

# SQL El lenguaje de Consulta Estructurado

Manejo de valores nulos,  
Reunión

## Manejo de valores nulos

- SQL utiliza una lógica trivaluada:  
TRUE, FALSE, NULL.
- La aplicación de una función a un valor nulo devuelve un valor nulo.  
$$F(\text{NULL}, \dots) = \text{NULL}$$
- La evaluación de un predicado donde una de sus partes es nula devuelve nulo.  $P(\text{NULL}, \dots) = \text{NULL}$ 
  - ¡Ojo! NULL es valor de verdad diferente de TRUE y de FALSE.

## Manejo de valores nulos

- `expr1 IS [NOT] NULL` devuelve TRUE si `expr1` es (no es) NULL.
  - Diferente de `expr1 = NULL`.
- `NVL (expr1, expr2)` devuelve
  - `expr1` si `expr1 IS NOT NULL`
  - `expr2` si `expr1 IS NULL`
- `expr1` y `expr2` han de ser del mismo tipo

## Manejo de valores nulos

- Listar el nombre y primer apellido de los profesores que no tienen email

```
SELECT Nombre, Apellido1  
FROM profesores  
WHERE email IS NULL;
```

NOMBRE	APELLIDO1
Michael	Brown
Miguel	Ortiz
Miguel	Hermoso
Juana	Hernandez
Antonio	Villanueva

```
SELECT Nombre, Apellido1  
FROM profesores  
WHERE email = NULL;
```

NOMBRE	APELLIDO1
--------	-----------

Ninguna fila se selecciona  
porque ninguna hace TRUE  
el predicado

## Manejo de valores nulos

- Listar el primer apellido y el código del director de tesis de cada profesor. En caso de no tener, se muestra 'No tiene'

```
SELECT apellido1,  
       NVL(director_tesis, 'No Tiene')  
FROM profesores;
```

⚡ APELLIDO1	⚡ NVL(DIRECTOR_TESIS,'NOTIENE')
Brown	A-34-CEU-01
Enciso	C-34-TU-00
Soler	No Tiene
Ortiz	A-89-CEU-99
Roldán	A-89-CEU-99
Gálvez	C-34-TU-00
Fernández	No Tiene
Jiménez	No Tiene
Hermoso	H-32-TU-99
Hernandez	No Tiene
Villanueva	No Tiene

## Reunión (join)

- Operación adecuada para enlazar información de dos tablas. Conceptualmente, se puede entender como un producto cartesiano al que se le aplica una selección posterior.

```
SELECT lista_atributos  
FROM lista_tablas  
WHERE condiciones_de_reunión
```

- Condición\_de\_reunión*:  $\text{tabla1.atributo1} = \text{tabla2.atributo2}$
- Una reunión correcta de  $n$  tablas necesitará, al menos,  $n-1$  condiciones de reunión.
- La operación de la reunión evita la ejecución del producto cartesiano y, por consiguiente, su alto coste asociado.

## Reunión. Ejemplo paso a paso

- Mostrar el código de las asignaturas asignadas a Manuel Enciso
- Vamos a definir la consulta en SQL de forma constructiva.

## Reunión. Ejemplo paso a paso

- Mostrar asignación docente añadiendo datos del profesor correspondiente.

```
SELECT *  
FROM Impartir I, profesores P;
```

¡¡Hace falta la condición de  
reunión!!

PROFESOR	ASIGNATURA	GRUPO	CURSO	CARGA_CREDITOS	ID	NOMBRE	APELLIDO1	APELLIDO2	DEPARTAMENTO	TELEFONO	EMAIL	DESPACHO
C-34-TU-00	112 B	15/16		7	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
C-34-TU-00	114 A	15/16		6	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
D-92-PC-92	113 A	15/16		6	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
D-92-PC-92	113 B	15/16		3	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
C-67-CEU-98	116 A	15/16		6	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
A-89-CEU-99	112 A	15/16		7	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
A-89-CEU-99	140 A	15/16		(null)	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
G-34-TEU-96	140 A	15/16		(null)	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
G-34-TEU-96	112 B	15/16		(null)	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
H-77-TU-99	123 A	15/16		4	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
H-32-TU-99	123 A	13/14		4	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
H-32-TU-99	114 C	15/16		6	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
F-12-AY-02	133 A	15/16		6	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
F-12-AY-02	101 B	15/16		7	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
F-12-AY-02	133 A	14/15		6	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
F-12-AY-02	101 B	14/15		7	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
C-34-TU-00	140 B	14/15		9	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)
C-34-TU-00	140 A	15/16		9	J-98766	Michael	Brown	(null)	4	(null)	(null)	(null)

...



## Reunión. Ejemplo paso a paso

- Mostrar asignación docente añadiendo datos del profesor correspondiente.

```
SELECT *  
FROM Impartir I, profesores P  
WHERE I.profesor=P.id;
```

PROFESOR	ASIGNATURA	GRUPO	CURSO	CARGA_CREDITOS	ID	NOMBRE	APELLIDO1	APELLIDO2	DEPARTAMENTO	TELEFONO	EMAIL
A-89-CEU-99	112 A	15/16		7	A-89-CEU-99	Manuel	Enciso	Garcia-Oliveros		1 3309	enciso_lcc@uma.es
A-89-CEU-99	140 A	15/16		(null)	A-89-CEU-99	Manuel	Enciso	Garcia-Oliveros		1 3309	enciso_lcc@uma.es
C-34-TU-00	140 A	13/14		4	C-34-TU-00	Enrique	Soler	Castillo		2 2789	esc@gisum.lcc.uma.es
C-34-TU-00	112 B	15/16		7	C-34-TU-00	Enrique	Soler	Castillo		2 2789	esc@gisum.lcc.uma.es
C-34-TU-00	114 A	15/16		6	C-34-TU-00	Enrique	Soler	Castillo		2 2789	esc@gisum.lcc.uma.es
C-34-TU-00	140 A	15/16		9	C-34-TU-00	Enrique	Soler	Castillo		2 2789	esc@gisum.lcc.uma.es
C-34-TU-00	140 B	14/15		9	C-34-TU-00	Enrique	Soler	Castillo		2 2789	esc@gisum.lcc.uma.es
C-67-CEU-98	116 A	15/16		6	C-67-CEU-98	Miguel	Ortiz	(null)		1 2875	(null)
D-92-PC-92	113 B	15/16		3	D-92-PC-92	Maria del Mar	Roldán	García		2 2875	mar@ctima.uma.es
D-92-PC-92	113 A	15/16		6	D-92-PC-92	Maria del Mar	Roldán	García		2 2875	mar@ctima.uma.es
F-12-AY-02	133 A	15/16		6	F-12-AY-02	Sergio	Gálvez	Rojas		1 7151	cesar.lcc.uma.es@uma.es
F-12-AY-02	101 B	14/15		7	F-12-AY-02	Sergio	Gálvez	Rojas		1 7151	cesar.lcc.uma.es@uma.es
F-12-AY-02	133 A	14/15		6	F-12-AY-02	Sergio	Gálvez	Rojas		1 7151	cesar.lcc.uma.es@uma.es
F-12-AY-02	101 B	15/16		7	F-12-AY-02	Sergio	Gálvez	Rojas		1 7151	cesar.lcc.uma.es@uma.es
G-34-TEU-96	112 B	15/16		(null)	G-34-TEU-96	Ana	Jiménez	Santos		3 (null)	rqg@lccuma.es
G-34-TEU-96	140 A	15/16		(null)	G-34-TEU-96	Ana	Jiménez	Santos		3 (null)	rqg@lccuma.es

...

## Reunión. Ejemplo paso a paso

- Mostrar asignación de docencia para el profesor Manuel Enciso

```
SELECT *  
FROM impartir I, profesores P  
WHERE I.profesor=P.id  
AND UPPER(P.Nombre) = 'MANUEL'  
AND UPPER(P.Apellido1) = 'ENCISO';
```

PROFESOR	ASIGNATURA	GRUPO	CURSO	CARGA_CREDITOS	ID	NOMBRE	APELLIDO1	APELLIDO2	DEPARTAMENTO	TELEFONO	EMAIL	
A-89-CEU-99	112	A	15/16	7	A-89-CEU-99	Manuel	Enciso	Garcia-Oliveros	1	3309	enciso_lcc@uma.es	...
A-89-CEU-99	140	A	15/16	(null)	A-89-CEU-99	Manuel	Enciso	Garcia-Oliveros	1	3309	enciso_lcc@uma.es	

## Reunión. Ejemplo paso a paso

- Mostrar el código de las asignaturas asignadas a Manuel Enciso

```
SELECT I.asignatura "Codigo Asignatura"  
FROM impartir I, profesores P  
WHERE I.profesor=P.id  
AND UPPER(P.Nombre) = 'MANUEL'  
AND UPPER(P.Apellido1) = 'ENCISO';
```

Codigo Asignatura
112
140

## Self join

- Reunión de una tabla consigo misma.
- Utilizar alias de tablas y atributos es fundamental para distinguir cuál de las dos instancias de la tabla estamos usando.

```
SELECT      t1.atrib_n "atribn tabla1",  
            t2.atrib_n "atribn tabla2"  
FROM tabla t1, tabla t2  
WHERE t1.atrib_i = t2.atrib_j
```

## Self join

- Mostrar los nombres de las parejas de profesores que pertenecen al mismo departamento.

```
SELECT P1.Nombre || ' ' || P1.Apellido1 "Nombre1",  
       P2.Nombre || ' ' || P2.Apellido1 "Nombre2"  
FROM Profesores P1, Profesores P2  
WHERE P1.departamento = P2.departamento;
```

¡¡Se empareja cada profesor consigo mismo!!

Nombre1	Nombre2
Antonio Villanueva	Michael Brown
Juana Hernandez	Michael Brown
Miguel Hermoso	Michael Brown
Michael Brown	Michael Brown
Sergio Gálvez	Manuel Enciso
Miguel Ortiz	Manuel Enciso
Manuel Enciso	Manuel Enciso
Maria del Mar Roldán	Enrique Soler
Enrique Soler	Enrique Soler
Sergio Gálvez	Miguel Ortiz
Miguel Ortiz	Miguel Ortiz
Manuel Enciso	Miguel Ortiz
Maria del Mar Roldán	Maria del Mar Roldán
Enrique Soler	Maria del Mar Roldán
Sergio Gálvez	Sergio Gálvez
Miguel Ortiz	Sergio Gálvez

...

## Self join

- Mostrar los nombres de las parejas de profesores que pertenecen al mismo departamento.

```
SELECT P1.Nombre || ' ' || P1.Apellido1 "Nombre1",  
       P2.Nombre || ' ' || P2.Apellido1 "Nombre2"  
FROM Profesores P1, Profesores P2  
WHERE P1.departamento = P2.departamento  
      AND P1.id != P2.id
```

¡¡Se repiten las parejas!!

Nombre1	Nombre2
Antonio Villanueva	Michael Brown
Juana Hernandez	Michael Brown
Miguel Hermoso	Michael Brown
Sergio Gálvez	Manuel Enciso
Miguel Ortiz	Manuel Enciso
Maria del Mar Roldán	Enrique Soler
Sergio Gálvez	Miguel Ortiz
Manuel Enciso	Miguel Ortiz
Enrique Soler	Maria del Mar Roldán
Miguel Ortiz	Sergio Gálvez
Manuel Enciso	Sergio Gálvez
Ana Jiménez	Carlos Fernández
Carlos Fernández	Ana Jiménez
Antonio Villanueva	Miguel Hermoso
Juana Hernandez	Miguel Hermoso
Michael Brown	Miguel Hermoso
Antonio Villanueva	Juana Hernandez

## Self join

- Mostrar los nombres de las parejas de profesores que pertenecen al mismo departamento.

```
SELECT P1.Nombre || ' ' || P1.Apellido1 "Nombre1",  
       P2.Nombre || ' ' || P2.Apellido1 "Nombre2"  
FROM Profesores P1, Profesores P2  
WHERE P1.departamento = P2.departamento  
      AND P1.id < P2.id
```

Nombre1	Nombre2
Antonio Villanueva	Michael Brown
Juana Hernandez	Michael Brown
Manuel Enciso	Miguel Ortiz
Enrique Soler	Maria del Mar Roldán
Miguel Ortiz	Sergio Gálvez
Manuel Enciso	Sergio Gálvez
Ana Jiménez	Carlos Fernández
Antonio Villanueva	Miguel Hermoso
Juana Hernandez	Miguel Hermoso
Michael Brown	Miguel Hermoso
Antonio Villanueva	Juana Hernandez

## Combinación de self Join y reunión

- Hay que tener especial cuidado para que no se produzca el producto cartesiano.
- Ej: Mostrar el nombre de las parejas de alumnos que han estado matriculados en las mismas asignaturas.

```
SELECT AL1.Nombre, AL2.Nombre
FROM Alumnos AL1, Matricular M1,
     Alumnos AL2, Matricular M2
WHERE AL1.dni = M1.Alumno
      AND AL2.dni = M2.Alumno
      AND M1.Asignatura = M2.Asignatura
      AND AL1.dni < AL2.dni;
```

¡¡Si los alumnos coinciden en más de una asignatura salen repeticiones!!

NOMBRE	NOMBRE_1
Stephan	ANTONIO JESUS
Stephan	SERGIO
Stephan	DANIEL
Stephan	RAFAEL
Stephan	ADRIAN
Stephan	ALFONSO
Stephan	Benjamín
Stephan	Sergio
Stephan	Carlos Alberto
Stephan	Carlos Alberto
Stephan	Manuel
Stephan	Jaime
Stephan	Rafael
Stephan	Rafael
Stephan	Isidro
Stephan	Isidro

...



## Combinación de self Join y reunión

- Si deseamos eliminar todas las repeticiones en este tipo de consultas habrá que usar DISTINCT.
- Ej: Mostrar el nombre de las parejas de alumnos que han estado matriculados en las mismas asignaturas.

```
SELECT DISTINCT AL1.Nombre, AL2.Nombre
FROM   Alumnos AL1, Matricular M1,
        Alumnos AL2, Matricular M2
WHERE  AL1.dni = M1.Alumno
        AND AL2.dni = M2.Alumno
        AND M1.Asignatura = M2. Asignatura
        AND AL1.dni < AL2.dni;
```

NOMBRE	NOMBRE_1
Stephan	ALFONSO
Stephan	Jose
Stephan	Arcadio
Stephan	Carlos
Stephan	PABLO
Stephan	Raziz
Stephan	Angel
Stephan	Marlon
Stephan	Esther
Benjamín	RAFAEL
Benjamín	ELOY
Benjamín	RUBEN
Benjamín	Carlos Alberto
Benjamín	Justo
Benjamín	Laura
Benjamín	ISABEL

## Natural Join

- Hay varias formas de especificar la operación de reunión completamente en la Cláusula FROM.
  - FROM *tabla1* **NATURAL JOIN** *tabla2*
  - FROM *tabla1* **JOIN** *tabla2* **USING** *atributos*
  - FROM *tabla1* **JOIN** *tabla2* **ON** *condición\_reunión*
  - FROM *tabla1* **OUTER JOIN** *tabla2* **ON** *condición\_reunión*
- Siempre habrá una consulta SQL equivalente especificando la condición de reunión en el WHERE.
- La reunión natural (**NATURAL JOIN**) realiza la reunión utilizando todos los atributos comunes (columnas con el mismo nombre) de las dos tablas.
  - Por tanto, en el resultado no habrá columnas repetidas (innecesarios los alias de tabla)

## Natural Join

- Mostrar un listado donde a cada alumno del curso 15/16 se le añada el identificador de cada profesor que le da clase.

```
SELECT M.Alumno, I.Profesor  
FROM impartir I, matricular M  
WHERE I.Asignatura = M.asignatura  
AND I.Grupo = M.Grupo  
AND I.Curso = M.Curso  
AND I.Curso = '15/16';
```

Condición  
de reunión

ALUMNO	PROFESOR
12312342	G-34-TEU-96
12312342	C-34-TU-00
12312342	C-67-CEU-98
25748955	G-34-TEU-96
25748955	C-34-TU-00
34234244	D-92-PC-92
34234444	A-89-CEU-99
45634322	C-67-CEU-98
46563277	D-92-PC-92
46563277	C-67-CEU-98
54354355	G-34-TEU-96
54354355	C-34-TU-00
67867433	D-92-PC-92
67867888	C-67-CEU-98
23456456	D-92-PC-92

```
SELECT Alumno, Profesor  
FROM impartir NATURAL JOIN matricular  
WHERE Curso = '15/16';
```

## Natural Join

- **Problema:** se utilizan TODOS los atributos con nombre idéntico.  
A veces no interesa reunir utilizando todos los atributos, sino sólo alguno de ellos.
- **Soluciones:**
  1. Utilizar otro tipo de reunión.
  2. Crear tablas temporales en el FROM.
- Efecto secundario de la solución 2:
  - determinación de orden de ejecución.

- Suponiendo que el primer apellido de profesores y alumnos fuera único, mostrar nombre y fecha de matrícula de los profesores que son también alumnos

```
SELECT P.Nombre, Fecha_prim_matricula
FROM profesores P, alumnos A
WHERE P.apellido1 = A.apellido1 ;
```

NOMBRE	FECHA_PRIM_MATRICULA
Antonio	30/08/15
Antonio	30/08/14
Antonio	10/09/13
Miguel	05/09/15
Antonio	11/09/13

```
SELECT Nombre, Fecha_prim_matricula
FROM profesores NATURAL JOIN
(alumnos sin atributos comunes excepto apellido1);
```

```
SELECT Nombre, Fecha_prim_matricula
FROM profesores NATURAL JOIN
(SELECT Apellido1, Fecha_prim_matricula FROM Alumnos);
```

## Join Using

- Operación de reunión que permite especificar los atributos **comunes** que se utilizarán en las condiciones de reunión.

FROM *tabla1* JOIN *tabla2* USING ( *atributos\_seleccionados* )

- atributos\_seleccionados*:  
    atributo\_común1 , atributo\_común2 , ...
- El resto de atributos comunes no seleccionados se tratan como atributos normales y no se utilizan en la condición de reunión
- Los alias de tabla serán necesarios si hay atributos comunes no seleccionados con USING que queremos utilizar en la proyección.

## Join Using

- Suponiendo que el primer apellido de profesores y alumnos fuera único, mostrar nombre y fecha de matrícula de los profesores que son también alumnos

```
SELECT P.Nombre, A.Fecha_prim_matricula  
FROM Profesores P JOIN Alumnos A USING (Apellido1);
```

## Conditional Join

- **Problema:** si los atributos de reunión no se llaman igual, no podemos utilizar NATURAL JOIN ni JOIN USING.
- **Solución:** especificar la condición de reunión que queremos, indicando los atributos que se solapan en las dos tablas.

FROM *tabla1* JOIN *tabla2* ON *condición\_reunión*



## Conditional Join

- Mostrar el numero de tramos de investigación del profesor Enrique Soler

```
SELECT I.Tramos
FROM Profesores P, Investigadores I
WHERE P.Id = I.Id_profesor
      AND upper(P.Nombre) = 'ENRIQUE'
      AND upper(P.Apellido1) = 'SOLER';
```



TRAMOS
2

```
SELECT I.Tramos
FROM Profesores P JOIN Investigadores I ON P.Id = I.Id_profesor
WHERE upper(P.Nombre) = 'ENRIQUE'
      AND upper(P.Apellido1) = 'SOLER' ;
```

## Ejercicios

- Mostrar ...
  1. Nombre y nota de los alumnos que han aprobado Bases de Datos.
  2. Nombre del alumno junto a nombre de asignatura y turno en el que está matriculado.
  3. Nombre de alumno, nombre de asignatura aprobada, la nota en ella y el nombre del profesor de la asignatura.

## Outer Join

- La reunión externa (**OUTER JOIN**) de dos tablas *A* y *B* es una operación de reunión que devuelve todas las filas que cumplen la condición de reunión (como un JOIN normal) y, además, aquellas filas de la tabla *A* y/o la tabla *B* que no cumplen la condición de reunión.

FROM *tabla1* [LEFT/RIGHT/FULL] **OUTER JOIN** *tabla2* **ON** *condición\_reunión*

- A **LEFT** OUTER JOIN *B* añade las filas de *A* que no cumplen *condición\_reunión*
- A **RIGHT** OUTER JOIN *B* añade las filas de *B* que no cumplen *condición\_reunión*
- A **FULL** OUTER JOIN *B* añade las filas de *A* y *B* que no cumplen *condición\_reunión*.

## Outer Join

- Hacer un listado de **todas** las asignaturas, que muestre el nombre de asignatura junto al nombre de la materia asociada

```
SELECT A.Nombre "ASIGNATURA", M.Nombre "MATERIA"  
FROM Asignaturas A, Materias M  
WHERE A.Cod_materia = M.Codigo;
```

ASIGNATURA	MATERIA
Programación Orientada a Objetos	Ingeniería del Software
Ingeniería Web	Ingeniería del Software
Administración de Bases de Datos	Ingeniería del Software
Bases de Datos	Ingeniería del Software
Estructura de Computadores	Dispositivos hardware
Computación Altas Prestaciones	Dispositivos hardware
Dispositivos Electronicos	Dispositivos hardware
Prácticas en empresa	Complementos de formación
Estadística	Fundamentos de la Informática
Calculo Numerico	Fundamentos de la Informática
Matemática Discreta	Fundamentos de la Informática
Logica Computacional	Fundamentos de la Informática

Las asignaturas que no tienen materia asignada no salen en el listado

## Outer Join

- Hacer un listado de **todas** las asignaturas, que muestre el nombre de asignatura junto al nombre de la materia asociada

```
SELECT A.Nombre "ASIGNATURA", M.Nombre "MATERIA"  
FROM   Asignaturas A LEFT OUTER JOIN Materias M  
      ON (A.Cod_materia = M.Codigo);
```

Para las filas añadidas,  
los atributos de la otra  
tabla se asignan a **null**

ASIGNATURA	MATERIA
Programación Orientada a Objetos	Ingeniería del Software
Ingeniería Web	Ingeniería del Software
Bases de Datos	Ingeniería del Software
Administración de Bases de Datos	Ingeniería del Software
Computación Altas Prestaciones	Dispositivos hardware
Dispositivos Electronicos	Dispositivos hardware
Estructura de Computadores	Dispositivos hardware
Prácticas en empresa	Complementos de formación
Matemática Discreta	Fundamentos de la Informática
Estadística	Fundamentos de la Informática
Logica Computacional	Fundamentos de la Informática
Calculo Numerico	Fundamentos de la Informática
Modelos Computacionales	(null)
Sistemas Operativos	(null)
Teoria de Automatas	(null)
Teoria de la señal	(null)

## Outer Join

- Mostrar el código de alumno junto al código de profesor que le da clase de todos los alumnos y todos los profesores de las asignaturas 112 y 114.

```
SELECT I.Profesor, M.Alumno
FROM   Impartir I FULL OUTER JOIN Matricular M
      ON (I.Asignatura = M.Asignatura AND
          I.Grupo = M.Grupo AND
          I.Curso = M.Curso)
WHERE  nvl(I.Asignatura,0) in (112,114) OR
       nvl(M.Asignatura,0) in (112,114)
```

PROFESOR	ALUMNO
C-34-TU-00	12312342
G-34-TEU-96	12312342
D-92-PC-92	23423332
(null)	23423332
A-89-CEU-99	23423424
(null)	23456456
D-92-PC-92	25744665
C-34-TU-00	25748955
G-34-TEU-96	25748955
(null)	25756456
A-89-CEU-99	34234244
(null)	34234444
(null)	34545346
C-34-TU-00	34558955
C-34-TU-00	34567567
D-92-PC-92	45345311
(null)	45345311

...

- Salen
  - Los profesores de las asignaturas sin grupo asignado y/o sin alumnos en sus grupos.
  - Los alumnos matriculados en las asignaturas sin grupo asignado y/o sin profesor asignado.

## Ejercicio

1. Mostrar todos los datos de TODOS los profesores junto a los datos de sus directores de tesis, en caso de tenerlos.