A logo of a university

Description automatically generatedA blue circle with white text and red triangle

Description automatically generatedUNIVERSIDAD AUTÓNOMA DE CHIHUAHUA

Facultad de Ingeniería

Ingeniería en Ciencias de la Computación

**Bases de Datos Avanzadas**

**Practica 3**

*Trabajo de:* ADRIAN ALEJANDRO GONZÁLEZ DOMÍNGUEZ *[359834]*

*Asesor:* JORGE ALBERTO CHAPARRO TARANGO

*27 de febrero de 2024*

**1. Visualizar el número de empleados de cada departamento.** Utilizar GROUP BY para agrupar por departamento.

SELECT dept\_no, COUNT(\*) as employeesNumber FROM emple

GROUP BY dept\_no;

A screenshot of a black box

Description automatically generated

**2. Visualizar los departamentos con más de 5 empleados.** Utilizar GROUP BY para agrupar por departamento y HAVING para establecer la condición sobre los grupos.

SELECT depart.\*, COUNT(\*) as employeesNumber FROM emple

JOIN depart ON emple.dept\_no=depart.dept\_no

GROUP BY emple.dept\_no

HAVING COUNT(\*) >5;

A black and white photo of a name

Description automatically generated

**3. Hallar la media de los salarios de cada departamento** (utilizar la función avg y GROUP BY).

SELECT depart.\*, AVG(emple.salario) as mediaSalario FROM emple

JOIN depart ON emple.dept\_no=depart.dept\_no

GROUP BY emple.dept\_no;

A black screen with white text

Description automatically generated

**4. Visualizar el nombre de los empleados vendedores del departamento ʻVENTASʼ (**Nombre del departamento=ʼVENTASʼ, oﬁcio=ʼVENDEDORʼ**).**

SELECT apellido FROM emple

WHERE (SELECT dept\_no FROM depart WHERE dnombre="VENTAS")=emple.dept\_no

AND emple.oficio="VENDEDOR";

A black and white screen with white text

Description automatically generated

**5. Visualizar el número de vendedores del departamento ʻVENTASʼ** (utilizarla función COUNT sobre la consulta anterior).

SELECT COUNT(\*) as vendedoresDeVentas FROM emple

WHERE (SELECT dept\_no FROM depart WHERE dnombre="VENTAS")=emple.dept\_no

AND emple.oficio="VENDEDOR";

A black screen with white text

Description automatically generated

**6. Visualizar los oﬁcios de los empleados del departamento ʻVENTASʼ.**

SELECT apellido, oficio FROM emple

WHERE (SELECT dept\_no FROM depart WHERE dnombre="VENTAS")=emple.dept\_no;

A black screen with white text

Description automatically generated

**7. A partir de la tabla EMPLE, visualizar el número de empleados de cada departamento cuyo oﬁcio sea ʻEMPLEADO ʼ** (utilizar GROUP BY para agrupar por departamento. En la cláusula WHERE habrá que indicar que el oﬁcio es ʻEMPLEADOʼ).

SELECT depart.\*, COUNT(\*) as numeroDeEmpleados FROM emple

JOIN depart ON emple.dept\_no=depart.dept\_no

WHERE emple.oficio="EMPLEADO"

GROUP BY emple.dept\_no;

A black and white screen with white text

Description automatically generated

**8. Visualizar el departamento con más empleados.**

SELECT depart.\*, COUNT(\*) AS numeroEmpleados FROM emple

JOIN depart ON emple.dept\_no=depart.dept\_no

GROUP BY emple.dept\_no

HAVING COUNT(\*)=(

SELECT MAX(numeroEmpleados) AS mayorNumeroDeEmpleados

FROM (

SELECT dept\_no,COUNT(\*) AS numeroEmpleados FROM emple

GROUP BY emple.dept\_no

) AS cuentaDepartamentos

);

A black and white text

Description automatically generated

**9. Mostrar los departamentos cuya suma de salarios sea mayor que la media de salarios de todos los empleados.**

SELECT depart.\*, SUM(salario) AS sumaSalario FROM emple

JOIN depart ON emple.dept\_no=depart.dept\_no

GROUP BY emple.dept\_no

HAVING SUM(salario)>(

SELECT AVG(salario) AS salario FROM emple

);

A screenshot of a black screen

Description automatically generated

**10. Para cada oﬁcio obtener la suma de salarios.**

SELECT emple.oficio, SUM(salario) AS totalSalarioDelOficio FROM emple

GROUP BY emple.oficio;

A screenshot of a black screen

Description automatically generated

**11. Visualizar la suma de salarios de cada oﬁcio del departamento ʻVENTASʼ.**

SELECT emple.oficio, SUM(salario) AS totalSalarioDelOficio FROM emple

WHERE dept\_no=(

SELECT dept\_no FROM depart WHERE dnombre="VENTAS"

)

GROUP BY emple.oficio;

A screenshot of a computer

Description automatically generated

**12. Visualizar el número de departamento que tenga más empleados cuyo oﬁcio sea empleado.**

CREATE VIEW empleadosOficina AS

SELECT dept\_no,COUNT(\*) AS numeroEmpleados FROM emple

WHERE emple.oficio="EMPLEADO"

GROUP BY emple.dept\_no;

CREATE VIEW maxEmpleadosOficina AS

SELECT MAX(numeroEmpleados) AS mayorNumeroDeEmpleados

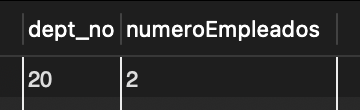
FROM empleadosOficina;

SELECT dept\_no, COUNT(\*) AS numeroEmpleados FROM emple

WHERE emple.oficio="EMPLEADO"

GROUP BY emple.dept\_no

HAVING COUNT(\*)=(SELECT \* FROM maxEmpleadosOficina);



**13. Mostrar el número de oﬁcios distintos de cada departamento.**

SELECT dept\_no, COUNT(DISTINCT(oficio)) AS numeroOficiosDistintos FROM emple

GROUP BY emple.dept\_no;

A screenshot of a black screen

Description automatically generated

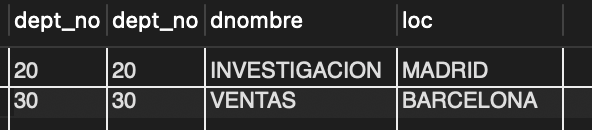
**14. Mostrar los departamentos que tengan más de dos personas trabajando en la misma profesión.**

SELECT DISTINCT(depart.dept\_no), depart.\* FROM emple

JOIN depart ON depart.dept\_no=emple.dept\_no

GROUP BY emple.oficio

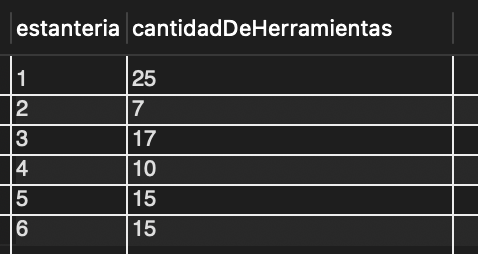
HAVING COUNT(\*)>=2;



**15. Dada la tabla HERRAMIENTAS, visualizar por cada estantería la suma de las unidades.**

SELECT estanteria, SUM(unidades) as cantidadDeHerramientas FROM herramientas

GROUP BY estanteria;



**16. Visualizar la estantería con más unidades de la tabla HERRAMIENTAS.**

CREATE VIEW unidadesPorEstanteria AS

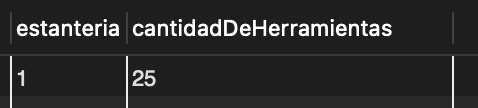
SELECT SUM(unidades) as cantidadDeHerramientas FROM herramientas

GROUP BY estanteria;

SELECT estanteria, SUM(unidades) as cantidadDeHerramientas FROM herramientas

GROUP BY estanteria

HAVING SUM(unidades)=(SELECT MAX(cantidadDeHerramientas) FROM unidadesPorEstanteria);



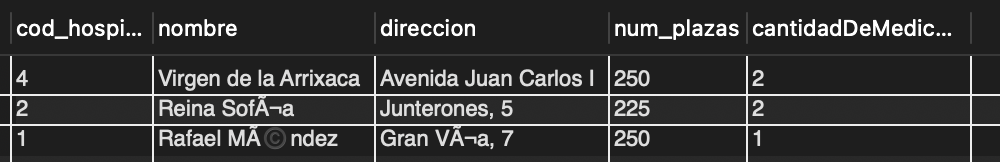
**17. Mostrar el número de médicos que pertenecen a cada hospital, ordenado por número descendente de hospital.**

SELECT hospitales.\*, COUNT(\*) AS cantidadDeMedicos FROM medicos

JOIN hospitales ON hospitales.cod\_hospital=medicos.cod\_hospital

GROUP BY medicos.cod\_hospital

ORDER BY medicos.cod\_hospital DESC;



**18. Realizar una consulta en la que se muestre por cada hospital el nombre de las especialidades que tiene.**

CREATE VIEW hospitalEspecialidad AS

SELECT hospitales.\*, medicos.especialidad AS especialidad FROM medicos

JOIN hospitales ON hospitales.cod\_hospital=medicos.cod\_hospital

GROUP BY medicos.cod\_hospital, medicos.especialidad;

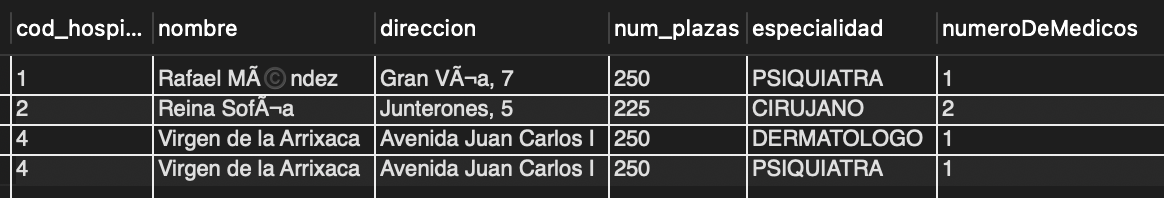
SELECT cod\_hospital, nombre, GROUP\_CONCAT(especialidad) as especialidades FROM hospitalEspecialidad

GROUP BY cod\_hospital; 

**19. Realizar una consulta en la que aparezca por cada hospital y en cada especialidad el número de médicos** (tendrás que partir de la consulta anterior y utilizar GROUP BY).

SELECT hospitales.\*, medicos.especialidad, COUNT(\*) as numeroDeMedicos FROM medicos

JOIN hospitales ON hospitales.cod\_hospital=medicos.cod\_hospital

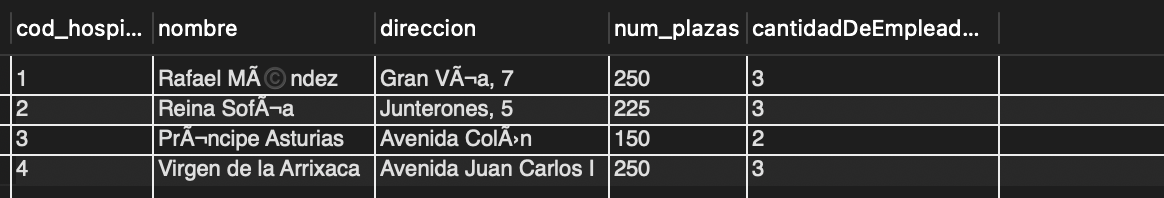
GROUP BY medicos.cod\_hospital, medicos.especialidad;

**20. Obtener por cada hospital el número de empleados.**

SELECT hospitales.\*, COUNT(\*) as cantidadDeEmpleados FROM personas

JOIN hospitales ON hospitales.cod\_hospital=personas.cod\_hospital

GROUP BY personas.cod\_hospital;



**21. Obtener por cada especialidad el número de trabajadores.**

SELECT especialidad, COUNT(\*) as numeroDeTrabajadores FROM medicos

GROUP BY especialidad;

A black and white screen with white text

Description automatically generated

**22. Visualizar la especialidad que tenga más médicos.**

SELECT especialidad, COUNT(\*) AS cantidadDeTrabajadores FROM medicos

GROUP BY especialidad

HAVING COUNT(\*)=(

SELECT MAX(cantidadDeTrabajadores) FROM (

SELECT COUNT(\*) as cantidadDeTrabajadores FROM medicos

GROUP BY especialidad

) AS cantidadDeTrabajadores

);

A black and white screen with white text

Description automatically generated

**23. ¿Cuál es el nombre del hospital que tiene mayor número de plazas?**

SELECT \* FROM hospitales

WHERE num\_plazas=(

SELECT MAX(num\_plazas) FROM hospitales

);

A black and white screen with white text

Description automatically generated

**24. Visualizar las diferentes estanterías de la tabla HERRAMIENTAS ordenados descendentemente por estantería.**

SELECT DISTINCT(estanteria) FROM herramientas

ORDER BY estanteria DESC;

A black and white rectangular object with white text

Description automatically generated

**25. Averiguar cuántas unidades tiene cada estantería.**

SELECT estanteria, SUM(unidades) as unidades FROM herramientas

GROUP BY estanteria;

A black and white rectangular sign with white text

Description automatically generated

**26. Visualizar las estanterías que tengan más de 15 unidades**

SELECT estanteria, SUM(unidades) as unidades FROM herramientas

GROUP BY estanteria

HAVING SUM(unidades)>15;

A black and white screen with white text

Description automatically generated

**27. ¿Cuál es la estantería que tiene más unidades?**

SELECT estanteria, SUM(unidades) as unidades FROM herramientas

GROUP BY estanteria

HAVING SUM(unidades)=(

SELECT MAX(unidades) FROM (

SELECT estanteria, SUM(unidades) as unidades FROM herramientas

GROUP BY estanteria

) AS unidades

);

A black and white text

Description automatically generated

**28. A partir de las tablas EMPLE y DEPART mostrar los datos del departamento que no tiene ningún empleado.**

DROP VIEW IF EXISTS empleadosEnDepartamento;

CREATE VIEW empleadosEnDepartamento AS

SELECT dept\_no, COUNT(\*) AS cantidadDeEmpleados FROM emple

GROUP BY dept\_no;

SELECT \* FROM depart

LEFT JOIN empleadosEnDepartamento ON depart.dept\_no=empleadosEnDepartamento.dept\_no

WHERE empleadosEnDepartamento.cantidadDeEmpleados IS NULL;A black and white screen with white text

Description automatically generated

**29. Mostrar el número de empleados de cada departamento. En la salida se debe mostrar también los departamentos que no tienen ningún empleado.**

DROP VIEW IF EXISTS empleadosEnDepartamento;

CREATE VIEW empleadosEnDepartamento AS

SELECT dept\_no, COUNT(\*) AS cantidadDeEmpleados FROM emple

GROUP BY dept\_no;

SELECT depart.\*, empleadosEnDepartamento.cantidadDeEmpleados FROM depart

LEFT JOIN empleadosEnDepartamento ON depart.dept\_no=empleadosEnDepartamento.dept\_no; A black and white screen with white text

Description automatically generated

**30. Obtener la suma de salarios de cada departamento, mostrando las columnas DEPT\_NO, SUMA DE SALARIOS y DNOMBRE. En el resultado también se deben mostrar los departamentos que no tienen asignados empleados.**

DROP VIEW IF EXISTS sumaSalariosDepartamento;

CREATE VIEW sumaSalariosDepartamento AS

SELECT dept\_no, SUM(salario) AS sumaSalarios FROM emple

GROUP BY dept\_no;

SELECT depart.dept\_no, depart.dnombre, sumaSalariosDepartamento.sumaSalarios FROM depart

LEFT JOIN sumaSalariosDepartamento ON depart.dept\_no=sumaSalariosDepartamento.dept\_no; A screenshot of a black and white screen

Description automatically generated

**31. Utilizar la función IFNULL en la consulta anterior para que en el caso de que un departamento no tenga empleados, aparezca como suma de salarios el valor 0.**

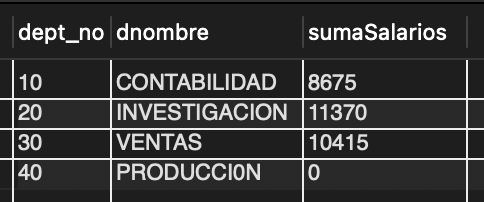
DROP VIEW IF EXISTS sumaSalariosDepartamento;

CREATE VIEW sumaSalariosDepartamento AS

SELECT dept\_no, SUM(salario) AS sumaSalarios FROM emple

GROUP BY dept\_no;

SELECT depart.dept\_no, depart.dnombre, IFNULL(sumaSalariosDepartamento.sumaSalarios,0) AS sumaSalarios FROM depart

LEFT JOIN sumaSalariosDepartamento ON depart.dept\_no=sumaSalariosDepartamento.dept\_no; 

**32. Obtener el número de médicos que pertenecen a cada hospital, mostrando las columnas COD\_HOSPITAL, NOMBRE y NÚMERO DE MÉDICOS. En el resultado deben aparecer también los datos de los hospitales que no tienen médicos.**

DROP VIEW IF EXISTS cantidadDeMedicosEnHospital;

CREATE VIEW cantidadDeMedicosEnHospital AS

SELECT cod\_hospital, COUNT(\*) AS cantidadDeMedicos FROM medicos

GROUP BY cod\_hospital;

SELECT hospitales.cod\_hospital, hospitales.nombre, cantidadDeMedicosEnHospital.cantidadDeMedicos FROM hospitales

LEFT JOIN cantidadDeMedicosEnHospital ON hospitales.cod\_hospital=cantidadDeMedicosEnHospital.cod\_hospital;

A black and white screen with white text

Description automatically generated