

EJERCICIO BASE DE DATOS RELACION NUMERO

1. empleados Consultas Basicas 3

1. Lista todas las columnas de la tabla empleado.

```
SELECT e.codigo, e.nif, e.nombre, e.apellido1, e.apellido2, e.codigo_departamento
FROM empleado e;
```

-- RESULTADO

	codigo	nif	nombre	apellido1	apellido2	codigo_departamento
1	32481596F		Aarón	Rivero	Gómez	1
2	Y5575632D		Adela	Salas	Díaz	2
3	R6970642B		Adolfo	Rubio	Flores	3
4	77705545E		Adrián	Suárez	NULL	4
5	17087203C		Marcos	Loyola	Méndez	5
6	38382980M		María	Santana	Moreno	1
7	80576669X		Pilar	Ruiz	NULL	2
8	71651431Z		Pepe	Ruiz	Santana	3
9	56399183D		Juan	Gómez	López	2
10	46384486H		Diego	Flores	Salas	5
11	67389283A		Marta	Herrera	Gil	1
12	41234836R		Irene	Salas	Flores	NULL
13	82635162B		Juan Antonio	Sáez	Guerrero	NULL

2. empleados Consultas Basicas 5

5. Lista el código de los departamentos de los empleados que aparecen en la tabla empleado.

```
SELECT e.codigo_departamento
FROM empleado e;
```

-- RESULTADO

codigo_departamento
NULL
NULL
1
1
1
2
2
2
3
3
4
5
5

3. empleados Consultas Basicas 11

11. Lista el nombre de cada departamento y el valor del presupuesto actual del que dispone. Para calcular este dato tendrá que restar al valor del presupuesto inicial (columna presupuesto) los gastos que se han generado (columna gastos). Tenga en cuenta que en algunos casos pueden existir valores negativos. Utilice un alias apropiado para la nueva columna columna que está calculando.

```
SELECT d.nombre, (d.presupuesto - d.gastos) AS presupuesto_actual
FROM departamento d;
```

```
-- RESULTADO
nombre presupuesto_actual
Desarrollo 114000
Sistemas 129000
Recursos Humanos 255000
Contabilidad 107000
I+D -5000
Proyectos 0
Publicidad -1000
```

4. empleados Consultas Basicas 13

13. Lista el nombre de todos los departamentos ordenados de forma ascendente.

```
SELECT d.nombre
FROM departamento d
ORDER BY d.nombre ASC;
```

```
-- RESULTADO
nombre
Contabilidad
Desarrollo
I+D
Proyectos
Publicidad
Recursos Humanos
Sistemas
```

5. empleados Consultas Basicas 19

19. Devuelve una lista con el nombre y el gasto, de los 2 departamentos que tienen menor gasto.

```
SELECT d.nombre, d.gastos
FROM departamento d
ORDER BY d.gastos ASC
LIMIT 2;
```

```
-- SQL Server
SELECT TOP 2 d.nombre, d.gastos
FROM departamento d
ORDER BY d.presupuesto ASC;
```

```
-- RESULTADO
nombre gastos
Proyectos    0
Publicidad   1000
```

6. empleados Consultas Basicas 21

21. Devuelve una lista con el nombre de los departamentos y el presupuesto, de aquellos que tienen un presupuesto mayor o igual a 150000 euros.

```
SELECT d.nombre, d.presupuesto
FROM departamento d
WHERE d.presupuesto >= 150000;
```

```
-- RESULTADO
nombre presupuesto
Sistemas    150000
Recursos Humanos  280000
I+D 375000
```

7. empleados Consultas Basicas 27

27. Devuelve una lista con el nombre de los departamentos, gastos y presupuesto, de aquellos departamentos donde los gastos sean mayores que el presupuesto del que disponen.

```
SELECT d.nombre, d.gastos, d.presupuesto
FROM departamento d
WHERE d.gastos > d.presupuesto;
```

```
-- RESULTADO
nombre gastos presupuesto
I+D 380000 375000
Publicidad 1000 0
```

8. empleados Consultas Basicas 29

29. Devuelve una lista con el nombre de los departamentos, gastos y presupuesto, de aquellos departamentos donde los gastos sean iguales al presupuesto del que disponen.

```
SELECT d.nombre, d.gastos, d.presupuesto
FROM departamento d
WHERE d.gastos = d.presupuesto;

-- RESULTADO
Proyectos    0    0
```

9. empleados Consultas Basicas 35

35. Lista los nombres, apellidos y nif de los empleados que trabajan en el departamento 3.

```
SELECT e.nombre, e.apellido1, e.apellido2, e.nif
FROM empleado e
WHERE e.codigo_departamento = 3;

-- RESULTADO
nombre apellido1 apellido2 nif
Adolfo Rubio Flores R6970642B
Pepe Ruiz Santana 71651431Z
```

10. empleados Composición interna 1

1. Devuelve un listado con los empleados y los datos de los departamentos donde trabaja cada uno.

```
-- SQL1
SELECT
    e.codigo AS codigo_empleado,
    e.nif,
    e.nombre AS nombre_empleado,
    e.apellido1,
    e.apellido2,
    e.codigo_departamento,
    d.codigo AS codigo_del_departamento,
    d.nombre AS nombre_departamento,
    d.presupuesto,
    d.gastos
FROM empleado e, departamento d
    WHERE e.codigo_departamento = d.codigo;

-- SQL2
SELECT
    e.codigo AS codigo_empleado,
    e.nif,
    e.nombre AS nombre_empleado,
    e.apellido1,
    e.apellido2,
    e.codigo_departamento,
```

```

    d.codigo AS codigo_del_departamento,
    d.nombre AS nombre_departamento,
    d.presupuesto,
    d.gastos
FROM empleado e INNER JOIN departamento d
    ON e.codigo_departamento = d.codigo;

-- RESULTADO
codigo_empleado nif nombre_empleado apellido1 apellido2 codigo_departamento
codigo_del_departamento nombre_departamento presupuesto gastos
1 32481596F Aarón Rivero Gómez 1 1 Desarrollo 120000 6000
6 38382980M María Santana Moreno 1 1 Desarrollo 120000 6000
11 67389283A Marta Herrera Gil 1 1 Desarrollo 120000 6000
2 Y5575632D Adela Salas Díaz 2 2 Sistemas 150000 21000
7 80576669X Pilar Ruiz NULL 2 2 Sistemas 150000 21000
9 56399183D Juan Gómez López 2 2 Sistemas 150000 21000
3 R6970642B Adolfo Rubio Flores 3 3 Recursos Humanos 280000 25000
8 71651431Z Pepe Ruiz Santana 3 3 Recursos Humanos 280000 25000
4 77705545E Adrián Suárez NULL 4 4 Contabilidad 110000 3000
5 17087203C Marcos Loyola Méndez 5 5 I+D 375000 380000
10 46384486H Diego Flores Salas 5 5 I+D 375000 380000

```

11. empleados Composición interna 7

7.Devuelve un listado con los datos de los empleados que trabajan en el departamento de I+D. Ordena el resultado alfabéticamente.

```

-- SQL1
SELECT e.codigo, e.nif, e.nombre, e.apellido1, e.apellido2, e.codigo_departamento
FROM empleado e, departamento d
    WHERE e.codigo_departamento = d.codigo AND d.nombre = 'I+D'
ORDER BY e.nombre ASC;

-- SQL2
SELECT e.codigo, e.nif, e.nombre, e.apellido1, e.apellido2, e.codigo_departamento
FROM empleado e INNER JOIN departamento d
    ON e.codigo_departamento = d.codigo
WHERE d.nombre = 'I+D'
ORDER BY e.nombre ASC;

-- RESULTADO
codigo nif nombre apellido1 apellido2 codigo_departamento
10 46384486H Diego Flores Salas 5
5 17087203C Marcos Loyola Méndez 5

```

12. empleados Composición interna 9

9. Devuelve una lista con el nombre de los empleados que tienen los departamentos que no tienen un presupuesto entre 100000 y 200000 euros.

```
-- SQL1
SELECT e.nombre AS nombre_empleado
FROM empleado e, departamento d
    WHERE e.codigo_departamento = d.codigo AND d.presupuesto NOT BETWEEN 100000
AND 200000;

-- SQL2
SELECT e.nombre AS nombre_empleado
FROM empleado e INNER JOIN departamento d
    ON e.codigo_departamento = d.codigo
WHERE NOT d.presupuesto BETWEEN 100000 AND 200000;

-- RESULTADO
nombre_empleado
Adolfo
Pepe
Marcos
Diego
```

13. empleados Composición externa 1

1. Devuelve un listado con todos los empleados junto con los datos de los departamentos donde trabajan. Este listado también debe incluir los empleados que no tienen ningún departamento asociado.

```
SELECT
    e.codigo AS codigo_empleado,
    e.nif,
    e.nombre AS nombre_empleado,
    e.apellido1,
    e.apellido2,
    d.codigo AS codigo_departamento,
    d.nombre AS nombre_departamento,
    d.presupuesto,
    d.gastos
FROM empleado e LEFT JOIN departamento d
    ON e.codigo_departamento = d.codigo;

-- RESULTADO
codigo_empleado nif nombre_empleado apellido1 apellido2 codigo_departamento
nombre_departamento presupuesto gastos
1 32481596F Aarón Rivero Gómez 1 Desarrollo 120000 6000
2 Y5575632D Adela Salas Díaz 2 Sistemas 150000 21000
3 R6970642B Adolfo Rubio Flores 3 Recursos Humanos 280000 25000
4 77705545E Adrián Suárez NULL 4 Contabilidad 110000 3000
5 17087203C Marcos Loyola Méndez 5 I+D 375000 380000
6 38382980M María Santana Moreno 1 Desarrollo 120000 6000
7 80576669X Pilar Ruiz NULL 2 Sistemas 150000 21000
8 71651431Z Pepe Ruiz Santana 3 Recursos Humanos 280000 25000
9 56399183D Juan Gómez López 2 Sistemas 150000 21000
10 46384486H Diego Flores Salas 5 I+D 375000 380000
```

11	67389283A	Marta	Herrera	Gil	1	Desarrollo	120000	6000	
12	41234836R	Irene	Salas	Flores		NULL	NULL	NULL	NULL
13	82635162B	Juan	Antonio	Sáez		Guerrero	NULL	NULL	NULL

14. empleados Composición externa 5

5. Devuelve un listado con los empleados que no tienen ningún departamento asociado y los departamentos que no tienen ningún empleado asociado. Ordene el listado alfabéticamente por el nombre del departamento.

```
(SELECT
    e.codigo AS codigo_empleado,
    e.nif,
    e.nombre AS nombre_empleado,
    e.apellido1, e.apellido2,
    d.codigo AS codigo_departamento,
    d.nombre AS nombre_departamento,
    d.presupuesto,
    d.gastos
FROM empleado e LEFT JOIN departamento d
    ON e.codigo_departamento = d.codigo
WHERE d.codigo IS NULL)

UNION

(SELECT
    e.codigo AS codigo_empleado,
    e.nif,
    e.nombre AS nombre_empleado,
    e.apellido1, e.apellido2,
    d.codigo AS codigo_departamento,
    d.nombre AS nombre_departamento,
    d.presupuesto,
    d.gastos
FROM empleado e RIGHT JOIN departamento d
    ON e.codigo_departamento = d.codigo
WHERE e.codigo_departamento IS NULL
ORDER BY d.nombre ASC);

-- OTRA FORMA:
((SELECT
    e.codigo AS codigo_empleado,
    e.nif,
    e.nombre AS nombre_empleado,
    e.apellido1, e.apellido2,
    d.codigo AS codigo_departamento,
    d.nombre AS nombre_departamento,
    d.presupuesto,
    d.gastos
FROM empleado e LEFT JOIN departamento d
    ON e.codigo_departamento = d.codigo)
UNION
```

```
(SELECT
  e.codigo AS codigo_empleado,
  e.nif,
  e.nombre AS nombre_empleado,
  e.apellido1, e.apellido2,
  d.codigo AS codigo_departamento,
  d.nombre AS nombre_departamento,
  d.presupuesto,
  d.gastos
FROM empleado e RIGHT JOIN departamento d
  ON e.codigo_departamento = d.codigo
ORDER BY d.nombre ASC))

EXCEPT

(SELECT
  e.codigo AS codigo_empleado,
  e.nif,
  e.nombre AS nombre_empleado,
  e.apellido1, e.apellido2,
  d.codigo AS codigo_departamento,
  d.nombre AS nombre_departamento,
  d.presupuesto,
  d.gastos
FROM empleado e INNER JOIN departamento d
  ON e.codigo_departamento = d.codigo);

-- RESULTADO
codigo_empleado nif nombre_empleado apellido1 apellido2 codigo_departamento
nombre_departamento presupuesto gastos
12 41234836R Irene Salas Flores NULL NULL NULL NULL
13 82635162B Juan Antonio Sáez Guerrero NULL NULL NULL NULL
NULL NULL NULL NULL NULL 6 Proyectos 0 0
NULL NULL NULL NULL NULL 7 Publicidad 0 1000
```

15. empleados Resumen 2

2. Calcula la media del presupuesto de todos los departamentos.

```
SELECT ROUND(AVG(d.presupuesto), 2) AS media_presupuestos
FROM departamento d;
```

```
-- RESULTADO
147857.14
```

16. empleados Resumen 4

4. Calcula el nombre del departamento y el presupuesto que tiene asignado, el departamento con menor presupuesto (de entre los que tienen presupuesto asignado).

```
SELECT d.nombre, d.presupuesto AS presupuesto_minimo
FROM departamento d
WHERE d.presupuesto = (
    SELECT MIN(d.presupuesto)
    FROM departamento d
);
```

```
-- RESULTADO
nombre presupuesto_minimo
Proyectos      0
Publicidad     0
```

17. empleados Resumen 8

8. Calcula el número de empleados que no tienen NULL en su segundo apellido.

```
SELECT COUNT(e.codigo) AS cantidad_empleados
FROM empleado e
WHERE e.apellido2 IS NULL;
```

```
-- RESULTADO
2
```

18. empleados Resumen 10

10. Calcula el nombre de los departamentos que tienen más de 2 empleados. El resultado debe tener dos columnas, una con el nombre del departamento y otra con el número de empleados que tiene asignados.

```
SELECT
    d.nombre AS nombre_departamento,
    COUNT(e.codigo) AS cantidad_empleados
FROM empleado e INNER JOIN departamento d
    ON e.codigo_departamento = d.codigo
GROUP BY d.nombre
HAVING cantidad_empleados > 2;
```

```
-- RESULTADO
nombre_departamento cantidad_empleados
Desarrollo          3
Sistemas            3
```

19. empleados Subconsultas 4

4. Devuelve el nombre del departamento con mayor presupuesto y la cantidad que tiene asignada. Sin hacer uso de MAX, ORDER BY ni LIMIT.

```
SELECT d.nombre, d.presupuesto
FROM departamento d
WHERE d.presupuesto >= ALL (SELECT d1.presupuesto
                           FROM departamento d1);
```

```
-- RESULTADO
I+D 375000
```

20. empleados Subconsultas 6

6. Devuelve los nombres de los departamentos que tienen empleados asociados. (Utilizando ALL o ANY).

```
SELECT d.nombre
FROM departamento d
WHERE d.codigo = ANY (
    SELECT e.codigo_departamento
    FROM empleado e
);
```

```
-- RESULTADO
nombre
Desarrollo
Sistemas
Recursos Humanos
Contabilidad
I+D
```