**INFSCI 2710 Database Management, Fall 2024**

**Homework 2: SQL**

**Q1 [10 pt] Write an SQL query to return the department name and the average salary of the employees in each department. Order the result by average salary in descending order.**

**Query:**

select department\_name, avg(salary) average\_salary

from Departments d

join Employees e

on d.department\_id = e.department\_id

group by department\_name

order by average\_salary desc;

**Q2 [10 pt] Write an SQL query to return the name of employees who have worked on more than one project, along with the number of projects they have participated in.**

**Query:**

select e.employee\_name, count(ep.project\_id) as number\_of\_projects

from Employees e

join Employee\_project ep

on e.employee\_id = ep.employee\_id

group by e.employee\_id, e.employee\_name

having count(ep.project\_id) > 1;

**Q3 [10 pt] Write an SQL query to return each employee's name and the total hours they have worked across all projects.**

**Query:**

select e.employee\_name, sum(ep.hours\_worked) as total\_hours

from Employees e

join Employee\_project ep

on e.employee\_id = ep.employee\_id

group by e.employee\_id, e.employee\_name;

**Q4 [10 pt] Write an SQL query to return the name, position, and salary of the highest-paid employee.**

**Query:**

select employee\_name, position, salary

from Employees

where salary = (select max(salary) from Employees);

**Q5 [10 pt] Write an SQL query to return the name of all employees and the names of projects they are involved in. If an employee is not involved in any projects, display NULL for the project name.**

**Query:**

select e.employee\_name, p.project\_name

from Employees e

left join Employee\_project ep

on e.employee\_id = ep.employee\_id

left join Projects p

on ep.project\_id = p.project\_id;

**Q6 [10 pt] Write an SQL query to return all customers who have never left a review, showing their name and email.**

**Query:**

select c.customer\_name, c.email

from Customers c

left join Reviews r

on c.customer\_id = r.customer\_id

where r.customer\_id is NULL;

**Q7 [10 pt] Write an SQL query to return all hotel bookings that occurred between March 1, 2024, and June 30, 2024, showing the booking ID, customer name, hotel name, and total cost.**

**Query:**

Following query is written considering only check\_in when bookings done between the timelines

select booking\_id, customer\_name, hotel\_name, total\_cost

from Customers c

join Bookings b

on c.customer\_id = b.customer\_id

join Hotels h

on h.hotel\_id = b.hotel\_id

where check\_in between '2024-03-01' and '2024-06-30'

order by booking\_id, check\_in;

Following query is written considering if check\_in is done prior and check\_out booking is done between the timelines

select booking\_id, customer\_name, hotel\_name, total\_cost

from Customers c

join Bookings b

on c.customer\_id = b.customer\_id

join Hotels h

on h.hotel\_id = b.hotel\_id

where check\_in between '2024-03-01' and '2024-06-30'

or check\_out between '2024-03-01' and '2024-06-30'

order by booking\_id, check\_in;

**Q8 [10 pt] Write an SQL query to return the top 3 longest hotel stays by customers, showing the customer name, hotel name, and number of nights stayed.**

**Query:**

select c.customer\_name,

h.hotel\_name,

datediff(b.check\_out, b.check\_in) as nights\_stayed

from Customers c

join Bookings b

on c.customer\_id = b.customer\_id

join Hotels h

on b.hotel\_id = h.hotel\_id

where datediff(b.check\_out, b.check\_in) >= (

select min(nights\_stayed)

from (

select distinct datediff(check\_out, check\_in) as nights\_stayed

from Bookings

order by nights\_stayed desc limit 3) as top3

)

order by nights\_stayed desc;

**Q9 [10 pt] Write an SQL query to find all customers who have made more than one hotel booking, displaying the customer name and the number of bookings they have made.**

**Query:**

select c.customer\_name, count(booking\_id) as number\_of\_bookings

from Customers c

join Bookings b

on c.customer\_id = b.customer\_id

group by c.customer\_id

having count(booking\_id) > 1;