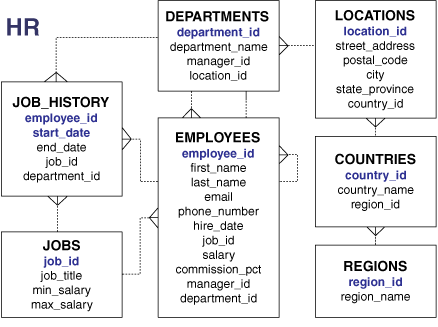
**Exercise 1:**

Given the following HR database diagram, Let’s write SQL statements to define these tables with all necessary integrity constraints (e.g. primary keys, foreign keys, domain contrains, etc)



create table JOBS (

-> job\_id int primary key,

-> job\_title varchar(20),

-> min\_salary decimal(10, 2),

-> max\_salary decimal(10, 2)

-> );

create table REGIONS(

-> region\_id int primary key,

-> region\_name varchar(20)

-> );

create table COUNTRIES(

-> country\_id int primary key,

-> country\_name varchar(20),

-> region\_id int not null,

-> FOREIGN KEY (region\_id) references REGIONS(region\_id)

-> );

**Exercise 2:**

Write SQL statements for the following duties:

1. Display the first\_name, last\_name of employees that work in departments located in “Hanoi”, ordered by last\_name in ascending order.

select first\_name, last\_name from employees e join departments d on e.department\_id = d.department\_id join locations l on l.location\_id = d.location\_id where l.CITY = 'Hanoi' order by last\_name ASC;

1. Create a query to display all the data from the Employees table.

select \* from employees;

1. Create a query to display the department number, department name, and manager number. Name the last column (manager number) heading as “MNG”.

select department\_id, department\_name, manager\_id "MNG" from departments;

1. Create a query to display the unique combination of values in department\_id and job\_id columns (Employees table).

select distinct concat(department\_id, " ", job\_id) from employees;

1. Display last\_name, first\_name of the employees that have the minimum salary w.r.t. a specific job title.

select last\_name, first\_name from employees e

-> join jobs j on j.job\_id = e.job\_id

-> where e.salary = j.min\_salary;

STRING FUNCTION

1. How many employees whose last\_name ends with “NH”.

select count(e.last\_name) from employees e where e.last\_name like "%nh";

1. Display the last\_name and the length of the last name for all employees where last name’s length is greater than 8 characters.

select e.last\_name, length(e.last\_name) length from employees e where length(e.last\_name) >= 8;

1. For all employees, display the last\_name, and username. The username will be composed from the first letter of first name concatenated with three last letters of last name concatenated with the string “2019”. (e.g. Trung Tran -> TTRA2019)

select last\_name, concat(left(first\_name, 1), right(last\_name, 3), "2019") username from employees ;

1. Create a query to display the last name concatenated with the first name, separated by space, and the telephone number concatenated with the email address, separated by hyphen. Name the column headings “FULL\_NAME” and “CONTACT\_DETAILS” respectively (Employees tables).

Select concat(first\_name, " ", last\_name) full\_name, concat(phone\_number, ";", email) contact\_details from employees;

NULL-RELATED FUNCTIONS

1. Display the first name, last name, phone number of all employees. Replace every null value in phone number with ‘N/A’.

select first\_name, last\_name, ifnull(phone\_number, 'N/A') from employees;

DATE FUNCTIONS

1. Display last name and first name of employees where the hired date is in “June”

select first\_name, last\_name from employees where month(hire\_date) = 6;

1. Display all the employees that were hired in the last 5 years.

select \* from employees where year(current\_date) - year(hire\_date) <= 5;

CASE FUNCTIONS

1. Display employee salary and a column “comment” whose value is set as follow:
   1. If salary >= 5000, comment = ‘Very high’
   2. If salary >= 3000 and salary < 5000, comment =’Hight’
   3. If salary < 300, comment = ‘low’

select salary, ( case when salary >= 5000 then "very high" when salary >= 3000 then "high" else "low" end) comment from employees ;

WHERE CLAUSE

1. Display the employee number, first name, job id and department number for all employees whose department number is not equal to 20, 60 and 80 (Employees table).

select employee\_id, first\_name, job\_id, department\_id from employees where department\_id not in (20, 60, 80);

1. Display all data from Employees table for all employees whose: salary is in the range of 6000 and 800 and their commission is not null

or department number is not equal to 80, 90 and 100 and their hire date is before January 1st, 1990.

select \* from employees where ((salary <= 6000 and salary >= 800) and commission\_pct is not null) or ( department\_id not in (80, 90, 100) and hire\_date < date '1990-01-01');

JOIN CLAUSE

1. For each department, display the department name, city, and state province.

select department\_name, city, state\_province from departments d join locations l on d.location\_id = l.location\_id;

1. For each employee, display the last name, and the manager’s last name.

select e1.last\_name last\_name, e2.last\_name manager\_lastname from

employees e1 join employees e2

on e2.employee\_id = e1.manager\_id ;

1. Display the last name and salary for all employees who earn less than employee number 103.

select last\_name, salary from employees where salary < (select salary from employees where employee\_id = 103);

AGGREGATE FUNCTIONS

1. Display total number of employees for department\_id = D122

select count(employee\_id) from employees

group by department\_id

having department\_id = "D122";

1. Display average salary for job\_id = J3224

select avg(salary) from employees group by job\_id having job\_id = "J3224";

GROUP BY CLAUSE

1. Display total number of employees for every departments.

select count(employee\_id) from employees group by department\_id;

1. Display total number of employees for every departments located in “Hanoi”.

select count(e.employee\_id) from employees e join departments d on e.department\_id = d.department\_id join locations l on l.location\_id = d.location\_id where l.city = "Hanoi" group by e.department\_id;

1. Display average salary for every departments that the total number of employees equal to 50.

select avg(salary) from employees

-> group by department\_id

-> having count(employee\_id) = 50;

1. Display the countries have more than 30 departments.

select country\_name, count( distinct (department\_id) ) from countries c join locations l using (country\_id) join departments d on d.location\_id = l.location\_id group by country\_name having count( distinct (department\_id)) >= 30;

1. Display the countries have both departments “ZARA” and “H&M”.

select distinct c.\* from countries c join locations l on c.country\_id = l.country\_id join departments d on d.location\_id = l.location\_id where d.department\_name in ("ZARA", "H&M") group by c.country\_id having count( distinct d.department\_name ) = 2;

1. Display the countries have departments “ZARA” or “H&M”.

select distinct c.\* from countries c join locations l on c.country\_id = l.country\_id join departments d on d.location\_id = l.location\_id where d.department\_name in ("ZARA", "H&M");

MISCELLANEOUS

1. Display the first name, salary, and department number for all employees who earn more than the minimum salary in department number 60 (Employees table).

select first\_name, salary, department\_id from employees where salary >= all(select min(salary) from employees where department\_id = 60);

1. Display the first name, salary, and department number for all employees who earn less than the average salary, and also work at the same department as employee whose first name is Kevin.

select first\_name, salary, department\_id from employees where salary < (select avg(salary) from employees) and department\_id in (select department\_id from employees where first\_name = "Kevin");

1. Display the countries that have no department “ZARA”.

select c.\* from countries c where c.country\_id not in (select l.country\_id from locations l join departments d using (location\_id) where department\_name = "ZARA");

1. Display any country that has the greatest number of departments

select c.\* from countries c where c.country\_id in (select country\_id from departments d join locations l using (location\_id) group by country\_id having count(department\_id) = (select count(department\_id) from departments d join locations l using(location\_id) group by country\_id order by count(department\_id) DESC limit 1));