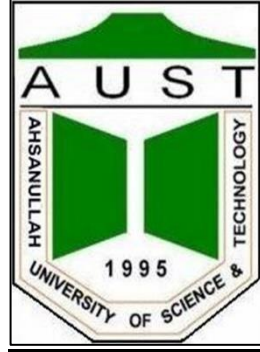


AHSANULLAH UNIVERSITY OF SCIENCE AND TECHNOLOGY



Department of Computer Science and Engineering Program: BSc in Computer Science and Engineering

Assignment No: 03

Date of Submission: 28/4/24

Submitted to

Ms. Nawshin Tabassum Tanny

Lecturer (Grade-I), Department of CSE, AUST. Mr. Md.

Siam Ansary

Lecturer (Grade-I), Department of CSE, AUST.

Submitted by,

Name: Mushfiqur Rahman Sajid

Student ID: 20210204004

Lab Section: A1

CSE 3104 – Assignment 03

1. For the tables salesman (sId, sName, sCity, sCommission) and customer (cId, cName, cCity, sId), write a query to find the names of the Customers and Salespersons who reside in the same city.
2. For the tables salesman (sId, sName, sCity, sCommission) and customer (cId, cName, cCity, sId), write a query to find the names of the Customers a Salesman represents.
3. For the tables salesman (sId, sName, sCity, sCommission) and customer (cId, cName, cCity, sId), write a query to find the names of the Salespersons who do not live in the same city as the Customers they represent.
4. For the tables salesman (sId, sName, sCity, sCommission) and customer (cId, cName, cCity, sId), write a query to display the names of all customers along with their representing salespersons.
5. For the tables salesman (sId, sName, sCity, sCommission) and customer (cId, cName, cCity, sId), write a query to display the names of all salespersons who represents one or more customers or still have not started representing any.
6. For the tables company (cId, cName) and products (pId, pName, price, cId), write a query to display the products' names, prices and producing company names.
7. For the tables company (cId, cName) and products (pId, pName, price, cId), write a query to display names of companies along with their products' average price.
8. For the tables department (dId, dName, dBudget) and employee (eId, fName, lName, dId), write a query to display names of the employees along with their department name.
9. For the tables department (dId, dName, dBudget) and employee (eId, fName, lName, dId), write a query to display names of the employees whose departments' budgets are greater than fifty thousand.
10. For the tables department (dId, dName, dBudget) and employee (eId, fName, lName, dId), write a query to display names of the departments that has more than two employees.

N.B.:

- Primary keys are underlined and foreign keys have a dashed underline.
- Queries of more than one question will not be the same.

Solution:

```
--drop database my_db;  
CREATE DATABASE my_db;
```

```
CREATE TABLE salesman  
(  
    sId int PRIMARY KEY,  
    sName varchar(200),  
    sCity varchar(200),  
    sCommission int  
);
```

```
insert into salesman values (1, 'Rohim', 'Italy',100);  
insert into salesman values (4, 'Maya', 'BANGLADESH',10);  
insert into salesman values (2, 'Vikrom', 'Italy',109);  
insert into salesman values (3, 'Lily', 'UK',2000);  
insert into salesman values (5, 'Manir', 'UK',2);  
insert into salesman values (6, 'Liam', 'INDIA',10);  
insert into salesman values (7, 'Brian', 'USA',20);
```

```
--drop table salesman;  
--drop table customer;
```

```
CREATE TABLE customer  
(  
    cId int PRIMARY KEY,  
    cName varchar(200),  
    cCity varchar(200),  
    sId int  
);
```

```
insert into customer values (1, 'Ashik', 'Italy',1);  
insert into customer values (4, 'Tonni', 'Italy',1);  
insert into customer values (2, 'Riaz', 'Italy',3);  
insert into customer values (3, 'Omi', 'UK',4);  
insert into customer values (5, 'Tanim', 'UK',3);  
insert into customer values (6, 'Priya', 'UK',6);  
insert into customer values (7, 'Shishir', 'UK',1);  
insert into customer values (8, 'Josh', 'UK',100);
```

1. `SELECT salesman.sName , customer.cName,salesman.sCity from customer inner join salesman on sCity=cCity;`
2. `SELECT customer.cName from customer inner join salesman on customer.sId=salesman.sId ;`
3. `SELECT salesman.sName from customer inner join salesman on (customer.sId=salesman.sId and cCity<>sCity) ;`
4. `SELECT customer.cName, salesman.sName from customer left join salesman on customer.sId=salesman.sId ;`

```

5. SELECT salesman.sName ,count(customer.sId) as noOfCustomerTheyRepresent from salesman
left join customer on customer.sId=salesman.sId group by salesman.sName order by
count(customer.sId) DESC;

--drop table company;
--drop table products;

CREATE TABLE company
(
  cId int PRIMARY KEY,
  cName varchar(200);
);

INSERT INTO company values(1, 'Amazon');
INSERT INTO company values(3, 'Bkash');
INSERT INTO company values(2, 'CoffeeClub');
INSERT INTO company values(4, 'Danish');

CREATE TABLE products
(
  pId int PRIMARY KEY,
  pName varchar(200),
  Price int,
  cId int FOREIGN KEY REFERENCES company(cId)
);

INSERT INTO products values(1, 'Apple', 100, 3);
INSERT INTO products values(3, 'Banana', 200, 1);
INSERT INTO products values(2, 'Chocolate', 50, 2);
INSERT INTO products values(4, 'Capachino', 10, 2);

6. select products.pName as Product_name , products.Price as product_price ,
company.cName as producing_company from company right join products on
products.cId=company.cId;

7. select company.cName ,avg(Price) as Average_Price from products right join company on
products.cId=company.cId group by company.cName ;

--drop table department;
--drop table employee;

CREATE TABLE department
(
  dId int PRIMARY KEY,
  dName varchar(200),
  dBudget int,
);

INSERT INTO department values(1, 'AUST', 100);
INSERT INTO department values(3, 'NETFLIX', 200);
INSERT INTO department values(2, 'APPLE', 50);
INSERT INTO department values(4, 'GOOGLE', 10);
INSERT INTO department values(5, 'MICROSOFT', 100000);

```

```
CREATE TABLE employee
(
    eId int PRIMARY KEY,
    fName varchar(200),
    lName varchar(200),
    dId int FOREIGN KEY REFERENCES department(dId)
);
```

```
INSERT INTO employee values(1, 'Josh', 'Butler', 1);
INSERT INTO employee values(3, 'Jack', 'Kalis', 2);
INSERT INTO employee values(2, 'Chris', 'Gayle', 1);
INSERT INTO employee values(4, 'Jhon', 'Cena', 2);
INSERT INTO employee values(5, 'Bruce', 'Banner', 5);
INSERT INTO employee values(6, 'Peter', 'Perker', 1);
```

8. `select employee.fname, employee.lname, department.dName from employee left join department on employee.dId=department.dId;`

9. `select employee.fname, employee.lname from employee inner join department on department.dId=employee.dId and department.dBudget>50000;`

10. `select department.dName , count(employee.dId) as noOfEmployee from employee inner join department on(employee.dId=department.dId) group by employee.dId, department.dName having count(employee.dId)>2;`