

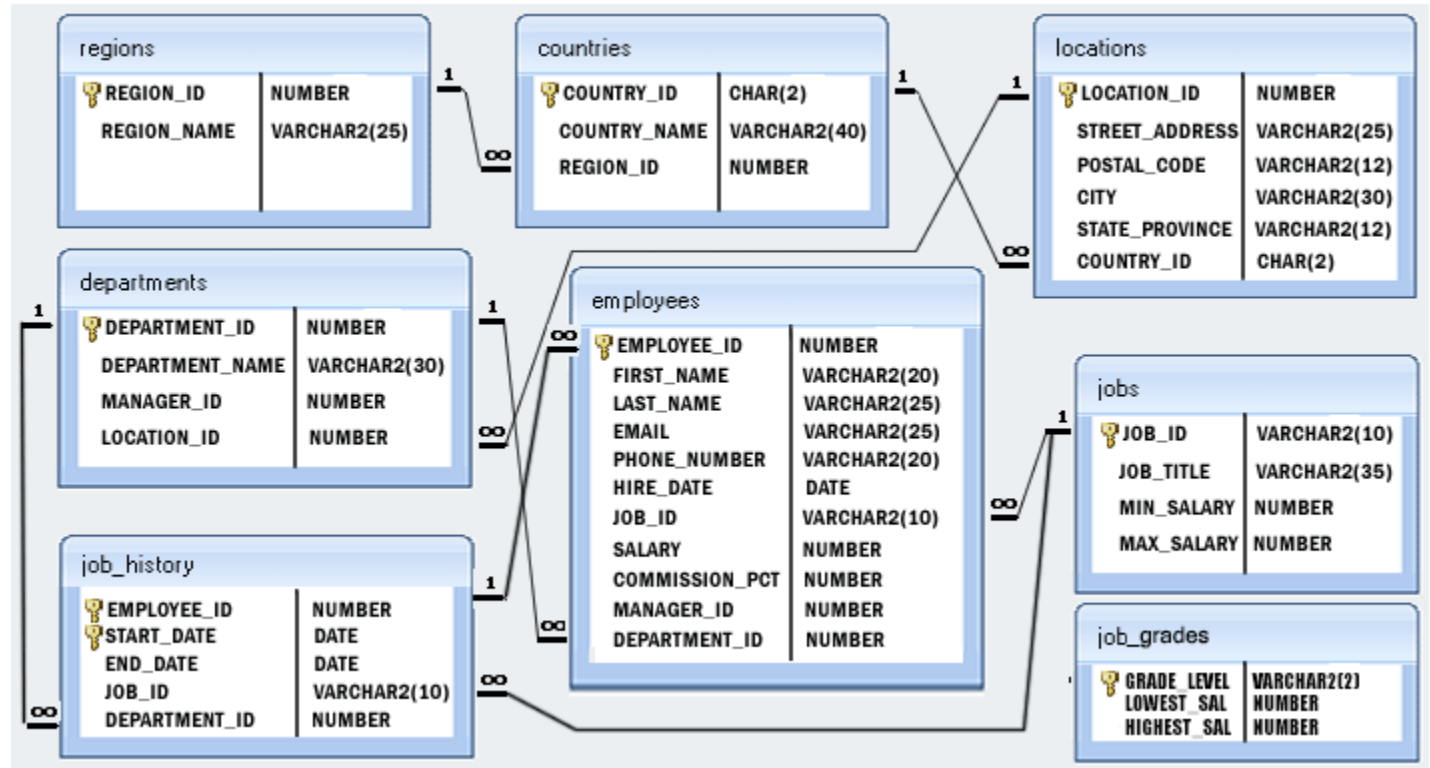
STAT 4260 – Databases
Spring 2018

Assignment 3 – [200 points]

For this assignment we will be using the provided database*. Follow the following steps to upload the **hr** database tables.

- Create a new database called **hr**; CREATE DATABASE hr;
- Link to the currently empty database; USE hr;
- Copy and paste the information from the hr.txt file into the command line to download the table information.

The following is the structure of the **hr** database:



It's good practice to make SQL keywords uppercase to distinguish them from other parts of your query. Please do this to make it easier for the grader. Also remember to include a semicolon at the end of your query. This tells SQL where the end of your query is!

[10pts] Electronic submission

1. [21 pts] Provide the SQL code to create the **regions**, **countries**, and **locations** tables with corresponding links between the three tables, making a number data type a integer no maximum length specified, and defining primary and foreign keys making sure they are correctly referenced. You do not need to provide the code that links **location** to the **department** table. Note for our purpose varchar2 is the same as varchar.

(If you are checking your code, keep in mind that the tables already exists so you may get an error, instead you should create a temporary database, e.g., CREATE DATABASE temp; USE temp; [check code]; DESC __;. Just remember to switch back to the hr database afterwards)

2. [6 pts] Provide the SQL code to add the following two rows to the **department** table: (These entries already exist so if you are checking your code you may run into errors in the **hr** database)

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800

3. [2 pts] Provide the SQL code to display the names (**first_name**, **last_name**) using alias names FirstName and LastName, respectively, from the **employees** table.
4. [2 pts] Provide the SQL code to get the unique department ID from the **employees** table.
5. [4 pts] Provide the SQL code to get the first and last names, salary and PF of all the employees (PF is calculated as 15% of salary) of all employees according to their first name in descending order.
6. [2 pts] Provide the SQL code to get the first and last names, salary of all employees in ascending order of salary.
7. [6 pts] Provide the SQL code to display the name (**first_name**, **last_name**) and **salary** for all employees whose salary is not in the range \$10,000 through \$15,000 and are in department 30 or 100.
8. [4 pts] Provide the SQL code to display the name (**first_name**, **last_name**) and hire date for all employees who were hired in 1987.
9. [7 pts] Provide the SQL code to display the **first_name** of all employees who have both "b" and "c" in their first name.
10. [8 pts] Provide the SQL code display the last name, job, and salary for all employees whose job is that of a IT Programmer (IT_PROG) or a Shipping Clerk (SH_CLERK), and whose salary is not equal to \$4,500, \$10,000, or \$15,000
11. [4 pts] Provide the SQL code to display the last name of employees whose names have exactly 5 characters.
12. [4 pts] Provide the SQL code to display the last name of employees having 'e' as the third character of their first name.

13. [4 pts] Provide the SQL code to display the last name and phone number of employees whose phone number contains the substring '124'.
14. [3 pts] Provide the SQL code to display the name (**first_name, last_name**), department ID and department name of all the employees.
15. [5 pts] Provide the SQL code to display the employee id, name (**last_name**) along with their manager id and manager name (**last_name**). Make sure to use column aliases to distinguish between the values.
16. [7 pts] Provide the SQL code to display the name (**first_name, last_name**), job, department ID, and department name of all the employees who work in London.
17. [3 pts] Provide the SQL code to display the employee ID, job title, number of days between ending date and starting date for all jobs in department 90 from job history.
18. [6 pts] Provide the SQL code to display department name, manager name, and city.
19. [4 pts] Provide the SQL code to display the employee id and department id from the **job_history** table for null job ids in chronological order of start date.
20. [4 pts] Provide the SQL code to display the employee id and department id from the **job_history** table for not null job ids in ascending order of department id.
21. [5 pts] Provide the SQL code to display the employee ID, job id, job title, and minimum salary all employee ids even if they have a null job id in the **job_history** table.
22. [5 pts] Provide the SQL code to display the employee ID, job title, and minimum salary all job titles even if there is no associated employee id.
23. [3 pts] Provide the SQL code to find the number of unique jobs in the **employees** table.
24. [2 pts] Provide the SQL code to find the number of jobs in the **employees** table.
25. [2 pts] Provide the SQL code to get the total salaries payable to employees.
26. [2 pts] Provide the SQL code to get the minimum salary from the **employees** table.
27. [3 pts] Provide the SQL code to get the maximum salary of an employee working as a Programmer (**IT_PROG**).
28. [5 pts] Provide the SQL code to average salary and number of employees working in department 90.
29. [8 pts] Provide the SQL code to get the highest, lowest, sum, and average salary of all employees rounded to the nearest integer.
30. [4 pts] Provide the SQL code to get the difference between the highest and lowest salary. Use a column alias for the output.