**EJERCICIOS CURSO SQL PARTE 2.**

1)-Seleccionar el nombre, apellido, id de job y departamentode los empleados que no tengan porcentaje de comision y que el salario sea menor o igual que 10000 y mayor o igual a 7000 (realizarlo de las dos formas posibles)

desc jobs;

desc employees;

desc departments;

select FIRST\_NAME, LAST\_NAME, JOB\_ID, DEPARTMENT\_NAME, SALARY, COMMISSION\_PCT from Employees AS emp inner join Departments as dep

on emp.department\_id = dep.department\_id

where (COMMISSION\_PCT is null or COMMISSION\_PCT = 0) and

(SALARY between 7000 and 10000)

-- otra manera (SALARY > 7000 and SALARY < 10000)

2)-Seleccionar el nombre, apellido, id de job de los empleados que el job\_id sea 'it\_prog','st\_man','pr\_rep' ordenados por id de job en forma ascendente.

desc employees;

select EMPLOYEE\_ID, LAST\_NAME, JOB\_ID, SALARY

from employees

WHERE SALARY <any (select SALARY from employees where job\_id = 'IT\_PROG')

order by JOB\_ID;

desc employees;

select EMPLOYEE\_ID, LAST\_NAME, JOB\_ID, SALARY

from employees

WHERE SALARY <any (select SALARY from employees where job\_id = 'ST\_MAN')

order by JOB\_ID;

desc employees;

select EMPLOYEE\_ID, LAST\_NAME, JOB\_ID, SALARY

from employees

WHERE SALARY <any (select SALARY from employees where job\_id = 'PR\_REP')

order by JOB\_ID;

3)-Seleccionar nombre,apellido,id de job, salario, % de comision, el salario total sumando la comisión de la tabla empleados donde el nombre sea Alberto

desc employees;

select FIRST\_NAME, LAST\_NAME, JOB\_ID, COMMISSION\_PCT, SALARY

from employees

WHERE FIRST\_NAME = 'Kimberely';

4)- Obtener de la tabla de empleados una lista de los departamentos, que no estén repetidos. Ordenarlo por id de departamento en forma ascendente.

desc employees;

select DEPARTMENT\_ID from employees;

select MIN(DEPARTMENT\_ID) FROM employees GROUP BY DEPARTMENT\_ID;

4.1)-Qué aparece en el último registro ?

DEPARTMENT\_ID = 110;

4.2)-Modificar la consulta para que aparezca en primer lugar.

desc employees;

select DEPARTMENT\_ID from employees;

select MIN(DEPARTMENT\_ID) FROM employees

GROUP BY DEPARTMENT\_ID

ORDER BY Department\_id DESC;

4.3)-Reescribir la consulta para reemplazar el valor nulo por -1

5)-Seleccionar id de manager, apellido de la tabla empleados en donde el id de job no sea 'SA\_REP','AD\_VP' el apellido empiece con la letra K pero que incluya tb los que tienen como manager al id 100. Ordenar por manager\_id en forma desc y por apellido en forma asc.

desc employees;

select MANAGER\_ID, LAST\_NAME, JOB\_ID from employees

WHERE JOB\_ID <any (select JOB\_ID from employees where job\_id = 'SA\_REP')

order by JOB\_ID;

select MANAGER\_ID, LAST\_NAME, JOB\_ID from employees

WHERE JOB\_ID >any (select JOB\_ID from employees where job\_id = 'AD\_VP')

order by JOB\_ID;

select MANAGER\_ID, LAST\_NAME, JOB\_ID from employees

WHERE LAST\_NAME like 'k%' or manager\_id = 100

order by MANAGER\_ID DESC;

select MANAGER\_ID, LAST\_NAME, JOB\_ID from employees

WHERE LAST\_NAME like 'k%' or manager\_id = 100

order by LAST\_NAME ASC;

6)-Contar para cada departamento cuantos empleados hay. Mostrar solo los registros que tengan departamento y que el departmento sea mayor o igual a 40. Ordenar los registros de forma que los que tengan mayor cantidad de empleados aparezcan primero.

desc employees;

SELECT COUNT(DEPARTMENT\_ID) AS Employees FROM Employees;

select DEPARTMENT\_ID, EMPLOYEE\_ID FROM EMPLOYEES

where DEPARTMENT\_ID > 40

ORDER BY EMPLOYEE\_ID DESC;

El siguiente query devuelve la misma cantidad de filas que el anterior ? Qué diferencia hay entre el count(\*) y el count(department\_id) ?

select count(department\_id)

from employees

where department\_id >=40

group by department\_id

order by 1 desc

La primera opción da la cantidad total de empleados con department\_id, mientras que la segunda opción da como resultado la suma de cada departmento\_id.

7)-Obtener de la tabla “Locations” la cantidad de locaciones que hay por país.

desc locations;

select count(location\_id)

from locations

group by city

order by 1 desc

8)-Construya una consulta que devuelva el salario medio, máximo y mínimo de los empleados agrupado por departamento.Etiquete las columnas como Maximo, Minimo, y Promedio. Sólo seleccionar los registros para el departamento 110 y 20.

desc locations;

SELECT MAX(salary), MIN(salary), AVG(salary) FROM Employees

where DEPARTMENT\_ID = 110 OR DEPARTMENT\_ID = 20;

8.1)-Ejecutar el siguiente comando que asigna valor nulo al salario de dos empleados. La idea es cómo se comportan los valores nulos en estas funciones. Ejecutar nuevamente la función. Tiene en cuenta los registros con valores nulos ? Antes de hacer update tomar nota de los salarios para luego del ejercicio volver al valor original.

update employees set salary=null

where employee\_id in (114,115)

commit

Sigue sumando ese empleado aunque esté nulo.

8.2)-Ejecutar el siguiente comando que asigna valor 0 al salario de dos empleados. Primero hay que deshabilitar la constraint EMP\_SALARY\_MIN.

ALTER TABLE HR.EMPLOYEES MODIFY CONSTRAINT EMP\_SALARY\_MIN DISABLE VALIDATE;

update employees set salary=0

where employee\_id in (114,115)

commit

Ejecutar nuevamente la consulta. Se modificó el resultado ? Antes de continuar volvamos los salarios a los valores iniciales:

update employees set salary=11000

where employee\_id =114;

update employees set salary=3100

where employee\_id =115;

commit;

9)-Escriba una consulta que despliegue el salario más alto, el salario más bajo, y la diferencia entre el más alto y más bajo salario por id de job. Etiquete la columna como DIFFERENCE.

SELECT MAX(salary),MIN(salary),MAX(salary)-MIN(salary) AS DIFFERENCE FROM Employees;

10)- Escriba una consulta que despliegue el Puesto , salario total mensual para cada puesto con una nómina total arriba de $13000. No incluír aquellos que los id de jobs no tengan la cadena de caracteres REP. Ordenar desde el mayor al menor salario.

desc jobs;

select job\_title, min\_salary + max\_salary as salary, job\_id from jobs

where job\_id <any (select AVG(min\_salary + max\_salary) >=13000) and job\_id != 'MK\_REP'

group by job\_id;

11)-Escriba una consulta que despliegue el nombre y la cantidad de veces que se repite ese nombre en la tabla de empleados.

SELECT first\_name , count(\*) FROM employees

GROUP BY employee\_id;

12)-Escribir una consulta que muestre el mayor salario promedio. Es decir del promedio de salarios por departamento obtener el mayor.

desc departments;

desc employees;

SELECT DEPARTMENT\_ID, SALARY FROM (

SELECT DEPARTMENT\_ID,MAX(SALARY) SALARY

FROM Employees

GROUP BY 1

) c1 JOIN employeeS USING(SALARY,DEPARTMENT\_ID)

WHERE SALARY=(SELECT MAX(SALARY) FROM EMPLOYEES);