* Índice:

**No se encontraron elementos de tabla de contenido.**



## Cuándo usar = / IN, <> / NOT IN , > / > ALL( ) / > ANY( )

|  |  |
| --- | --- |
| **= (1 valor)** | **IN** (0,1,2,... valores) |
| **<> (1valor)** | **NOT IN** (0,1,2,... valores) |
| **> (1 valor)**  **<**  **>=**  **<=** | **> ALL** (0,1,2,... valores) *mayor que todos los valores del listado*  **> SOME** (0,1,2,... valores) *mayor que algún valor del listado*  **> ANY** (0,1,2,... valores) *mayor que algún valor del listado*  ***<, >=, >=*** *se utilizan de la misma manera que* ***>*** |

## Consulta con subconsulta

Resolver los siguientes ejercicios utilizando Consulta con subconsulta:

1. queremos saber la lista de empleados cuyo salario supere el salario medio

* Esta sería la manera de hacerlo ejecutando dos consultas, una tras otra a mano:

**SELECT AVG (salario)**

**FROM empleados;**

**#La consulta anterior informa de que el salario medio es de 249.000**

**#Ahora utilizamos ese valor para la consulta final:**

**SELECT \***

**FROM empleados**

**WHERE salario > 249.000;**

* Esta es la manera de juntarlo todo en una consulta con subconsulta:

**SELECT \***

**FROM empleados**

**WHERE salario >**

**(**

**SELECT AVG (salario)**

**FROM empleados**

**)**

**;**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | **EMPLEADOS** | | | | | | | | | --- | --- | --- | --- | --- | --- | --- | --- | | **EMP\_NO** | **APELLIDO** | **OFICIO** | **DIRECTOR** | **FECHA\_ALTA** | **SALARIO** | **COMISION** | **DEP\_NO** | | 7876 | GIL | ANALISTA | 7782 | 06/05/1982 | 335000 |  | 20 | | 7782 | MARTINEZ | DIRECTOR | 7839 | 09/06/1981 | 245000 |  | 10 | | 7698 | GARRIDO | DIRECTOR | 7839 | 01/05/1981 | 385000 |  | 30 | | 7900 | JIMENEZ | EMPLEADO | 7782 | 24/03/1983 | 140000 |  | 20 | | 7521 | LOPEZ | EMPLEADO | 7782 | 08/05/1981 | 135000 |  | 10 | | 7839 | REY | PRESIDENTE |  | 17/11/1981 | 600000 |  | 10 | | 8998 | CORTES | VENDEDOR | 7698 | 20/02/1999 | 180000 |  | 30 | | 7844 | CALVO | VENDEDOR | 7698 | 08/09/1981 | 180000 | 0 | 30 | | 7654 | MARTIN | VENDEDOR | 7698 | 28/09/1981 | 150000 | 160000 | 30 | | 7499 | ALONSO | VENDEDOR | 7698 | 20/02/1981 | 140000 | 40000 | 30 | | | **EMPLEADOS** | | | | | | | | | --- | --- | --- | --- | --- | --- | --- | --- | | **EMP\_NO** | **APELLIDO** | **OFICIO** | **DIRECTOR** | **FECHA\_ALTA** | **SALARIO** | **COMISION** | **DEP\_NO** | | 7876 | GIL | ANALISTA | 7782 | 06/05/1982 | 335000 |  | 20 | | 7782 | MARTINEZ | DIRECTOR | 7839 | 09/06/1981 | 245000 |  | 10 | | 7698 | GARRIDO | DIRECTOR | 7839 | 01/05/1981 | 385000 |  | 30 | | 7900 | JIMENEZ | EMPLEADO | 7782 | 24/03/1983 | 140000 |  | 20 | | 7521 | LOPEZ | EMPLEADO | 7782 | 08/05/1981 | 135000 |  | 10 | | 7839 | REY | PRESIDENTE |  | 17/11/1981 | 600000 |  | 10 | | 8998 | CORTES | VENDEDOR | 7698 | 20/02/1999 | 180000 |  | 30 | | 7844 | CALVO | VENDEDOR | 7698 | 08/09/1981 | 180000 | 0 | 30 | | 7654 | MARTIN | VENDEDOR | 7698 | 28/09/1981 | 150000 | 160000 | 30 | | 7499 | ALONSO | VENDEDOR | 7698 | 20/02/1981 | 140000 | 40000 | 30 | |

1. Obtener todos los empleados que tienen el mismo oficio que el empleado con apellido1 ‘Alonso’ .

**SELECT \***

**FROM empleados**

**WHERE oficio =**

**(**

**SELECT oficio**

**FROM empleados**

**WHERE apellido1 = 'Alonso'**

**)**

**;**

1. Obtener información de los empleados que ganan más que cualquier empleado del departamento 30 (con esto quiero decir que ningún empleado del departamento 30 gana más que tú). Hacerla de 2 formas distintas: usando MAX y usando > ALL.

**SELECT \***

**FROM empleados**

**WHERE salario > ALL**

**(**

**SELECT IFNULL(salario,0)**

**FROM empleados**

**WHERE dep\_no=30**

**)**

**;**

**#Hemos usado el IFNULL, porque sin él, en caso de que un salario del dep 30 fuese NULL no funcionaría la consulta**

#Otra forma:

**SELECT \***

**FROM empleados**

**WHERE salario >**

**(**

**SELECT MAX(salario)**

**FROM empleados**

**WHERE dep\_no=30**

**)**

**;**

1. Visualizar el número de empleados con oficio VENDEDOR del departamento de VENTAS, resolverlo de dos maneras:

* utilizando subconsulta, sin utilizar JOIN.
* utilizando JOIN, sin usar subconsulta.

**SELECT COUNT(\*)**

**FROM empleados**

**WHERE oficio = 'VENDEDOR'**

**AND**

**dep\_no =**

**(**

**SELECT dep\_no**

**FROM departamentos**

**WHERE dnombre = 'ventas'**

**)**

**;**

**SELECT COUNT(\*)**

**FROM departamentos INNER JOIN empleados ON departamentos.dep\_no = empleados.dep\_no**

**WHERE dnombre = 'ventas' AND oficio = 'vendedor';**

1. Visualizar la suma de los salarios para cada oficio de los empleados del departamento de VENTAS. Realizarlo de 2 maneras distintas:

* Utilizando subconsulta para obtener el dep\_no del departamento, sin usar JOIN.
* Utilizando JOIN, sin usar subconsultas.

**SELECT oficio, SUM(salario) AS 'Suma de Salarios'**

**FROM empleados**

**WHERE dep\_no =**

**(SELECT dep\_no**

**FROM departamentos**

**WHERE dnombre = ‘ventas’**

**)**

**GROUP BY oficio**

**;**

**SELECT oficio, SUM(salario)**

**FROM departamentos INNER JOIN empleados ON departamentos.dep\_no=empleados.dep\_no**

**WHERE dnombre = 'ventas'**

**GROUP BY oficio;**

1. Listar, en orden alfabético, aquellos empleados que no trabajen ni en Madrid ni en Barcelona (es decir, su departamento no se encuentra en esas ciudades). Hacerlo de 2 maneras

* con subconsulta.
* con JOIN.

**SELECT \***

**FROM empleados**

**WHERE dep\_no NOT IN**

**(**

**SELECT dep\_no**

**FROM departamentos**

**WHERE localidad IN ('Madrid', 'Barcelona')**

**)**

**ORDER BY APELLIDO1, APELLIDO2, NOMBRE**

**;**

**#equivalente**

**SELECT \***

**FROM empleados**

**WHERE dep\_no IN**

**(**

**SELECT dep\_no**

**FROM departamentos**

**WHERE localidad NOT IN ('Madrid', 'Barcelona')**

**)**

**ORDER BY APELLIDO1, APELLIDO2, NOMBRE**

**;**

**SELECT \***

**FROM empleados**

**WHERE dep\_no <> ALL**

**(**

**SELECT dep\_no**

**FROM departamentos**

**WHERE localidad IN ('Madrid', 'Barcelona')**

**)**

**ORDER BY APELLIDO1, APELLIDO2, NOMBRE**

**;**

#mediante subconsulta

**SELECT \***

**FROM empleados NATURAL JOIN departamentos**

**WHERE localidad NOT IN ('Madrid', 'Barcelona')**

**ORDER BY APELLIDO1, APELLIDO2, NOMBRE**

**;**

1. Listar los nombres de los departamentos que tengan algún empleado con fecha de alta anterior a 1982. Hacerlo de 2 maneras:

* con subconsulta.
* con JOIN.

**SELECT dnombre**

**FROM departamentos**

**WHERE dep\_no IN**

**(**

**SELECT dep\_no**

**FROM empleados**

**WHERE fecha\_alta <'1982/01/01'**

**)**

**;**

Otra forma:

#Con JOIN

**SELECT DISTINCT dnombre**

**FROM empleados NATURAL JOIN departamentos**

**WHERE fecha\_alta <'1982/01/01';**

1. Obtener el número de departamento y su nombre, siempre que haya más de 2 empleados trabajando en ellos.

**#resolución sencilla:**

**SELECT dep\_no, dnombre, COUNT(\*)**

**FROM empleados NATURAL JOIN departamentos**

**GROUP BY dep\_no**

**HAVING COUNT(\*) > 2;**

**#resolución alternativa con subconsultas, mucho más compleja.**

**SELECT dep\_no, dnombre**

**FROM departamentos**

**WHERE dep\_no IN**

**(**

**SELECT dep\_no**

**FROM empleados**

**GROUP BY dep\_no**

**HAVING COUNT(\*) > 2**

**)**

**;**

1. Listar las localidades donde existan departamentos con empleados cuya comisión supere el 10% del salario. Utilizando subconsulta con **IN**

**SELECT localidad**

**FROM departamentos**

**WHERE dep\_no IN**

**(**

**SELECT dep\_no**

**FROM empleados**

**WHERE IFNULL(COMISION,0) > 0.10 \* IFNULL(SALARIO,0)**

**)**

**;**

**SELECT localidad**

**FROM departamentos**

**WHERE dep\_no = ANY**

**(**

**SELECT dep\_no**

**FROM empleados**

**WHERE IFNULL(COMISION,0) > 0.10 \* IFNULL(SALARIO,0)**

**)**

**;**

1. Deseamos comprobar si todos los empleados tienen asignado un código de departamento existente en la tabla de departamentos. Para ello vamos a mostrar a los empleados que están en alguna de estas situaciones:

* empleados que tienen NULL en su dep\_no
* empleados que tienen asignado un dep\_no erróneo (ese dep\_no no existe en la tabla departamentos)

Hacer el ejercicio utilizando subconsulta

#solución con subconsulta

**SELECT apellido1, apellido2, nombre**

**FROM empleados**

**WHERE dep\_no IS NULL OR dep\_no NOT IN**

**(**

**SELECT dep\_no**

**FROM departamentos**

**)**

**;**

#Otra solución con subconsulta

**SELECT \* FROM empleados WHERE emp\_no NOT IN**

**(SELECT emp\_no FROM empleados WHERE dep\_no IN**

**(SELECT dep\_no FROM departamentos));**

**# ESTA SOLUCIÓN ES INCORRECTA PORQUE NO MUESTRA A EMPLEADOS SIN DEPARTATAMENTO.**

**SELECT apellido1, apellido2, nombre**

**FROM empleados**

**WHERE dep\_no NOT IN**

**(**

**SELECT dep\_no**

**FROM departamentos**

**)**

**;**

**# de la manera que se muestra a continuación descubriríamos a los emplados que no tienen asignado ningún departamento, pero no seríamos capaces de descubrir a aquellos empleados que tienen asignado un departamento inexistente.**

**SELECT \***

**FROM empleados**

**WHERE dep\_no IS NULL ;**

1. Eliminada por errónea

**SELECT apellido**

**FROM empleados**

**WHERE (dep\_no = ANY (**

**SELECT dep\_no**

**FROM departamentos**

**));**

**SELECT apellido**

**FROM empleados**

**WHERE NOT (dep\_no IN (**

**SELECT dep\_no**

**FROM departamentos**

**));**

1. Seleccionar aquellos departamentos en los que al menos exista un empleado con comisión. Utilizando **ANY**.

**SELECT dnombre**

**FROM departamentos**

**WHERE dep\_no = ANY (**

**SELECT dep\_no**

**FROM empleados**

**WHERE comision > 0**

**);**

1. Seleccionar aquellos departamentos en los que al menos exista un empleado con comisión. Utilizando **IN**.

**SELECT dnombre**

**FROM departamentos**

**WHERE dep\_no IN (**

**SELECT dep\_no**

**FROM empleados**

**WHERE comision > 0**

**);**

1. Listar aquellos departamentos en los que todos sus empleados carezcan de información sobre su comisión. Utilizando **ALL**

**#ESTA SOLUCION PARECE CORRECTA PERO NO ES CORRECTA (EL DEPARTAMENTO 20 DEBERÍA MOSTRATRSE Y NO LO HACE).**

**SELECT dnombre**

**FROM departamentos**

**WHERE dep\_no NOT IN (**

**SELECT dep\_no**

**FROM empleados**

**WHERE comision IS NOT NULL**

**);**

**#ESTA SOLUCION SÍ ES CORRECTA.**

**SELECT dnombre**

**FROM departamentos**

**WHERE dep\_no <> ALL (**

**SELECT dep\_no**

**FROM empleados**

**WHERE comision IS NOT NULL AND dep\_no IS NOT NULL**

**);**

1. Listar aquellos departamentos en los que todos sus empleados carezcan de información sobre su comisión. Utilizando **NOT IN**

**#ESTA SOLUCION PARECE CORRECTA PERO NO ES CORRECTA (EL DEPARTAMENTO 20 DEBERÍA MOSTRATRSE Y NO LO HACE).**

**SELECT dnombre**

**FROM departamentos**

**WHERE dep\_no NOT IN (**

**SELECT dep\_no**

**FROM empleados**

**WHERE comision IS NOT NULL**

**);**

**#ESTA SOLUCION SÍ ES CORRECTA.**

**SELECT dnombre**

**FROM departamentos**

**WHERE dep\_no NOT IN (**

**SELECT dep\_no**

**FROM empleados**

**WHERE comision IS NOT NULL AND dep\_no IS NOT NULL**

**);**

1. Visualizar todos os datos do producto con menor stock\_disponible
2. Visualizar todos os datos do empregado que maior salario ten
3. Visualizar el departamento con más empleados.

**SELECT dnombre**

**FROM departamentos**

**WHERE dep\_no IN**

**(**

**SELECT dep\_no**

**FROM empleados**

**GROUP BY dep\_no**

**HAVING COUNT (\*) >= ALL**

**(**

**SELECT COUNT(\*)**

**FROM empleados**

**GROUP BY dep\_no**

**)**

**);**

#otra forma

SELECT COUNT(\*), dnombre

FROM empleados NATURAL JOIN departamentos

GROUP BY departamentos.dep\_no

HAVING COUNT(\*) >= (

SELECT COUNT(\*)

FROM empleados

GROUP BY dep\_no

ORDER BY 1 DESC

LIMIT 1

);

1. Visualizar los departamentos en los que el salario medio de sus empleados sea mayor o igual que la media de todos los salarios de la empresa.

**SELECT dnombre**

**FROM departamentos**

**WHERE dep\_no IN**

**(**

**SELECT dep\_no**

**FROM empleados**

**GROUP BY dep\_no**

**HAVING AVG (salario) >=**

**(**

**SELECT AVG (salario)**

**FROM empleados**

**)**

**);**

1. Visualizar el departamento con más presupuesto asignado para pagar el salario y la comisión de sus empleados.

**SELECT dnombre**

**FROM departamentos**

**WHERE dep\_no IN**

**(**

**SELECT dep\_no**

**FROM empleados**

**GROUP BY dep\_no**

**HAVING SUM(IFNULL(salario,0))+ SUM(IFNULL(commision,0)) >= ALL**

**(**

**SELECT SUM(IFNULL(salario,0))+ SUM(IFNULL(commision,0))**

**FROM empleados**

**GROUP BY dep\_no**

**)**

**);**

1. Visualizar el departamento con más personal del oficio ‘*empleado*’.

**SELECT dnombre**

**FROM departamentos**

**WHERE dep\_no IN**

**(**

**SELECT dep\_no**

**FROM empleados**

**WHERE oficio = 'empleado'**

**GROUP BY dep\_no**

**HAVING COUNT(\*) >= ALL**

**(**

**SELECT COUNT (\*)**

**FROM empleados**

**WHERE oficio = 'empleado'**

**GROUP BY dep\_no**

**)**

**);**

1. Visualizar el número y el nombre del departamento con más personal de oficio “*empleado*”, usando las tablas de empleados y de departamentos.

**SELECT dnombre**

**FROM empleados NATURAL JOIN departamentos**

**WHERE oficio = 'empleado'**

**GROUP BY dep\_no**

**HAVING COUNT(\*) >= ALL**

**(**

**SELECT COUNT (\*)**

**FROM empleados**

**WHERE oficio = "empleado"**

**GROUP BY dep\_no**

**);**