON emp.employee_id=p.employee_id GROUP BY p.employee_id HAVING COUNT(p.employee_id)>1;

11. Counting number of employees who have less than 20000 salaries.
    - SELECT COUNT(employee_id) as totalEmployees FROM salary where amount<20000;

12. Incrementing salary of all employees by 10%.
    - UPDATE Salary SET amount=(amount*0.1)+amount;

13. Giving bonus of 10% to all employee hired before 2000-09-30.
    - UPDATE Salary s LEFT JOIN Employee emp ON s.employee_id = emp.employee_id set bonus=amount*0.1 WHERE emp.hire_date<'2000-09-30';

14. Finding the average salary of each department, number of employees working on that department.
    - select emp.dep_id,avg(s.amount) as AverageSalary,count(emp.employee_id) as NumofEmp FROM Employee emp inner join Salary s ON emp.employee_id=s.employee_id group by emp.dep_id;

15. Selecting the employees from each department which has a maximum salary.
    - SELECT emp.*, max(s.amount) as maxsalary FROM Employee AS emp JOIN Salary s ON emp.employee_id=s.employee_id GROUP BY emp.dep_id;

16. Selecting the employee from each department which has a maximum salary without using group by clause.
    - CREATE VIEW EmployeeSalary AS SELECT Employee.*, Salary.Amount AS Salary FROM Employee JOIN Salary ON Employee.employee_id = Salary.employee_id;

      SELECT Department.name AS 'Department', EmployeeSalary.first_name AS Employee, EmployeeSalary.Salary FROM Department, EmployeeSalary WHERE Department.department_id = EmployeeSalary.dep_id AND EmployeeSalary.Salary = (SELECT MAX(EmployeeSalary.Salary) FROM EmployeeSalary WHERE EmployeeSalary.dep_id = Department.department_id);

17. Checking what happens when you want to delete an employee who has resigned; what needs to be done to delete.

- DELETE FROM Employee Where resigned_date IS NOT NULL;
- Error: ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`employeesystem`.`project`, CONSTRAINT `project_ibfk_1` FOREIGN KEY (`employee_id`) REFERENCES `employee` (`employee_id`))
- To avoid these constraint errors during table deletion we can drop table in the correct order i.e., child table first and parent table as the last one. In case of a loop in foreign keys, we can remove this loop and redefine tables structure before dropping tables, we can also temporarily set "FOREIGN_KEY_CHECKS=0" and drop the table and again put "FOREIGN_KEY_CHECKS=1".