**Ejercicios SQL : Group By**

Tablas Empleados y Departamentos

1. Mostrar por cada oficio cuantos empleados tiene, cuál es el mejor salario, la suma total de los salarios y cuántos empleados tienen comisión.

**SELECT OFICIO, COUNT(\*), MAX(SALARIO), SUM(SALARIO), COUNT(COMISION)**

**FROM EMPLE**

**GROUP BY OFICIO;**

1. Mostrar el código y nombre del departamento de los departamentos que cuya suma de salarios supera los 9000€

**SELECT DEPART.DEPT\_NO, DNOMBRE, SUM(SALARIO)**

**FROM DEPART JOIN EMPLE ON EMPLE.DEPT\_NO = DEPART.DEPT\_NO**

**GROUP BY DNOMBRE, DEPART.DEPT\_NO**

**HAVING SUM(SALARIO)>9000;**

1. Mostrar la media del salario de los departamentos que tienen más de 3 empleados sin tener en cuenta ni a los presidentes ni a los directores.

**SELECT DEPT\_NO, AVG(SALARIO), COUNT(\*)**

**FROM EMPLE**

**WHERE OFICIO NOT IN ('PRESIDENTE', 'DIRECTOR')**

**GROUP BY DEPT\_NO**

**HAVING COUNT(\*) > 3;**

1. Mostrar por cada año cuantos empleados han entrado a la empresa

**SELECT TO\_CHAR(FECHA\_ALTA, 'YYYY'), COUNT(\*)**

**FROM EMPLE**

**GROUP BY TO\_CHAR(FECHA\_ALTA, 'YYYY');**

1. Mostrar el año en el que han entrado más empleados

**SELECT TO\_CHAR(FECHA\_ALTA, 'YYYY'), COUNT(\*)**

**FROM EMPLE**

**GROUP BY TO\_CHAR(FECHA\_ALTA, 'YYYY')**

**HAVING COUNT (\*) = (SELECT MAX(COUNT(\*))**

**FROM EMPLE**

**GROUP BY TO\_CHAR(FECHA\_ALTA, 'YYYY'));**

1. Mostrar el año en el que han entrado más empleados sin contar con los directores.

**SELECT TO\_CHAR(FECHA\_ALTA, 'YYYY'), COUNT(\*)**

**FROM EMPLE**

**WHERE OFICIO NOT LIKE 'DIRECTOR'**

**GROUP BY TO\_CHAR(FECHA\_ALTA, 'YYYY')**

**HAVING COUNT (\*) = (SELECT MAX(COUNT(\*))**

**FROM EMPLE**

**WHERE OFICIO NOT LIKE 'DIRECTOR'**

**GROUP BY TO\_CHAR(FECHA\_ALTA, 'YYYY'));**

Tablas Alumnos, Asignaturas, Notas

1. Mostrar la mejor y peor nota en cada una de las asignaturas junto al nombre de la asignatura.

**SELECT NOMBRE, MAX(NOTA), MIN(NOTA)**

**FROM NOTAS JOIN ASIGNATURAS ON ASIGNATURAS.COD = NOTAS.COD**

**GROUP BY NOMBRE;**

1. Mostrar las asignaturas que tienen más de un alumno suspenso.

**SELECT COUNT(\*), NOMBRE**

**FROM NOTAS JOIN ASIGNATURAS ON ASIGNATURAS.COD = NOTAS.COD**

**WHERE NOTA < 5**

**GROUP BY NOMBRE**

**HAVING COUNT(\*)>1;**

1. Mostrar para cada alumno su nombre junto a la media de sus notas pero solo para alumnos que esta media sea al menos de 6

**SELECT APENOM, AVG(NOTA)**

**FROM ALUMNOS JOIN NOTAS ON NOTAS.DNI = ALUMNOS.DNI**

**GROUP BY APENOM**

**HAVING AVG(NOTA)>=6;**

1. Mostrar el código y el nombre de la asignatura en la que se han obtenido más suspensos.

**SELECT ASIGNATURAS.COD, NOMBRE, COUNT (\*)**

**FROM ASIGNATURAS JOIN NOTAS ON NOTAS.COD = ASIGNATURAS.COD**

**WHERE NOTA < 5**

**GROUP BY NOMBRE, ASIGNATURAS.COD**

**HAVING COUNT(\*) = (SELECT MAX(COUNT(\*))**

**FROM ASIGNATURAS JOIN NOTAS ON NOTAS.COD = ASIGNATURAS.COD**

**WHERE NOTA < 5**

**GROUP BY NOMBRE);**

Tablas Parque de Atracciones

1. Mostrar por cada zona cuantas averías arregladas ha tenido.

**SELECT NOM\_ZONA, COUNT(\*)**

**FROM AVERIAS\_PARQUE JOIN ATRACCIONES ON ATRACCIONES.COD\_ATRACCION = AVERIAS\_PARQUE.COD\_ATRACCION**

**WHERE FECHA\_ARREGLO IS NOT NULL**

**GROUP BY NOM\_ZONA;**

1. Mostrar la zona que más averías arregladas tiene.

**SELECT COUNT(\*), NOM\_ZONA**

**FROM AVERIAS\_PARQUE JOIN ATRACCIONES ON ATRACCIONES.COD\_ATRACCION = AVERIAS\_PARQUE.COD\_ATRACCION**

**WHERE FECHA\_ARREGLO IS NOT NULL**

**GROUP BY NOM\_ZONA**

**HAVING COUNT(\*) = (SELECT MAX(COUNT(\*))**

**FROM AVERIAS\_PARQUE JOIN ATRACCIONES ON ATRACCIONES.COD\_ATRACCION = AVERIAS\_PARQUE.COD\_ATRACCION**

**WHERE FECHA\_ARREGLO IS NOT NULL**

**GROUP BY NOM\_ZONA);**

1. Mostrar por cada día de la semana (en letra) cuantas averías se han producido.

**SELECT TO\_CHAR(FECHA\_FALLA, 'DAY'),COUNT(\*)**

**FROM AVERIAS\_PARQUE**

**GROUP BY TO\_CHAR(FECHA\_FALLA, 'DAY');**

1. Mostrar el día de la semana en el que se producen más averías.

**SELECT TO\_CHAR(FECHA\_FALLA, 'DAY'),COUNT(\*)**

**FROM AVERIAS\_PARQUE**

**GROUP BY TO\_CHAR(FECHA\_FALLA, 'DAY')**

**HAVING COUNT(\*) = (SELECT MAX(COUNT(\*))**

**FROM AVERIAS\_PARQUE**

**GROUP BY TO\_CHAR(FECHA\_FALLA, 'DAY'));**

1. Mostrar por cada empleado su nombre y la fecha de la última avería que ha terminado de arreglar.

**SELECT NOM\_EMPLEADO, MAX(FECHA\_ARREGLO)**

**FROM EMPLE\_PARQUE JOIN AVERIAS\_PARQUE ON AVERIAS\_PARQUE.DNI\_EMPLE = EMPLE\_PARQUE.DNI\_EMPLE**

**WHERE FECHA\_ARREGLO IS NOT NULL**

**GROUP BY NOM\_EMPLEADO;**