

```
1 -- (1) Seleccionar todas las columnas de la tabla de alumno
2 SELECT *
3 FROM alumno;
4
5 -- (2) Seleccionar únicamente las columnas clave y nombre de la tabla clase
6 SELECT clave, nombre
7 FROM clase;
8
9 -- (3) Ordenar los resultados anteriores por clave de forma ascendente
10 SELECT clave, nombre
11 FROM clase
12 ORDER BY clave ASC;
13
14 -- (4) Seleccionar las distintas áreas de investigación de los profesores y
15 --      ordenarlos de forma descendente
16 SELECT DISTINCT area_de_investigacion
17 FROM profesor
18 ORDER BY area_de_investigacion DESC;
19
20 -- (5) Seleccionar todos los salones, que se encuentran en Río Hondo
21 --      ordénalos de forma por aforo de mayor a menor
22 SELECT *
23 FROM salon
24 WHERE campus = 'RH'
25 ORDER BY aforo DESC;
26
27 -- (6) Selecciona el nombre y campus de todos los salones que no tienen computadoras
28 SELECT nombre, campus, tiene_computadoras
29 FROM salon
30 WHERE tiene_computadoras IS NOT TRUE
31 ORDER BY aforo DESC;
32
33 -- (7) Selecciona los salones que tienen un aforo mayor de 34 o que tienen
34 --      computadoras
35 --      ordénalos de forma descendente por aforo
36 SELECT *
37 FROM salon
38 WHERE tiene_computadoras IS TRUE OR aforo > 34
39 ORDER BY aforo DESC;
40
41 -- (8) Selecciona a todos los alumnos que no tienen apodo
42 --      ordénalos de forma ascendente por apellido y por nombre.
43 SELECT *
44 FROM alumno
45 WHERE apodo IS NULL
46 ORDER BY nombre ASC, apellido ASC;
47
48 -- (9) Selecciona a todos los alumnos que nacieron entre 1995 y 1999.
49 --      Ordénalos de mayor a menor edad. Si dos alumnos tienen la misma fecha de
50 --      nacimiento,
51 --      deben aparecer de manera descendente por apellido y luego nombre
52 SELECT *
53 FROM alumno
54 WHERE fecha_de_nacimiento BETWEEN '1995-01-01' AND '1999-12-31'
55 ORDER BY fecha_de_nacimiento ASC, nombre DESC, apellido DESC;
56
57 -- (10) Selecciona y concatena las iniciales de los profesores. Dale el nombre de '
58 --      iniciales' a la columna generada
59 SELECT DISTINCT substring(nombre, 1, 1) || substring(apellido, 1, 1) AS iniciales
60 FROM profesor;
61
62 -- (11) Selecciona el nombre y apodo de los alumnos cuyo apodo comience con la misma
63 --      letra que su nombre
64 SELECT nombre, apodo
65 FROM alumno
66 WHERE substring(nombre, 1, 1) = substring(apodo, 1, 1);
```

```
64
65 -- (12) Selecciona todos los profesores que su correo acabe con .org
66 SELECT *
67 FROM profesor
68 WHERE correo LIKE '%.org';
69
70 -- (13) Selecciona todos los alumnos que tengan las cadena 'an' o 'An' en algún
71 --      lugar de su nombre
72 SELECT *
73 FROM alumno
74 WHERE nombre ILIKE '%an%';
75
76 -- (14) Selecciona los distintos nombre de los alumnos que empiecen y terminen con
77 --      la letra 'a'
78 --      ordena los resultados de forma descendente
79 SELECT DISTINCT nombre
80 FROM alumno
81 WHERE nombre ILIKE 'a%a'
82 ORDER BY nombre DESC;
83
84 -- (15) Selecciona los 5 profesores mas jóvenes que se especialicen en economía o
85 --      derecho
86 SELECT *
87 FROM profesor
88 WHERE area_de_investigacion IN ('Economía', 'Derecho')
89 ORDER BY fecha_de_nacimiento DESC
90 LIMIT 5;
91
92 -- (16) Selecciona los distintos nombres de profesores cuya primera letra del nombre
93 --      sea la misma que la última
94 --      ordena el resultado de forma alfabética ascendente
95 SELECT DISTINCT nombre
96 FROM profesor
97 WHERE LOWER(LEFT(nombre, 1)) = LOWER(RIGHT(nombre, 1))
98 ORDER BY nombre ASC;
```