

Implementación de la base de datos de DataSafe, empresa de seguridad bancaria

Código del proyecto: ICX0_P3

Producto 3 - Consultas avanzadas de base de datos

Alumnos:	Ana Real Tovar Jesús Real Tovar Raúl Vargas Molinero Marc Sánchez Sanz Enric Loren Parrondo
Consultor/ra:	Rita de la Torre Chirivella
Fecha de entrega:	18 de abril de 2019

Índice de Contenidos

1. Registros	1
2. Ejercicio 1 - Realizar las siguientes consultas y anotar la sentencia SQL y su salida en un documento	3
2.1. Haz un listado de todos los trabajadores que han sido trasladados a una agencia que no es su lugar de residencia. Añade el año en el que fueron trasladados.	3
2.2. Realiza un listado de los empleados que han solicitado un préstamo, y este no ha sido concedido. Añade al listado el código de préstamo.	4
2.3. Muestra el nombre de los empleados y la fecha en la que han sido trasladados, aquellos empleados que llevan menos de 5 años desde el último traslado. Muestra el resultado ordenado por el nombre y apellido del trabajador	4
2.4. Haz un listado de las agencias que en mediana trabajan los trabajadores más de una vez y media. En el listado se tiene que mostrar el código del trabajador, donde reside y la mediana de años que lleva el trabajador.	5
2.5. Muestra un listado de los empleados con titulación que estén afiliados al sindicato. .	6
2.6. Inventa una consulta que contenga un group by + having + agregación (count). . .	6
2.7. Inventa una consulta que contenga una combinación externa + ordenación (order by).	7
3. Ejercicio 2 - Realizar las siguientes modificaciones en los datos insertados y anotar la sentencia SQL y su salida en un documento	8
3.1. Habilitar/Deshabilitar Safe Mode Updates	8
3.2. Incrementa en 1 el número de veces que un trabajador ha sido trasladado, para todos aquellos trabajadores que llevan más de 5 años en la agencia bancaria.	8
3.3. Elimina los trabajadores que nunca han sido trasladados fuera de su ciudad de residencia.	9

1. Registros

```
1  INSERT INTO empleado (codigo_empleado, dni, nss, nombre, apellido, id_nombre_cat,
2  ↪ id_central, id_ciudad_res, email, telefono_movil)
3  VALUES
4  ('ppm80832', '45215844M', '2812645678', 'Pablo', 'Perez', 1, 1, 1, 'pperez@gmail.com',
5  ↪ '+34659223176'),
6  ('ars11234', '36584125P', '7279568404', 'Ana', 'Rio', 2, 4, 5, 'ario@gmail.com', '+34674734566'),
7  ('lcs52675', '15426895P', '7364952675', 'Laura', 'Caceres', 2, 4, 5, 'lcac@gmail.com',
8  ↪ '+34678956432'),
9  ('gtp56984', '95683452P', '7568941568', 'Gregorio', 'Trapero', 5, 3, 5, 'gregt@gmail.com',
10 ↪ '+34915869256'),
11 ('frl45763', '15486925P', '4896414587', 'Francisco', 'Roldan', 7, 2, 4, 'froid@gmail.com',
12 ↪ '+34916853654');
13
14 INSERT INTO fijo (codigo_empleado, antiguedad)
15 VALUES
16 ('ppm80832', '2012-07-24'),
17 ('ars11234', '2013-11-04'),
18 ('lcs52675', '2015-06-08'),
19 ('frl45763', '2013-10-16');
20
21 INSERT INTO titulacion (codigo_empleado, id_titulo)
22 VALUES
23 ('ppm80832', 1),
24 ('ars11234', 2),
25 ('lcs52675', 1),
26 ('frl45763', 2);
27
28 INSERT INTO tipoprestamo (codigo_prestamo, tipo_interes, vigencia)
29 VALUES
30 (11, 4.5, '2020-01-01'),
31 (12, 3.0, '2021-01-01'),
32 (13, 3.1, '2024-01-01'),
33 (14, 2.9, '2022-01-01'),
34 (15, 4.0, '2025-01-01'),
35 (16, 1.0, '2027-01-01');
36
37 INSERT INTO fecha (fecha)
38 VALUES
39 ('2013-07-24'),
40 ('2017-10-25'),
41 ('2014-10-04'),
42 ('2012-12-21'),
43 ('2012-05-13'),
44 ('2015-11-11'),
45 ('2016-01-07'),
46 ('2011-09-05'),
47 ('2014-02-28'),
48 ('2014-10-12'),
49 ('2013-12-25'),
50 ('2016-05-01'),
51 ('2011-12-03');
52
53 INSERT INTO peticion (codigo_empleado, id_codigo_prestamo, id_fecha, si_no)
54 VALUES
55 ('ppm80832', 11, 21, FALSE),
56 ('ars11234', 12, 22, FALSE),
57 ('mrr99583', 13, 23, FALSE),
58 ('mrg12375', 14, 24, FALSE),
59 ('lcs52675', 15, 25, FALSE),
```

```
55      ('frl45763', 16, 26, FALSE);
56
57  INSERT INTO traslado (codigo_empleado, id_fecha, id_ciudad, id_agencia, fecha_fin)
58  VALUES
59      ('ppm80832', 27, 2, 1, '2021-7-25'),
60      ('ars11234', 28, 4, 4, '2017-11-23'),
61      ('mrr99583', 29, 2, 5, '2015-1-29'),
62      ('mgg12375', 30, 6, 7, '2016-9-8'),
63      ('gtp56984', 31, 5, 8, '2015-10-16'),
64      ('frl45763', 32, 2, 1, '2017-9-4'),
65      ('lcs52675', 33, 2, 5, '2018-12-10');
```

2. Ejercicio 1 - Realizar las siguientes consultas y anotar la sentencia SQL y su salida en un documento

2.1. Haz un listado de todos los trabajadores que han sido trasladados a una agencia que no es su lugar de residencia. Añade el año en el que fueron trasladados.

```
1 SELECT e.codigo_empleado, e.nombre, e.apellido, e.id_ciudad_res, a.id_ciudad,  
   ↪ c.nombre_ciudad, a.nombre_agencia, YEAR(f.fecha) AS año FROM empleado AS e  
2 INNER JOIN traslado AS t ON t.id_agencia = e.id_central  
3 INNER JOIN ciudad AS c ON c.id = t.id_ciudad  
4 INNER JOIN agencia AS a ON a.id_ciudad = t.id_ciudad  
5 INNER JOIN fecha AS f ON f.id = t.id_fecha  
6 WHERE e.id_ciudad_res != a.id_ciudad;
```

codigo_empleado	nombre	apellido	id_ciudad_res	id_ciudad	nombre_ciudad	nombre_agencia	año
ars11234	Ana	Rio	5	4	Sevilla	agencia_04	2011
ars11234	Ana	Rio	5	4	Sevilla	agencia_04	2013
crl38160	Carmen	Ruiz	7	1	Madrid	agencia_03	2011
crl38160	Carmen	Ruiz	7	1	Madrid	agencia_09	2011
gtp56984	Gregorio	Trapero	5	1	Madrid	agencia_03	2016
gtp56984	Gregorio	Trapero	5	1	Madrid	agencia_09	2016
idl20862	Isabel	Díaz	6	1	Madrid	agencia_03	2016
idl20862	Isabel	Díaz	6	1	Madrid	agencia_09	2016
jrm89012	Juan	Rodríguez	1	2	Barcelona	agencia_01	2016
jrm89012	Juan	Rodríguez	1	2	Barcelona	agencia_05	2016
jrm89012	Juan	Rodríguez	1	2	Barcelona	agencia_01	2013
jrm89012	Juan	Rodríguez	1	2	Barcelona	agencia_05	2013
jrm89012	Juan	Rodríguez	1	2	Barcelona	agencia_01	2016
jrm89012	Juan	Rodríguez	1	2	Barcelona	agencia_05	2016
lcs52675	Laura	Caceres	5	4	Sevilla	agencia_04	2011

codigo_empleado	nombre	apellido	id_ciudad_res	id_ciudad	nombre_ciudad	nombre_agencia	año
jrm89012	Juan	Rodríguez	1	2	Barcelona	agencia_05	2016
lcs52675	Laura	Caceres	5	4	Sevilla	agencia_04	2011
lcs52675	Laura	Caceres	5	4	Sevilla	agencia_04	2013
mgg12375	María	González	8	6	Malaga	agencia_07	2019
mgg12375	María	González	8	6	Malaga	agencia_07	2014
mgs91128	Marcos	Gómez	5	4	Sevilla	agencia_04	2011
mgs91128	Marcos	Gómez	5	4	Sevilla	agencia_04	2013
mrr99583	Mario	Ramos	4	5	Zaragoza	agencia_08	2013
mrr99583	Mario	Ramos	4	5	Zaragoza	agencia_08	2017
ppm80832	Pablo	Perez	1	2	Barcelona	agencia_01	2016
ppm80832	Pablo	Perez	1	2	Barcelona	agencia_05	2016
ppm80832	Pablo	Perez	1	2	Barcelona	agencia_01	2013
ppm80832	Pablo	Perez	1	2	Barcelona	agencia_05	2013
ppm80832	Pablo	Perez	1	2	Barcelona	agencia_01	2016
ppm80832	Pablo	Perez	1	2	Barcelona	agencia_05	2016

2.2. Realiza un listado de los empleados que han solicitado un préstamo, y este no ha sido concedido. Añade al listado el código de préstamo.

```
1 SELECT e.nombre, e.apellido, t.codigo_prestamo FROM empleado AS e
2 INNER JOIN peticion AS p ON p.codigo_empleado = e.codigo_empleado
3 INNER JOIN tipoprestamo AS t ON t.id = p.id_codigo_prestamo
4 WHERE p.si_no = 0;
```



	nombre	apellido	codigo_prestamo
▶	Ana	Rio	12
	Francisco	Roldan	16
	Laura	Caceres	15
	María	González	14
	Mario	Ramos	13
	Pablo	Perez	11

2.3. Muestra el nombre de los empleados y la fecha en la que han sido trasladados, aquellos empleados que llevan menos de 5 años desde el último traslado. Muestra el resultado ordenado por el nombre y apellido del trabajador

```
1 SELECT e.nombre, e.apellido, f.fecha FROM empleado AS e
2 INNER JOIN traslado AS t ON t.codigo_empleado = e.codigo_empleado
3 INNER JOIN fecha AS f ON f.id = t.id_fecha
4 WHERE f.fecha > DATE_SUB(CURDATE(), INTERVAL 5 YEAR)
5 ORDER BY e.nombre, e.apellido;
```



	nombre	apellido	fecha
▶	Claudia	López	2019-03-09 00:00:00
	Francisco	Roldan	2016-05-01 00:00:00
	José	García	2016-10-08 00:00:00
	Laura	Álvarez	2015-11-02 00:00:00
	Marcos	Gómez	2014-09-16 00:00:00
	María	González	2014-10-12 00:00:00
	Mario	Ramos	2017-05-05 00:00:00
	Pablo	Perez	2016-01-07 00:00:00
	Roger	Martín	2017-04-13 00:00:00

2.4. Haz un listado de las agencias que en mediana trabajan los trabajadores más de una vez y media. En el listado se tiene que mostrar el código del trabajador, donde reside y la mediana de años que lleva el trabajador.

```

1  SET sql_mode=(SELECT REPLACE(@@sql_mode, 'ONLY_FULL_GROUP_BY', ''));
2
3  CREATE OR REPLACE VIEW traslados_trabajadores AS
4  SELECT CONCAT(t.id_fecha, '_', t.codigo_empleado) AS id, a.nombre_agencia, a.id AS id_agencia,
   ↳ e.codigo_empleado, c.nombre_ciudad AS ciudad_de_residencia, TIMESTAMPDIFF(YEAR, f.fecha,
   ↳ t.fecha_fin) AS años
5  FROM agencia AS a
6  INNER JOIN traslado AS t ON t.id_agencia = a.id
7  INNER JOIN empleado AS e ON e.codigo_empleado = t.codigo_empleado
8  INNER JOIN ciudad AS c ON c.id = e.id_ciudad_res
9  INNER JOIN fecha AS f ON f.id = t.id_fecha
10 GROUP BY id;
11
12 CREATE OR REPLACE VIEW numero_traslados AS
13 SELECT t.id_agencia, COUNT(t.id_agencia) AS n FROM traslado AS t
14 GROUP BY t.id_agencia;
15
16 CREATE OR REPLACE VIEW media_años_trabajados AS
17 SELECT tt.codigo_empleado, AVG(tt.años) AS media_de_años FROM traslados_trabajadores AS tt
18 GROUP BY tt.codigo_empleado
19 ORDER BY tt.codigo_empleado;
20
21 SELECT tt.nombre_agencia, tt.id_agencia, tt.codigo_empleado, tt.ciudad_de_residencia,
   ↳ mt.media_de_años FROM traslados_trabajadores AS tt
22 INNER JOIN numero_traslados AS nt
23 INNER JOIN media_años_trabajados AS mt ON mt.codigo_empleado = tt.codigo_empleado
24 WHERE nt.id_agencia = tt.id_agencia
25 GROUP BY tt.id
26 HAVING AVG(nt.n) > 1.5
27 ORDER BY tt.nombre_agencia;
28
29 DROP VIEW IF EXISTS traslados_trabajadores;
30 DROP VIEW IF EXISTS numero_traslados;
31 DROP VIEW IF EXISTS media_años_trabajados;

```

nombre_agencia	id_agencia	codigo_empleado	ciudad_de_residencia	media_de_años
agencia_01	1	jrm89012	Madrid	3.0000
agencia_01	1	ppm80832	Madrid	5.0000
agencia_01	1	fri45763	Sevilla	1.0000
agencia_04	4	ars11234	Zaragoza	6.0000
agencia_04	4	mgg12375	Palma	1.0000
agencia_05	5	mrr99583	Sevilla	0.5000
agencia_05	5	lcs52675	Zaragoza	7.0000
agencia_05	5	rms10184	Bilbao	2.0000
agencia_07	7	mgg12375	Palma	1.0000
agencia_08	8	mrr99583	Sevilla	0.5000
agencia_08	8	gtp56984	Zaragoza	1.0000

2.5. Muestra un listado de los empleados con titulación que estén afiliados al sindicato.


```
1 SELECT e.codigo_empleado, e.nombre, e.apellido, s.nombre_sindicato, ti.nombre_titulo
   ↪ FROM empleado AS e
2     INNER JOIN titulacion AS t ON t.codigo_empleado = e.codigo_empleado
3     INNER JOIN sindicato AS s ON s.id = e.id_central
4     INNER JOIN titulo AS ti ON ti.id = t.id_titulo
5     WHERE t.id_titulo IS NOT NULL
6     AND e.id_central IS NOT NULL;
```

	codigo_empleado	nombre	apellido	nombre_sindicato	nombre_titulo
▶	jrm89012	Juan	Rodríguez	UGT	E.S.O.
	lcs52675	Laura	Caceres	CGT	E.S.O.
	ppm80832	Pablo	Perez	UGT	E.S.O.
	ars11234	Ana	Rio	CGT	Bachiller
	fri45763	Francisco	Roldan	CCOO	Bachiller
	mgs91128	Marcos	Gómez	CGT	Bachiller
	jgr23778	José	García	SAT	CFGS_DAM
	mgs12375	María	González	LAB	CFGS_DAW
	rms10184	Roger	Martín	COS	CFGS_ASIR
	cri38160	Carmen	Ruiz	AST	CFGS_AF
	clm03866	Claudia	López	CCOO	CFGS_MP
	mrr99583	Mario	Ramos	SO	CFGS_GVEC
	idi20862	Isabel	Díaz	CNT	G_ADE
	lap33670	Laura	Álvarez	CIG	G_II

2.6. Inventa una consulta que contenga un group by + having + agregación (count).

```
1 /*Hacer un listado de empleados con peticiones de prestamos con más de 1 petición*/
2 SELECT e.codigo_empleado, e.nombre, e.apellido, COUNT(p.codigo_empleado) AS
   ↪ numero_peticiones FROM empleado AS e
3     INNER JOIN peticion AS p ON e.codigo_empleado = p.codigo_empleado
4     GROUP BY codigo_empleado
5     HAVING numero_peticiones > 1;
```

```
1 /*Filtra el número de peticiones que ha hecho cada empleado, mostrando dicho número y su
   ↪ dni*/
2 SELECT empleado.dni, COUNT(peticion.id_codigo_prestamo) AS PeticionesPorEmpleado
3 FROM peticion
4 INNER JOIN empleado ON peticion.codigo_empleado = empleado.codigo_empleado
5 GROUP BY empleado.dni
6 HAVING COUNT(peticion.id_codigo_prestamo) > 0;
```


Result Grid		 Filter Rows:	
	dni	PeticionesPorEmpleado	
▶	15426895P	1	
	15486925P	1	
	19983245Y	1	
	36584125P	1	
	39573446T	2	
	43597258K	3	
	45215844M	1	
	45678844M	3	
	87977552P	1	
	89234713M	2	

2.7. Inventa una consulta que contenga una combinación externa + ordenación (order by).

```
1  /*Mostrar listado de empleados y su antigüedad como fijos ordenados por antigüedad
   ↪ ascendentemente. */
2  SELECT e.codigo_empleado, e.nombre, e.apellido, f.antigüedad FROM empleado AS e
3  RIGHT OUTER JOIN fijo AS f ON f.codigo_empleado = e.codigo_empleado
4  ORDER BY f.antigüedad ASC;
```

Result Grid		Filter Rows:	Export:	Wra
codigo_empleado	nombre	apellido	antigüedad	
jrm89012	Juan	Rodríguez	2012-07-24 00:00:00	
ppm80832	Pablo	Perez	2012-07-24 00:00:00	
cr138160	Carmen	Ruiz	2013-05-21 00:00:00	
fri45763	Francisco	Roldan	2013-10-16 00:00:00	
mvg12375	Maria	González	2013-10-16 00:00:00	
ars11234	Ana	Rio	2013-11-04 00:00:00	
mgs91128	Marcos	Gómez	2013-11-04 00:00:00	
idi20862	Isabel	Díaz	2013-12-05 00:00:00	
lap33670	Laura	Álvarez	2014-08-12 00:00:00	
jgr23778	José	García	2015-06-08 00:00:00	
lcs52675	Laura	Caceres	2015-06-08 00:00:00	
mrr99583	Mario	Ramos	2015-09-02 00:00:00	
rms10184	Roger	Martín	2016-01-07 00:00:00	
clm03866	Claudia	López	2017-06-15 00:00:00	

```
1  /* Empleados temporales ordenados por fecha de inicio */
2  SELECT e.codigo_empleado, e.nombre, e.apellido, t.fecha_inicio_cont FROM empleado AS e
3  RIGHT OUTER JOIN temporal AS t ON t.codigo_empleado = e.codigo_empleado
4  ORDER BY t.fecha_inicio_cont ASC;
```

Result Grid		Filter Rows:	Export:	Wra
codigo_empleado	nombre	apellido	fecha_inicio_cont	
jrm89012	Juan	Rodríguez	2012-07-24 00:00:00	
cr138160	Carmen	Ruiz	2013-05-21 00:00:00	
mvg12375	Maria	González	2013-10-16 00:00:00	
mgs91128	Marcos	Gómez	2013-11-04 00:00:00	
idi20862	Isabel	Díaz	2013-12-05 00:00:00	
lap33670	Laura	Álvarez	2014-08-12 00:00:00	
jgr23778	José	García	2015-06-08 00:00:00	
mrr99583	Mario	Ramos	2015-09-02 00:00:00	
rms10184	Roger	Martín	2016-01-07 00:00:00	
clm03866	Claudia	López	2017-06-15 00:00:00	

3. Ejercicio 2 - Realizar las siguientes modificaciones en los datos insertados y anotar la sentencia SQL y su salida en un documento

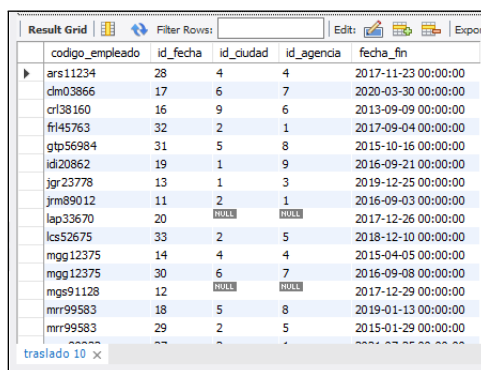
3.1. Habilitar/Deshabilitar Safe Mode Updates

```
1  /* Deshabilitamos safe mode update */
2  SET SQL_SAFE_UPDATES = 0;
```

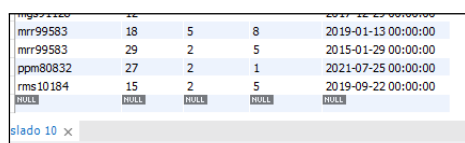
```
1  /* Habilitamos safe mode update */
2  SET SQL_SAFE_UPDATES = 1;
```

3.2. Incrementa en 1 el número de veces que un trabajador ha sido trasladado, para todos aquellos trabajadores que llevan más de 5 años en la agencia bancaria.

```
1  INSERT INTO fecha (fecha) VALUES (NOW());
2
3  INSERT INTO traslado (codigo_empleado, id_fecha, id_ciudad, id_agencia, fecha_fin)
4  SELECT t.codigo_empleado, (SELECT f.id FROM fecha AS f ORDER BY f.id DESC LIMIT
   ↳ 1) AS id_fecha, t.id_ciudad, t.id_agencia, DATE_ADD((SELECT f.fecha FROM fecha AS f
   ↳ ORDER BY f.id DESC LIMIT 1), INTERVAL 5 YEAR) as fecha_fin
5  FROM traslado AS t
6  INNER JOIN fecha AS f ON f.id = t.id_fecha
7  WHERE YEAR(f.fecha) <= YEAR(SYSDATE()) - 5
8  AND t.id_ciudad IS NOT NULL
9  AND id_agencia IS NOT NULL;
```



codigo_empleado	id_fecha	id_ciudad	id_agencia	fecha_fin
ars11234	28	4	4	2017-11-23 00:00:00
dm03866	17	6	7	2020-03-30 00:00:00
cr138160	16	9	6	2013-09-09 00:00:00
fr145763	32	2	1	2017-09-04 00:00:00
gtp56984	31	5	8	2015-10-16 00:00:00
idi20862	19	1	9	2016-09-21 00:00:00
jgr23778	13	1	3	2019-12-25 00:00:00
jrm89012	11	2	1	2016-09-03 00:00:00
lap33670	20	NULL	NULL	2017-12-26 00:00:00
lcs52675	33	2	5	2018-12-10 00:00:00
mgg12375	14	4	4	2015-04-05 00:00:00
mgg12375	30	6	7	2016-09-08 00:00:00
mgs91128	12	NULL	NULL	2017-12-29 00:00:00
mrr99583	18	5	8	2019-01-13 00:00:00
mrr99583	29	2	5	2015-01-29 00:00:00



codigo_empleado	id_fecha	id_ciudad	id_agencia	fecha_fin
mrr99583	18	5	8	2019-01-13 00:00:00
mrr99583	29	2	5	2015-01-29 00:00:00
ppm80832	27	2	1	2021-07-25 00:00:00
rms10184	15	2	5	2019-09-22 00:00:00
NULL	NULL	NULL	NULL	NULL

3.3. Elimina los trabajadores que nunca han sido trasladados fuera de su ciudad de residencia.

```

1  CREATE OR REPLACE VIEW traslados_fuera AS
2  SELECT e.codigo_empleado AS codigo_empleado FROM empleado AS e
3  INNER JOIN traslado AS t ON t.codigo_empleado = e.codigo_empleado
4  WHERE e.id_ciudad_res != t.id_ciudad;
5
6  CREATE OR REPLACE VIEW traslados_dentro AS
7  SELECT e.codigo_empleado AS codigo_empleado FROM empleado AS e
8  INNER JOIN traslado AS t ON t.codigo_empleado = e.codigo_empleado
9  WHERE e.id_ciudad_res = t.id_ciudad;
10
11 CREATE OR REPLACE VIEW traslados_null AS
12 SELECT e.codigo_empleado AS codigo_empleado FROM empleado AS e
13 INNER JOIN traslado AS t ON t.codigo_empleado = e.codigo_empleado
14 WHERE e.id_ciudad_res IS NULL OR t.id_ciudad IS NULL;
15
16 DELETE FROM empleado WHERE codigo_empleado IN (
17     SELECT codigo_empleado FROM (
18         SELECT e.codigo_empleado AS codigo_empleado FROM empleado AS e
19         WHERE e.codigo_empleado NOT IN (SELECT codigo_empleado FROM traslado) OR
20         e.codigo_empleado IN (SELECT codigo_empleado FROM traslados_null) OR
21         (e.codigo_empleado NOT IN (SELECT codigo_empleado FROM traslados_fuera)
22     AND
23         e.codigo_empleado IN (SELECT codigo_empleado FROM traslados_dentro))
24     ) AS codigo_empleado
25 );
26 DROP VIEW IF EXISTS traslados_fuera;
27 DROP VIEW IF EXISTS traslados_dentro;
28 DROP VIEW IF EXISTS traslados_null;

```

codigo_empleado	dni	nss	nombre	apellido	id_nombre_cat	id_central	id_ciudad_res	telefono_movil	email
ars11234	36584125P	7279568404	Ana	Rio	2	4	5	+34674734566	ario@gmail.com
cr138160	39573446T	7821437691	Carmen	Ruiz	9	9	7	+34622196890	cruiz@gmail.com
fr145763	15486925P	4896414587	Francisco	Roldan	7	2	4	+34916853654	froid@gmail.com
idi20862	65489902L	2021125989	Isabel	Díaz	5	3	6	+34680026491	idiaz@gmail.com
jgr23778	38923235G	5437635590	José	García	7	5	2	+34665812239	jpgarcia@gmail.com
jrm89012	45678844M	2812345678	Juan	Rodríguez	1	1	1	+34659278576	jrodriguez@gmail.com
lcs52675	15426895P	7364952675	Laura	Caceres	2	4	5	+34678956432	lcac@gmail.com
mgg12375	89234713M	3732997415	María	González	4	7	8	+34653389124	mgonzalez@gmail.com
mrr99583	43597258K	2237984169	Mario	Ramos	3	8	4	+34672830755	mramos@gmail.com
ppm80832	45215844M	2812645678	Pablo	Perez	1	1	1	+34659223176	pperez@gmail.com
rms10184	94571879U	3696581102	Roger	Martín	8	6	9	+34698470415	rmartin@gmail.com
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL