

```

1  /* BD.Evaluable3.1.DQL.DAM - JOSE MANUEL MORENO BOLIVAR - 52659570N */
2
3  /* CONSULTA 1B: CARTESIANO VS JOIN [2 puntos]
4  Mostrar el nombre completo (concatenando con comas los apellidos y el nombre) y la
5  fecha de grado de los maestros Jedi que fueron antiguos alumnos y han
6  impartido/imparten algún curso, ordenados por fecha de grado de más reciente a más
7  antigua.
8  Resultado: 12 filas.*/
9
10 /* 1B.1) Sin JOIN */
11
12 SELECT DISTINCT CONCAT('Maestro: ', m.nombre, ' ', m.apellidos) AS 'Nombre Completo',
13 m.fechagrado AS 'Fecha de grado'
14 FROM Maestro m, Curso c, Alumno a
15 WHERE concat(m.nombre, m.apellidos) = concat(a.nombre, a.apellidos)
16 AND m.idmaestro = c.idmaestro;
17
18 /* 1B.2) Con JOIN */
19
20 SELECT DISTINCT CONCAT('Maestro: ', m.nombre, ' ', m.apellidos) AS 'Nombre Completo',
21 m.fechagrado AS 'Fecha de grado'
22 FROM Maestro m
23 JOIN Curso c
24 JOIN Alumno a
25 ON concat(m.nombre, m.apellidos) = concat(a.nombre, a.apellidos)
26 AND m.idmaestro = c.idmaestro;
27
28 /* CONSULTA 2B: GROUP BY VS SUBCONSULTAS [2 puntos]
29 Obtener el concepto y la suma de todos los importes de los costes cuyos conceptos
30 empiezan por 'Alquiler' o 'Seguro'. Ordena los resultados de mayor a menor coste y
31 alfabéticamente por concepto.
32 Resultado: 4 filas. */
33
34 /* 2B.1) Con GROUP BY, sin SUBCONSULTA */
35
36 SELECT concepto, SUM(importe) AS importeTotal FROM coste
37 WHERE concepto LIKE 'Alquiler %'
38 OR concepto LIKE 'Seguro %'
39 GROUP BY concepto
40 ORDER BY importeTotal DESC, concepto;
41
42 /* 2B.2) Sin GROUP BY, con SUBCONSULTA */
43
44 SELECT DISTINCT c.concepto, ( SELECT SUM(importe)
45 FROM coste
46 WHERE concepto = c.concepto) AS importeTotal
47 FROM coste c
48 WHERE c.concepto LIKE 'Alquiler %'
49 OR c.concepto LIKE 'Seguro %'
50 ORDER BY importeTotal DESC, c.concepto;
51
52 /* CONSULTA 3B: EXISTS VS IN [2 puntos]
53 Listar los apellidos y el nombre (concatenados por coma) de los alumnos que hayan
54 sido Padawan y Senior, ordenados por orden alfabético inverso, es decir, ambos de la
55 Z a la A.
56 Resultado: 4 filas. */
57
58 /* 3B.1) Sin IN, con EXISTS */
59
60 SELECT CONCAT(a.apellidos, ', ', a.nombre) AS alumno
61 FROM alumno a
62 WHERE EXISTS ( SELECT s.idalumno
63 FROM senior s
64 WHERE s.idalumno = a.idalumno)
65 AND EXISTS (SELECT p.idalumno
66 FROM padawan p
67 WHERE p.idalumno = a.idalumno)
68 ORDER BY a.apellidos DESC;
69

```

```

65  /* 3B.2) Con IN, sin EXISTS */
66
67  SELECT CONCAT(a.apellidos, ', ', a.nombre) AS alumno FROM alumno a
68  WHERE idalumno IN (SELECT s.idalumno
69                      FROM senior s)
70  AND idalumno IN ( SELECT p.idalumno
71                  FROM padawan p)
72  ORDER BY a.apellidos DESC;
73
74
75
76  /* CONSULTA 4B: MÁXIMO DE UNA SUMA [2 puntos]
77  Indicar el nombre de la academia, los conceptos de gastos y el importe total de cada
78  uno en todas las academias Jedi. Ordena los resultados por academia en orden
79  alfabético, por importe de mayor a menor y por concepto. */
80
81  /* 4B.1) Muestra todos
82  Resultado: 26 filas. */
83
84  SELECT a.nombre, c.concepto, SUM(c.importe) AS Total
85  FROM academia a, coste c
86  WHERE c.codacad = a.codacad
87  GROUP BY a.nombre, c.concepto
88  ORDER BY Total;
89
90  /* 4B.2) Muestra solo el MÁXIMO COSTE de todos (sin usar LIMIT ni TOP)
91  Resultado: 1 fila. */
92
93  SELECT a.nombre, c.concepto, c.importe
94  FROM academia a, coste c
95  WHERE c.codacad = a.codacad
96  AND c.importe = ( SELECT MAX(c.importe)
97                  FROM coste c);
98
99
100 /* CONSULTA 5B: CREAR UNA VISTA Y USARLA [2 puntos]
101 Crea una vista (Cursos_con_alumnos) para obtener los cursos con alumnos. Del curso
102 nos interesa el nombre y descripción, del alumno su código y de ofertar la fecha de
103 inicio y fecha de fin. Los datos deben ordenarse por nombre y descripción. Una vez
104 creada, realiza una consulta sobre esa vista para obtener solo el nombre, descripción
105 y número de alumnos de los cursos que se hayan ofertado entre el 1/09/2022 y el
106 30/06/2023 y tengan más de 1 alumno.*/
107
108 /* 5B.1) Crea la vista y lista todos sus elementos
109 Resultado: 9 filas. */
110
111 DROP VIEW IF EXISTS Cursos_con_alumnos;
112
113 CREATE VIEW
114 Cursos_con_alumnos AS
115 SELECT c.nombre, c.descripcion, a.idalumno, o.fechaini, o.fechafin
116 FROM curso c, ofertar o, alumno a
117 WHERE a.codcurso = c.codcurso
118 AND c.codcurso = o.codcurso
119 AND a.codacad = c.codacad
120 AND c.codacad = o.codacad
121 ORDER BY c.nombre, c.descripcion;
122
123 /* 5B.2) Crea una consulta sobre esa vista
124 Resultado: 2 filas. */
125
126 SELECT nombre, descripcion, COUNT(*) AS n_alumnos
127 FROM Cursos_con_alumnos
128 WHERE fechaini BETWEEN '2022-09-01'
129                AND '2023-06-30'
130 GROUP BY nombre, descripcion
131 HAVING n_alumnos > 1;

```