

Grado en Ingeniería del Software  
Grado en Ingeniería del Software y Matemática Computacional

# Bases de Datos



## Práctica 1 Manejo Lenguaje SQL

Rafael Socas Gutiérrez  
Amador Maho Etohá

Curso 2020/21



## Datos de los Alumnos

#	Nombre y Apellidos	Curso
1		
2		
3		

## Instrucciones

- Completa la práctica en este mismo power point rellenando las páginas en blanco o incluyendo más páginas si necesitas más espacio para los pantallazos y las explicaciones.
- Una vez completado el power point, guárdalo en formato pdf. **A la plataforma BB sube el pdf resultante.**
- **IMPORTANTE.** Adjunta también un script con todas las queries SQL que has utilizado para resolver las diferentes preguntas.
- Rellene el nombre/apellidos y el curso de los participantes del grupo.
- Fecha máxima de entrega: **Domingo 28 de Marzo 2021 a las 23:59**

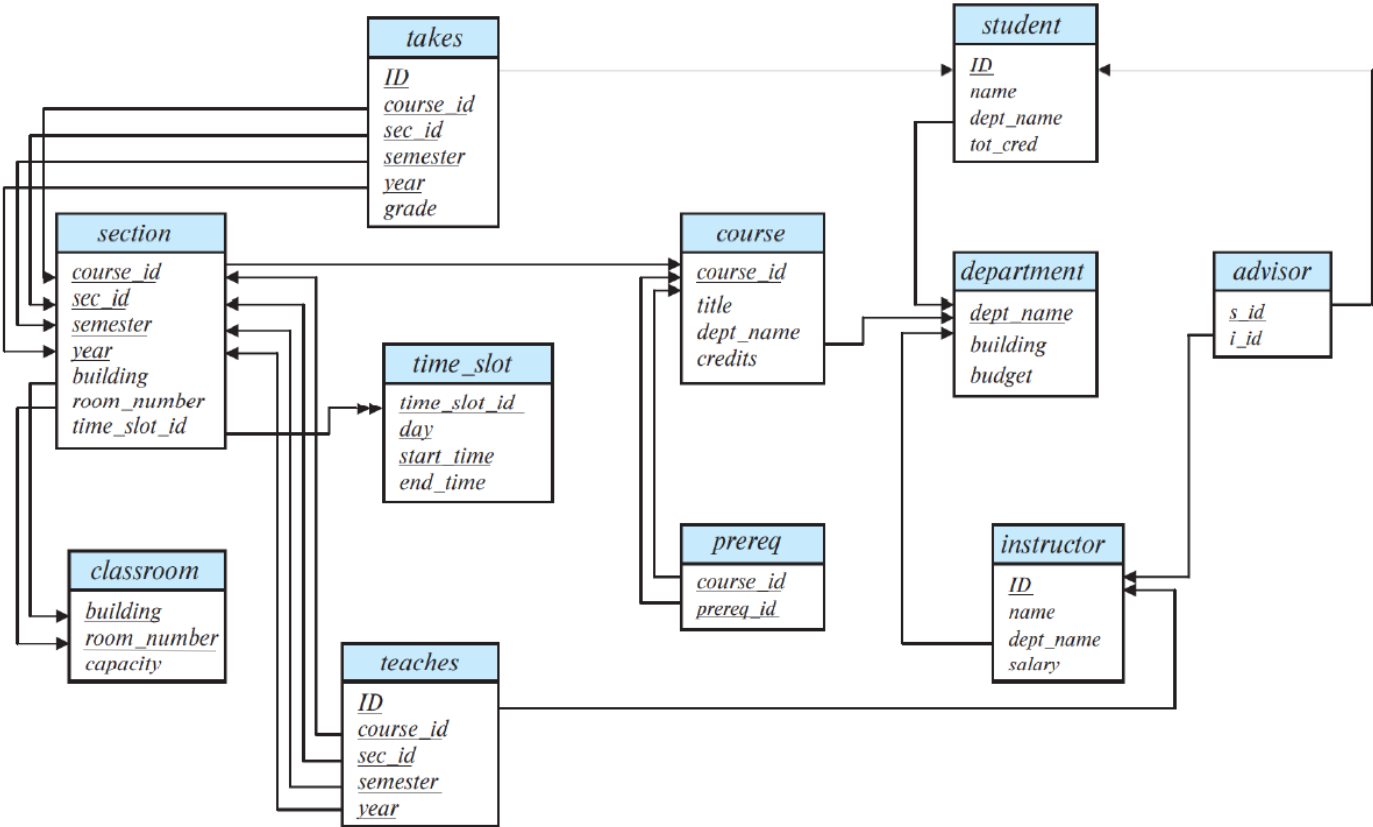
## Enunciado

Se dispone de la BBDD de la Universidad de Yale (<https://www.yale.edu/>) cuyo **Esquema** se presenta a continuación:

```

classroom(building, room_number, capacity)
department(dept_name, building, budget)
course(course_id, title, dept_name, credits)
instructor(ID, name, dept_name, salary)
section(course_id, sec_id, semester, year, building, room_number, time_slot_id)
teaches(ID, course_id, sec_id, semester, year)
student(ID, name, dept_name, tot_cred)
takes(ID, course_id, sec_id, semester, year, grade)
advisor(s_ID, i_ID)
time_slot(time_slot_id, day, start_time, end_time)
prereq(course_id, prereq_id)

```



## Enunciado

Para crear la BBDD anterior en el DBMS (MySQL) se deben seguir los siguientes pasos:

- Crear en MySQL una BBDD que se llame **university**.
- Los comandos SQL DDL para definir este esquema en el gestor de Base de Datos se encuentran en el fichero **DDL.sql**.
- Por otro lado, los comandos SQL DML para insertar los datos se proporcionan a su vez en el fichero **smallRelationsInsertFile.sql**.
- Ambos ficheros se proporcionan junto a este enunciado. También pueden descargarse del siguiente enlace [https://www.db-book.com/db7/university-lab-dir/sample\\_tables-dir/index.html](https://www.db-book.com/db7/university-lab-dir/sample_tables-dir/index.html)

Una vez tengamos la BBDD **university** en el gestor MySQL responda las cuestiones que se le plantean a continuación:



**1) PREGUNTA (1 punto):** En la tabla **instructor** de la BBDD university indique qué atributos tiene y de qué tipo de datos son cada uno de ellos. Al mismo tiempo indique qué atributo actúa como primary key y cuáles como foreign key si es que existen. Presente pantallazos de la respuesta de los comandos SQL que utiliza para obtener esta información.

```
mysql> describe instructor;
```

Field	Type	Null	Key	Default	Extra
ID	varchar(5)	NO	PRI	NULL	
name	varchar(20)	NO		NULL	
dept_name	varchar(20)	YES	MUL	NULL	
salary	decimal(8,2)	YES		NULL	

4 rows in set (0.00 sec)

ID es Primary Key

dept\_name es Foreign Key

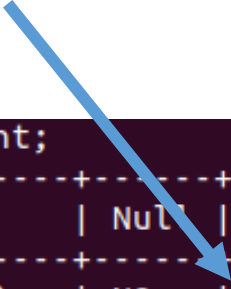
```
mysql> SELECT * from information_schema.table_constraints where table_name = 'instructor';
```

CONSTRAINT_CATALOG	CONSTRAINT_SCHEMA	CONSTRAINT_NAME	TABLE_SCHEMA	TABLE_NAME	CONSTRAINT_TYPE	ENFORCED
def	university	PRIMARY	university	instructor	PRIMARY KEY	YES
def	university	instructor_ibfk_1	university	instructor	FOREIGN KEY	YES
def	university	instructor_chk_1	university	instructor	CHECK	YES

3 rows in set (0.00 sec)

**2) PREGUNTA (1 punto):** Indague en el fichero **DDL.sql** para verificar que las respuestas dadas en la pregunta anterior respecto a la tabla **instructor** corresponden a los datos obtenidos. En caso de que existiera una foreign key, indique a que tabla hace referencia.

‘dept\_name’ es la clave foránea de la tabla ‘instructor’  
porque es la clave primaria de la tabla ‘department’



```
mysql> describe instructor;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| ID         | varchar(5)    | NO   | PRI | NULL    |       |
| name       | varchar(20)   | NO   |     | NULL    |       |
| dept_name  | varchar(20)   | YES  | MUL | NULL    |       |
| salary     | decimal(8,2)  | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> describe department;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| dept_name  | varchar(20)   | NO   | PRI | NULL    |       |
| building   | varchar(15)   | YES  |     | NULL    |       |
| budget     | decimal(12,2) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

```
mysql> SELECT * from information_schema.table_constraints where table_name = 'department';
+-----+-----+-----+-----+-----+-----+-----+
| CONSTRAINT_CATALOG | CONSTRAINT_SCHEMA | CONSTRAINT_NAME | TABLE_SCHEMA | TABLE_NAME | CONSTRAINT_TYPE | ENFORCED |
+-----+-----+-----+-----+-----+-----+-----+
| def                | university        | PRIMARY         | university    | department  | PRIMARY KEY     | YES      |
| def                | university        | department_chk_1 | university    | department  | CHECK           | YES      |
+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

**3) PREGUNTA (1 punto):** Con la tabla instructor, realice las siguientes consultas (queries):

- Muestres los instructores que pertenecen al departamento de “Physics”.
- Indique cuantos instructores hay.
- Ordene de mayor a menor los instructores en función de su salario.

```
mysql> SELECT * FROM instructor WHERE dept_name = 'Physics';
+-----+-----+-----+-----+
| ID    | name    | dept_name | salary |
+-----+-----+-----+-----+
| 22222 | Einstein | Physics   | 95000.00 |
| 33456 | Gold     | Physics   | 87000.00 |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

```
mysql> SELECT COUNT(DISTINCT ID) FROM instructor;
+-----+
| COUNT(DISTINCT ID) |
+-----+
| 12 |
+-----+
1 row in set (0.00 sec)
```

```
mysql> SELECT * FROM instructor ORDER BY salary DESC;
+-----+-----+-----+-----+
| ID    | name    | dept_name | salary |
+-----+-----+-----+-----+
| 22222 | Einstein | Physics   | 95000.00 |
| 83821 | Brandt   | Comp. Sci. | 92000.00 |
| 12121 | Wu       | Finance   | 90000.00 |
| 33456 | Gold     | Physics   | 87000.00 |
| 76543 | Singh    | Finance   | 80000.00 |
| 98345 | Kim      | Elec. Eng. | 80000.00 |
| 45565 | Katz      | Comp. Sci. | 75000.00 |
| 76766 | Crick     | Biology    | 72000.00 |
| 10101 | Srinivasan | Comp. Sci. | 65000.00 |
| 58583 | Califieri | History    | 62000.00 |
| 32343 | El Said   | History    | 60000.00 |
| 15151 | Mozart    | Music      | 40000.00 |
+-----+-----+-----+-----+
12 rows in set (0.00 sec)
```

**4) PREGUNTA (1 punto):** ¿En qué semestre, en que año y en qué grado se imparte el curso ID “CS-101” (revise la tabla takes)?. De las clases disponibles en la universidad, indique qué clase tiene mayor capacidad y cuál tiene menos (revise la tabla classroom).

```
mysql> SELECT DISTINCT grade, semester, year, course_id FROM takes WHERE course_id = 'CS-101';
+-----+-----+-----+-----+
| grade | semester | year | course_id |
+-----+-----+-----+-----+
| A     | Fall     | 2017 | CS-101     |
| C     | Fall     | 2017 | CS-101     |
| F     | Fall     | 2017 | CS-101     |
| A-    | Fall     | 2017 | CS-101     |
| C-    | Fall     | 2017 | CS-101     |
| B+    | Spring   | 2018 | CS-101     |
+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM classroom WHERE capacity = (SELECT MAX(capacity) FROM classroom);
+-----+-----+-----+
| building | room_number | capacity |
+-----+-----+-----+
| Packard  | 101         | 500      |
+-----+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> SELECT * FROM classroom WHERE capacity = (SELECT MIN(capacity) FROM classroom);
+-----+-----+-----+
| building | room_number | capacity |
+-----+-----+-----+
| Painter  | 514         | 10       |
+-----+-----+-----+
1 row in set (0.00 sec)
```



**5) PREGUNTA (1 punto):** Este ha sido un buen año para todos los departamentos de la universidad porque han ganado varios proyectos internacionales. Esto supondrá que el presupuesto que tienen asignado crecerá un 20%. Haga una query que presente el presupuesto actual de cada departamento y cuál sería el nuevo presupuesto con el incremento del 20% ordenado de mayor a menor (revise la tabla department).

```
mysql> SELECT * FROM department ORDER BY budget DESC;
+-----+-----+-----+
| dept_name | building | budget |
+-----+-----+-----+
| Finance   | Painter  | 120000.00 |
| Comp. Sci. | Taylor   | 100000.00 |
| Biology    | Watson   | 90000.00  |
| Elec. Eng. | Taylor   | 85000.00  |
| Music      | Packard  | 80000.00  |
| Physics    | Watson   | 70000.00  |
| History    | Painter  | 50000.00  |
+-----+-----+-----+
7 rows in set (0.00 sec)
```

Multiplicamos la columna 'budget' por 1,2

```
mysql> SELECT dept_name , building , (budget*1.2) FROM department ORDER BY budget DESC;
+-----+-----+-----+
| dept_name | building | (budget*1.2) |
+-----+-----+-----+
| Finance   | Painter  | 144000.000 |
| Comp. Sci. | Taylor   | 120000.000 |
| Biology    | Watson   | 108000.000 |
| Elec. Eng. | Taylor   | 102000.000 |
| Music      | Packard  | 96000.000  |
| Physics    | Watson   | 84000.000  |
| History    | Painter  | 60000.000  |
+-----+-----+-----+
7 rows in set (0.00 sec)
```

**6) PREGUNTA (1 punto):** Presente en una **sola query** los siguientes campos: name, dept\_name, tot\_cred (tabla student) y couse\_id, semester, year, grade (table takes) para el alumno Sanchez. Recuerde que tiene que extraer datos de dos tablas simultáneamente.

```
mysql> SELECT name, dept_name, tot_cred, course_id , semester, year, grade FROM student INNER JOIN takes WHERE student.name = 'Sanchez';
```

name	dept_name	tot_cred	course_id	semester	year	grade
Sanchez	Music	38	CS-101	Fall	2017	A
Sanchez	Music	38	CS-347	Fall	2017	A-
Sanchez	Music	38	CS-101	Fall	2017	C
Sanchez	Music	38	CS-190	Spring	2017	A
Sanchez	Music	38	CS-315	Spring	2018	A
Sanchez	Music	38	CS-347	Fall	2017	A
Sanchez	Music	38	HIS-351	Spring	2018	B
Sanchez	Music	38	FIN-201	Spring	2018	C+
Sanchez	Music	38	PHY-101	Fall	2017	B-
Sanchez	Music	38	CS-101	Fall	2017	F
Sanchez	Music	38	CS-101	Spring	2018	B+
Sanchez	Music	38	CS-319	Spring	2018	B
Sanchez	Music	38	CS-101	Fall	2017	A-
Sanchez	Music	38	CS-190	Spring	2017	B+
Sanchez	Music	38	MU-199	Spring	2018	A-
Sanchez	Music	38	CS-101	Fall	2017	A
Sanchez	Music	38	CS-319	Spring	2018	A
Sanchez	Music	38	EE-181	Spring	2017	C
Sanchez	Music	38	CS-101	Fall	2017	C-
Sanchez	Music	38	CS-315	Spring	2018	B
Sanchez	Music	38	BIO-101	Summer	2017	A
Sanchez	Music	38	BIO-301	Summer	2018	NULL

22 rows in set (0.00 sec)

**7) PREGUNTA (1 punto):** Repita la query anterior, es decir, presentar en una **sola query** los siguientes campos: name, dept\_name, tot\_cred (tabla student) y couse\_id, semester, year, grade (table takes) pero en este caso para el departamento de “Comp. Sci” (Computer Science). Recuerde que tiene que extraer datos de dos tablas simultáneamente.

```
mysql> SELECT name, dept_name, tot_cred, course_id , semester, year, grade FROM student INNER JOIN takes WHERE student.dept_name = 'Comp. Sci.';
```

name	dept_name	tot_cred	course_id	semester	year	grade
Brown	Comp. Sci.	58	CS-101	Fall	2017	A
Williams	Comp. Sci.	54	CS-101	Fall	2017	A
Shankar	Comp. Sci.	32	CS-101	Fall	2017	A
Zhang	Comp. Sci.	102	CS-101	Fall	2017	A
Brown	Comp. Sci.	58	CS-347	Fall	2017	A-
Williams	Comp. Sci.	54	CS-347	Fall	2017	A-
Shankar	Comp. Sci.	32	CS-347	Fall	2017	A-
Zhang	Comp. Sci.	102	CS-347	Fall	2017	A-
Brown	Comp. Sci.	58	CS-101	Fall	2017	C
Williams	Comp. Sci.	54	CS-101	Fall	2017	C
Shankar	Comp. Sci.	32	CS-101	Fall	2017	C
Zhang	Comp. Sci.	102	CS-101	Fall	2017	C
Brown	Comp. Sci.	58	CS-190	Spring	2017	A
Williams	Comp. Sci.	54	CS-190	Spring	2017	A
Shankar	Comp. Sci.	32	CS-190	Spring	2017	A
Zhang	Comp. Sci.	102	CS-190	Spring	2017	A
Brown	Comp. Sci.	58	CS-315	Spring	2018	A
Williams	Comp. Sci.	54	CS-315	Spring	2018	A
Shankar	Comp. Sci.	32	CS-315	Spring	2018	A
Zhang	Comp. Sci.	102	CS-315	Spring	2018	A
Brown	Comp. Sci.	58	CS-347	Fall	2017	A
Williams	Comp. Sci.	54	CS-347	Fall	2017	A
Shankar	Comp. Sci.	32	CS-347	Fall	2017	A
Zhang	Comp. Sci.	102	CS-347	Fall	2017	A
Brown	Comp. Sci.	58	HIS-351	Spring	2018	B
Williams	Comp. Sci.	54	HIS-351	Spring	2018	B
Shankar	Comp. Sci.	32	HIS-351	Spring	2018	B
Zhang	Comp. Sci.	102	HIS-351	Spring	2018	B
Brown	Comp. Sci.	58	FIN-201	Spring	2018	C+
Williams	Comp. Sci.	54	FIN-201	Spring	2018	C+
Shankar	Comp. Sci.	32	FIN-201	Spring	2018	C+
Zhang	Comp. Sci.	102	FIN-201	Spring	2018	C+
Brown	Comp. Sci.	58	PHY-101	Fall	2017	B-
Williams	Comp. Sci.	54	PHY-101	Fall	2017	B-
Shankar	Comp. Sci.	32	PHY-101	Fall	2017	B-
Zhang	Comp. Sci.	102	PHY-101	Fall	2017	B-
Brown	Comp. Sci.	58	CS-101	Fall	2017	F
Williams	Comp. Sci.	54	CS-101	Fall	2017	F
Shankar	Comp. Sci.	32	CS-101	Fall	2017	F
Zhang	Comp. Sci.	102	CS-101	Fall	2017	F
Brown	Comp. Sci.	58	CS-101	Spring	2018	B+
Williams	Comp. Sci.	54	CS-101	Spring	2018	B+

**8) PREGUNTA (1 punto):** Obtenga para el alumno “Sanchez” mediante una **SOLA** query que presente:

- name (tabla student)
- building, room\_number , capacity (tabla classroom)

En definitiva, mostrar el edificio, el número de la clase y la capacidad de la misma donde Sánchez recibe las clases.

**Ayuda:** Aunque vas a extraer datos de dos tablas, por la relación de las BBDD debes actuar sobre cuatro tablas simultáneamente.

```
mysql> SELECT name, building, room_number, capacity FROM student INNER JOIN classroom WHERE student.name = 'Sanchez';
```

name	building	room_number	capacity
Sanchez	Packard	101	500
Sanchez	Painter	514	10
Sanchez	Taylor	3128	70
Sanchez	Watson	100	30
Sanchez	Watson	120	50

```
5 rows in set (0.00 sec)
```



**9) PREGUNTA (1 punto):** Obtenga para el mismo alumno “Sanchez” mediante una **SOLA** query que presente :

- name (tabla student)
- name, dept\_name , salary (tabla instructor)

En definitiva, mostrar el profesor que imparte las clases al alumno Sanchez donde además se muestra el departamento y su salario.

**Ayuda:** Aunque vas a extraer datos de dos tablas, por la relación de las BBDD debes actuar sobre cinco tablas simultáneamente.

```
mysql> SELECT student.name, instructor.name, instructor.dept_name, salary FROM student INNER JOIN instructor WHERE student.name = 'Sanchez';
```

name	name	dept_name	salary
Sanchez	Srinivasan	Comp. Sci.	65000.00
Sanchez	Wu	Finance	90000.00
Sanchez	Mozart	Music	40000.00
Sanchez	Einstein	Physics	95000.00
Sanchez	El Said	History	60000.00
Sanchez	Gold	Physics	87000.00
Sanchez	Katz	Comp. Sci.	75000.00
Sanchez	Califieri	History	62000.00
Sanchez	Singh	Finance	80000.00
Sanchez	Crick	Biology	72000.00
Sanchez	Brandt	Comp. Sci.	92000.00
Sanchez	Kim	Elec. Eng.	80000.00

12 rows in set (0.00 sec)

**10) PREGUNTA (1 punto):** Obtenga para el alumno “Levy” mediante una **SOLA** query que presente :

- name, tot\_cred (tabla student)
- semester, year, grade (tabla takes)
- title, dept\_name , credits (tabla course)


En definitiva, mostrar el/los cursos que está haciendo el alumno Levy indicando también el semestre, año, etc..

**Ayuda:** Aunque vas a extraer datos de tres tablas, por la relación de las BBDD debes actuar sobre cuatro tablas simultáneamente.


```
mysql> SELECT * FROM (SELECT student.name, student.tot_cred, takes.semester, takes.year, takes.grade FROM student INNER JOIN takes W
HERE student.name = 'Levy') as t1 INNER JOIN (SELECT title, dept_name, credits FROM course) as t2;
```

name	tot_cred	semester	year	grade	title	dept_name	credits
Levy	46	Fall	2017	A	Physical Principles	Physics	4
Levy	46	Fall	2017	A	Music Video Production	Music	3
Levy	46	Fall	2017	A	World History	History	3
Levy	46	Fall	2017	A	Investment Banking	Finance	3
Levy	46	Fall	2017	A	Intro. to Digital Systems	Elec. Eng.	3
Levy	46	Fall	2017	A	Database System Concepts	Comp. Sci.	3
Levy	46	Fall	2017	A	Image Processing	Comp. Sci.	3
Levy	46	Fall	2017	A	Robotics	Comp. Sci.	3
Levy	46	Fall	2017	A	Game Design	Comp. Sci.	4
Levy	46	Fall	2017	A	Intro. to Computer Science	Comp. Sci.	4
Levy	46	Fall	2017	A	Computational Biology	Biology	3
Levy	46	Fall	2017	A	Genetics	Biology	4
Levy	46	Fall	2017	A	Intro. to Biology	Biology	4
Levy	46	Fall	2017	A	Physical Principles	Physics	4



 Calle Playa de Liencres, 2 bis  
(entrada por calle Rozabella)  
Parque Europa Empresarial  
Edificio Madrid  
28290 Las Rozas, Madrid

 900 373 379  [info@u-tad.com](mailto:info@u-tad.com)

 [SOLICITA MÁS INFORMACIÓN](#)



CENTRO ADSCRITO A:

 **Universidad  
Camilo José Cela**

PROYECTO COFINANCIADO POR:

