



A lo largo de la sesión se va a trabajar con la base de datos correspondiente al modelo Entidad-Relación de la Fig.1, popularizada con los datos de la Tabla 1, y representados en los ficheros “universidad_estructura.sql” y “universidad_datos.sql”.

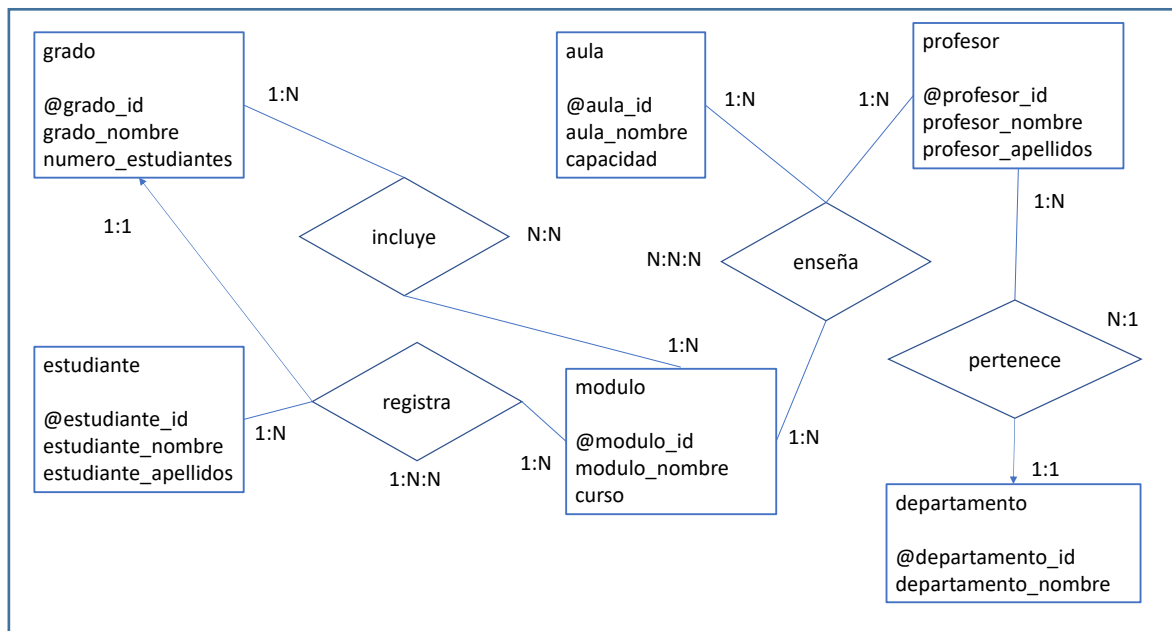


Figura 1. Modelo Entidad-Relación de la base de datos universidad.

El contenido se refleja en la Tabla 1.

Tabla 1. Datos de la base de datos universidad.

Tablas	Contenido
grado	1, Ingenieria Informatica en Tecnologias de la Informacion, 115
Atributos:	2, Ingenieria en Tecnologias y Servicios de Telecomunicacion, 110
• <i>grado_id (PK)</i>	3, Ingenieria Mecanica, 240
• <i>grado_nombre</i>	4, Ingenieria en Tecnologias Industriales, 145
• <i>numero_estudiantes</i>	5, Ingenieria Quimica Industrial, 60
	6, Ingenieria Electronica Industrial y Automatica, 110
	7, Ingenieria Electrica, 80
	8, Ingenieria Nautica y Transporte Maritimo, 35
	9, Ingenieria de Organización Industrial, 60
	10, Marina, 35
modulo	1, Calculo, 1
Atributos:	2, Estadistica, 1
• <i>modulo_id (PK)</i>	3, Algebra Lineal, 1
• <i>modulo_nombre</i>	4, Introduccion a la Programacion, 1
• <i>curso</i>	5, Estructura de Datos, 2
	6, Ingenieria del Software, 3
	7, Sistemas de Informacion, 3



	8, Tecnologías Web, 4
	9, Algoritmia, 2
	10, Fundamentos de Informatica 1
profesor	1, 1, Jorge Diez Pelaez
Atributos:	2, 1, Pedro Hernandez Arauzo
• <i>profesor_id</i> (PK)	3, 1, Alfredo Santiago Alguero Garcia
• <i>departamento_id</i> (PK)	4, 1, Cristian Augusto Alonso
• <i>profesor_nombre</i>	5, 1, Pablo Suarez-Otero Gonzalez
• <i>profesor_apellidos</i>	6, 1, Pablo Javier Tuya Gonzalez
	7, 2, Maria Zulima Fernandez Muñiz
	8, 2, Virginia Selgas Buznego
“profesor_id” y	9, 3, Hortensia Lopez Garcia
“departamento_id” forman la	10, 1, Maria Jose Suarez Cabal
clave primaria (PK) en la tabla	11, 1, Jose Garcia Fanjul
profesore, y además	12, 1, Maria Teresa Gonzalez Aparicio
“departamento_id” es una clave	13, 2, Matias Pico Serrano
ajena (FK) de la tabla	
departamento.	
departamento	1, Ciencias de la Computacion
Atributos:	2, Matematicas
• <i>departamento_id</i> (PK)	3, Estadistica
• <i>departamento_nombre</i>	4, Telecomunicacion
aula	1, AN-A, 60
Atributos:	2, PL1, 20
• <i>aula_id</i> (PK)	3, PL2, 20
• <i>aula_nombre</i>	4, PL3, 30
• <i>capacidad</i>	5, PL4, 30
	6, DO-9, 50
	7, DE-3, 15
	8, AN-E, 20
	9, AN-B, 60
estudiante	1, 'Diego', 'Ramos Mendez', true
Atributos:	2, 'Mercedes', 'Garcia Menendez', false
• <i>estudiante_id</i> (PK)	3, 'Maria', 'Alvarez Gomez', true
• <i>estudiante_nombre</i>	4, 'Sara', 'Prendes Pardo', true
• <i>estudiante_apellidos</i>	5, 'Juan', 'Prieto Vazquez', true
• <i>erasmus</i>	6, 'Alberto', 'Palacio Gomez', false
	7, 'Pedro', 'Gancedo Alvarez', true
estudiante_grado_modulo	*Diego (1), Ingenieria Informatica en Tecnologias de la Informacion (1), Calculo (1)
Atributos:	*Diego (1), Ingenieria Informatica en Tecnologias de la Informacion (1), Introduccion a la Programacion (4)
• <i>estudiante_id</i> (PK)	* Diego(1), Ingenieria Informatica en Tecnologias de la Informacion (1), Ingenieria del Software (6)
• <i>grado_id</i> (PK)	*Diego(1), Ingenieria Informatica en Tecnologias de la Informacion (1), Sistemas de Informacion (7)
• <i>modulo_id</i> (PK)	



<p>Three of them make the primary key in student_bachelor_module, but also they are foreign keys in relation with student, bachelor and module respectively.</p>	<p>*Diego(1), Ingeniería Informática en Tecnologías de la Información (1), Tecnologías Web (8) *Mercedes(2), Ingeniería Informática en Tecnologías de la Información (1), Cálculo (1) *Mercedes(2), Ingeniería Informática en Tecnologías de la Información (1), Ingeniería del Software (6) *Mercedes(2), Ingeniería Informática en Tecnologías de la Información (1), Tecnologías Web (8) *Mercedes(2), Ingeniería Informática en Tecnologías de la Información (1), Algoritmia (9) *María (3), Ingeniería Informática en Tecnologías de la Información (1), Cálculo (1) *María (3), Ingeniería Informática en Tecnologías de la Información (1), Estadística (2) *María (3), Ingeniería Informática en Tecnologías de la Información (1), Ingeniería del Software (6) *María (3), Ingeniería Informática en Tecnologías de la Información (1), Sistemas de Información (7) *María (3), Ingeniería Informática en Tecnologías de la Información (1), Tecnologías Web (8) *María (3), Ingeniería Informática en Tecnologías de la Información (1), Algoritmia (9) *Sara (4), Ingeniería Informática en Tecnologías de la Información (1), Cálculo (1) *Sara (4), Ingeniería Informática en Tecnologías de la Información (1), Estructura de Datos (5) *Sara (4), Ingeniería Informática en Tecnologías de la Información (1), Ingeniería del Software (6) *Sara (4), Ingeniería Informática en Tecnologías de la Información (1), Algoritmia (9) *Juan (5), Ingeniería Química Industrial (5), Cálculo (1) *Juan (5), Ingeniería Química Industrial (5), Estadística (2) *Alberto (6), Ingeniería Química Industrial (5), Cálculo (1) *Alberto (6), Ingeniería Química Industrial (5), Estadística (2) *Alberto (6), Ingeniería Química Industrial (5), Álgebra Lineal (3)</p>
<p>grado_modulo</p> <p>Atributos:</p> <ul style="list-style-type: none"> • bachelor_id (PK) • module_id (PK) <p>Ambos forman la clave primaria en grado_modulo, pero además son claves ajenas en relación a las tablas grado y modulo respectivamente.</p>	<p>*Todos los grados tienen Cálculo, Álgebra Lineal y Estadística. *Todos los grados, excepto el grado Ingeniería Informática en Tecnologías de la Información, tienen Fundamentos de Informática</p>
<p>modulo_profesor_aula</p>	<p>Estadística (2), Hortensia (9), AN-A (1), DE-3 (7)</p>



Atributos: <ul style="list-style-type: none"> • modulo_id (PK) • profesor_id (PK) • aula_id (PK) <p>Los tres atributos forman la clave primaria, y además son claves ajenas en las tablas modulo, profesor y aula respectivamente</p>	Algebra Lineal (3), Zulima (7), AN-A (1), DE-3 (7) Calculo (1), Virginia (8), AN-A (1), DE-3 (7) Tecnologías Web (8), Maria Jose (10), AN-E (8), AN-A (1) Algoritmia (9), Maria Teresa (12), DE-3 (7), AN-A (1) Sistemas de Informacion (7), Pablo Javier (6), DO-9 (6), AN-A (1) Estructura de Datos (5), Pedro Hernandez (2), PL1 (2), PL2 (3), AN-A (1) Introduccion a la Programacion (4), Jorge Diez (1), AN-A (1), PL3(4) Ingenieria del Software (6), Jose Garcia Fanjul (11), DO-9 (6), AN-A (1) Fundamentos de Informatica (10), Alfredo (3), Pablo (5) y Cristian (4), PL1(2), PL2 (3), PL3(4), PL4 (5), AN-A (1)
--	---

El principal objetivo de esta sesión es trabajar con procedimientos y funciones. En el fichero “Estudiante-FuncionesYProcedimientos.doc”, los estudiantes deberán de escribir la respuesta en PL/SQL a cada una de las cuestiones planteadas.

Cuestiones:

1. Cargar los ficheros “universidad_estructura.sql” y “universidad_datos.sql” (\i path\\<nombre fichero sql> o \i path/<nombre fichero sql>).
2. Escribir las siguientes consultas como parte de una función, y escribir la instrucción que la invoque.
2.1 Escribir una función denominada getInfoGrados() que liste el nombre y el número de estudiantes del grado que tenga el menor número de estudiantes.

```
select gra.grado_nombre, gra.numero_estudiantes from grado gra where
numero_estudiantes= (select min(numero_estudiantes) from grado);
```

```
getinfofgrados
-----
("Ingenieria Nautica y Transporte Maritimo",35)
(Marina,35)
(2 filas)
```

- 2.2. Amplia la consulta anterior, para ello escribe una función denominada getAulas() que liste solo aquellas aulas con un número de profesores mayor que 2.

```
select au.aula_nombre, count(*)
from aula au, profesor pro, modulo_profesor_aula mta
where au.aula_id=mta.aula_id and mta.profesor_id=pro.profesor_id
group by(au.aula_id)
having count(pro.profesor_id) >2;
```



Nota: El segundo parámetro de la tabla que se retornará debe ser definido de tipo bigint, en caso contrario se producirá un error.

```
getaulas
-----
(PL3,4)
(PL1,4)
(PL2,4)
(PL4,3)
(DE-3,4)
(AN-A,12)
(6 filas)
```

2.3. Escribe una función denominada getGradoMinEstudiantes() que presente el nombre del grado con el número más bajo de estudiantes. Recuerda que min(count(*)) no es posible.

```
select gra.grado_nombre, count(*) from grado gra inner join
estudiante_grado_modulo using(grado_id) inner join modulo mod
using(modulo_id) group by grado_id having (count(*)) <= all
(select count(*) from grado gra inner join estudiante_grado_modulo using
(grado_id) inner join modulo using(modulo_id) group by gra.grado_id);
```

Nota: El segundo parámetro de la tabla que se retornará debe ser definido de tipo bigint, en caso contrario se producirá un error.

```
getgradominestudiantes
-----
("Ingenieria Quimica Industrial",5)
(1 fila)
```

3. Convierte las siguientes consultas en procedimientos, y escribe la sentencia correspondiente que lo invoque.

3.1. Escribe un procedimiento denominado getCapacidadTotalAula() que devuelva por parámetro la capacidad total entre todas las aulas.

```
select sum(capacidad) from aula;
```

```
numero_asientos
-----
305
(1 fila)
```

3.2. Escribe un procedimiento denominada getEstudiantesIngenieriaIndustrial() que presente el nombre y los apellidos de aquellos estudiantes que estén estudiando 'Ingenieria Quimica Industrial', y también aquellos que son de erasmus.



```
select estudiante_nombre, estudiante_apellidos from estudiante est
inner join estudiante_grado_modulo egm using(estudiante_id)
inner join grado gra using(grado_id)
where lower(gra.grado_nombre)= 'Ingenieria Quimica Industrial'
union
select estudiante_nombre, estudiante_apellidos from estudiante est where
est.erasmus=true;
```

```
NOTICE: (Sara, Prendes Pardo)
NOTICE: (Juan, Prieto Vazquez)
NOTICE: (Pedro, Gancedo Alvarez)
NOTICE: (Maria, Alvarez Gomez)
CALL
```

3.3. Escribe un procedimiento denominado `getProfesoresComputacionNoAlgoritmia(n int)` que presente el nombre y los apellidos de los 'n' primeros profesores que pertenezcan al departamento 'Ciencias de la Computacion', pero sin tener en cuenta aquellos que imparten el módulo de 'Algoritmia'.

```
select profesor_nombre, profesor_apellidos from profesor pro
inner join departamento dep using(departamento_id)
where dep.departamento_nombre= 'Ciencias de la Computacion'
except
select profesor_nombre, profesor_apellidos from profesor pro
inner join modulo_profesor_aula mpa using(profesor_id)
inner join modulo mod using(modulo_id)
where lower(mod.modulo_nombre)='algoritmia'
```

Todos los que no imparten 'Algoritmia' serían:

profesor_nombre	profesor_apellidos
Jose	Garcia Fanjul
Pablo	Suarez-Otero Gonzalez
Maria Jose	Suarez Cabal
Cristian Augusto	Alonso
Alfredo Santiago	Alguero Garcia
Pablo Javier	Tuya Gonzalez
Jorge	Diez Pelaez
Pedro	Hernandez Arauzo

(8 filas)



De acuerdo al orden en que se han presentado los profesores que no imparten el módulo de ‘Algoritmia’, los cuatro primeros serían:

```
NOTICE: (Jose, Garcia Fanjul)
NOTICE: (Pablo, Suarez-Otero Gonzalez)
NOTICE: (Maria Jose, Suarez Cabal)
NOTICE: (Cristian Augusto, Alonso)
CALL
```

Al finalizar la sesión, renombra el fichero Estudiante-FuncionesYProcedimientos.doc con tu nombre y apellidos (nombre_apellido1_apellido2.doc), y súbelo al Campus Virtual.

Objetivos de aprendizaje:

- ✓ Crear funciones
- ✓ Crear procedimientos